Supplementary file for Bessho and Otto, “Fixation probability in a haploid-diploid population”.

Here we put supplementary information for our manuscript, “Fixation probability in a haploid-diploid population”.
Note that, you might have to reset the variables as necessary because we reuse same variables in the different sections.

Deriving the fixation probability in the haploid-diploid Moran model

▪ Frequency of haploids in the resident population

Original Equations (Eqs.(A.7))

$$c_R = \beta_{RR} * x_{RR} + \frac{\beta_{RM}}{2} * x_{RM};$$
$$c_M = \frac{\beta_{RM}}{2} * x_{RM} + \beta_{MM} * x_{MM};$$
$$c_{RR} = \frac{f_R}{2} * \frac{\beta_R{}^2 * x_R{}^2}{\beta_R * x_R + \beta_M * x_M};$$
$$c_{RM} = \frac{f_R + f_M}{2} * \frac{\beta_R * \beta_M * x_R * x_M}{\beta_R * x_R + \beta_M * x_M};$$
$$c_{MM} = \frac{f_M}{2} * \frac{\beta_M{}^2 * x_M{}^2}{\beta_R * x_R + \beta_M * x_M};$$

$$\gamma = \frac{d_R * x_R + d_M * x_M + d_{RR} * x_{RR} + d_{RM} * x_{RM} + d_{MM} * x_{MM}}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}};$$
$$d_{mean} = \frac{d_R * x_R + d_M * x_M + d_{RR} * x_{RR} + d_{RM} * x_{RM} + d_{MM} * x_{MM}}{x_R + x_M + x_{RR} + x_{RM} + x_{MM}};$$
$$\text{eqHR} = \frac{\gamma * c_R - d_R * x_R}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$
$$\text{eqHM} = \frac{\gamma * c_M - d_M * x_M}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$
$$\text{eqDRR} = \frac{\gamma * c_{RR} - d_{RR} * x_{RR}}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$
$$\text{eqDRM} = \frac{\gamma * c_{RM} - d_{RM} * x_{RM}}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$
$$\text{eqDMM} = \frac{\gamma * c_{MM} - d_{MM} * x_{MM}}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$
$$\text{HRnext} = \text{eqHR} + x_R;$$
$$\text{HMnext} = \text{eqHM} + x_M;$$
$$\text{DRRnext} = \text{eqDRR} + x_{RR};$$
$$\text{DRMnext} = \text{eqDRM} + x_{RM};$$
$$\text{DMMnext} = \text{eqDMM} + x_{MM};$$

Calculating the frequency of haploids within a population composed entirely of residents (ρ_H), assuming the population size is large:

$$\text{assumption} = 0 < \text{SUM} \;\&\&\; 0 < \beta_R \;\&\&\; 0 < \beta_{RR} \;\&\&\; 0 < d_R \;\&\&\; 0 < d_{RR} \;\&\&\; 0 < f_R \;\&\&\; f_R < 1;$$
$$\text{assumption2} = 0 < p_M \;\&\&\; p_M < 1 \;\&\&\; 0 < \rho_H \;\&\&\; \rho_H < 1;$$

$$\text{Factor[Normal[Series[eqHR /. }\{x_M \to 0,\ x_{RM} \to 0,\ x_{MM} \to 0\}\text{ /. } x_{RR} \to N\,(1 - \rho_H)\text{ /. } x_R \to N\,\rho_H\text{ /. } N \to N/\epsilon,\ \{\epsilon,\ 0,\ 0\}]]]$$

$$\frac{2\, d_{RR}\, \beta_{RR} - 4\, d_{RR}\, \beta_{RR}\, \rho_H - d_R\, f_R\, \beta_R\, \rho_H^2 + 2\, d_{RR}\, \beta_{RR}\, \rho_H^2}{(-d_{RR} - d_R\, \rho_H + d_{RR}\, \rho_H)\;(-2\, \beta_{RR} - f_R\, \beta_R\, \rho_H + 2\, \beta_{RR}\, \rho_H)}$$

$$\text{Simplify[Solve[\% == 0, }\rho_H\text{]]}$$

$$\left\{\left\{\rho_H \to -\frac{\sqrt{2}\,\sqrt{d_R}\,\sqrt{d_{RR}}\,\sqrt{f_R}\,\sqrt{\beta_R}\,\sqrt{\beta_{RR}} + 2\, d_{RR}\, \beta_{RR}}{d_R\, f_R\, \beta_R - 2\, d_{RR}\, \beta_{RR}}\right\},\ \left\{\rho_H \to \frac{\sqrt{2}\,\sqrt{d_R}\,\sqrt{d_{RR}}\,\sqrt{f_R}\,\sqrt{\beta_R}\,\sqrt{\beta_{RR}} - 2\, d_{RR}\, \beta_{RR}}{d_R\, f_R\, \beta_R - 2\, d_{RR}\, \beta_{RR}}\right\}\right\}$$

The second of these can be rewritten as the strictly positive quantity (A.8b)

$$\frac{\sqrt{\beta_{RR} * d_{RR}}}{\sqrt{\frac{f_R * \beta_R * d_R}{2}} + \sqrt{\beta_{RR} * d_{RR}}};$$

$$\text{Simplify}\left[\frac{\sqrt{2}\ \sqrt{d_R}\ \sqrt{d_{RR}}\ \sqrt{f_R}\ \sqrt{\beta_R}\ \sqrt{\beta_{RR}} - 2\ d_{RR}\ \beta_{RR}}{d_R\ f_R\ \beta_R - 2\ d_{RR}\ \beta_{RR}} - \%,\ \text{assumption}\right]$$

0

- **Branching Process Approximation**

Here we check the main calculations derived in Appendix B.

Checking the derivation of Eq. (B.5).

The probability that the allele will be lost if it first appears in a haploid is equal to the sum of the following three events in the first generation:

(a) the mutant is lost immediately with probability $\tilde{d}_{MH}$
(b) another individual within the population dies (probability $1 - \tilde{d}_{MH}$) and is replaced with anybody but a diploid offspring of the haploid mutant parent, so that again we have only one haploid mutant that must ultimately be lost $(1 - d_{MH})\ (1 - \tilde{p}_{RMH})\ \pi 0H$
(c) another individual within the population dies (probability $1 - \tilde{d}_{MH}$) and is replaced with a diploid offspring of the haploid mutant parent with loss probability now equalling $\tilde{d}_{RH}\ \tilde{p}_{RMH}\ \pi 0H\ \pi 0D$, where loss now requires both the haploid mutant parent and the mutant diploid offspring to be ultimately lost (here assumed to be independent probabilities)

This gives Eq. (B.4), and (B.5) is derived similarly. These two equations can be solved to yield:

$$\text{lossprob} = \text{Simplify}\Big[\text{Solve}\Big[\Big\{\pi 0H == \tilde{d}_{MH} + \left(1 - \tilde{d}_{MH}\right) * (1 - \tilde{p}_{RMH}) * \pi 0H + \left(1 - \tilde{d}_{MH}\right) * \tilde{p}_{RMH} * \pi 0H * \pi 0D,$$
$$\pi 0D == \tilde{d}_{RMD} + \left(1 - \tilde{d}_{RMD}\right) * (1 - \tilde{p}_{MD}) * \pi 0D + \left(1 - \tilde{d}_{RMD}\right) * \tilde{p}_{MD} * \pi 0H * \pi 0D\Big\},\ \{\pi 0H,\ \pi 0D\}\Big]\Big]$$

$$\{\{\pi 0H \to 1,\ \pi 0D \to 1\},$$
$$\left\{\pi 0H \to -\frac{\tilde{d}_{MH}\ \left(\tilde{d}_{RMD}\ (-1 + \tilde{p}_{MD}) - \tilde{p}_{MD}\right)}{\left(-1 + \tilde{d}_{RMD}\right)\ \tilde{p}_{MD}\ \left(\tilde{d}_{MH}\ (-1 + \tilde{p}_{RMH}) - \tilde{p}_{RMH}\right)},\ \pi 0D \to \frac{\tilde{d}_{RMD}\ \left(-\tilde{d}_{MH}\ (-1 + \tilde{p}_{RMH}) + \tilde{p}_{RMH}\right)}{\left(-1 + \tilde{d}_{MH}\right)\ \left(\tilde{d}_{RMD}\ (-1 + \tilde{p}_{MD}) - \tilde{p}_{MD}\right)\ \tilde{p}_{RMH}}\right\}\}$$

The relevant solution is the second one. Calculating the probability of fixation from these probabilities of loss, we obtain equations (B.6):

$$\text{EqB6a} = \text{Simplify}\left[\frac{\tilde{p}_{RMH} * \left(1 - \tilde{d}_{MH}\right) - \frac{\tilde{d}_{MH} * \tilde{d}_{RMD}}{\tilde{p}_{MD} * \left(1 - \tilde{d}_{RMD}\right)}}{\tilde{p}_{RMH} * \left(1 - \tilde{d}_{MH}\right) + \tilde{d}_{MH}},\ 0 < d_R\right];$$

$$\text{EqB6b} = \text{Simplify}\left[\frac{\tilde{p}_{MD} * \left(1 - \tilde{d}_{RMD}\right) - \frac{\tilde{d}_{MH} * \tilde{d}_{RMD}}{\tilde{p}_{RMH} * \left(1 - \tilde{d}_{MH}\right)}}{\tilde{p}_{MD} * \left(1 - \tilde{d}_{RMD}\right) + \tilde{d}_{RMD}},\ 0 < d_R\right];$$

```
EqB6a - (1 - π0H) /. lossprob[[2]] // Factor
EqB6b - (1 - π0D) /. lossprob[[2]] // Factor
```

0

0

Defining the variables as in Appendix B:

$$\text{assumption} = 0 < \text{SUM} \,\&\&\, 0 < d_R \,\&\&\, 0 < d_{RR} \,\&\&\, 0 < f_R \,\&\&\, f_R < 1 \,\&\&\, 0 < \beta_R \,\&\&\, 0 < \beta_{RR} \,\&\&\, 0 < d_M \,\&\&\, 0 < d_{RM} \,\&\&\, 0 < f_M \,\&\&\, f_M < 1 \,\&\&\, 0 < \beta_M \,\&\&\, 0 < \beta_{RM};$$

$$\text{freqH} = \frac{\sqrt{\beta_{RR} * d_{RR}}}{\sqrt{\frac{f_R*\beta_R*d_R}{2}} + \sqrt{\beta_{RR} * d_{RR}}};$$

$$\text{freqD} = \frac{\sqrt{\frac{f_R*\beta_R*d_R}{2}}}{\sqrt{\frac{f_R*\beta_R*d_R}{2}} + \sqrt{\beta_{RR} * d_{RR}}};$$

$$\hat{\beta} = \frac{f_R * \beta_R}{2} * \hat{\rho}_H + \beta_{RR} * \hat{\rho}_D \;/.\; \hat{\rho}_H \to \text{freqH} \;/.\; \hat{\rho}_D \to \text{freqD};$$

$$\tilde{p}_{RMH} = \frac{(f_R + f_M) * \beta_M}{2 * \hat{\beta} * \text{SUM}};$$

$$\tilde{p}_{MD} = \frac{\beta_{RM}}{2 * \hat{\beta} * \text{SUM}};$$

$$\hat{d} = d_R * \hat{\rho}_H + d_{RR} * \hat{\rho}_D \;/.\; \hat{\rho}_H \to \text{freqH} \;/.\; \hat{\rho}_D \to \text{freqD};$$

$$\tilde{d}_{MH} = \frac{d_M}{\hat{d} * \text{SUM}};$$

$$\tilde{d}_{RMD} = \frac{d_{RM}}{\hat{d} * \text{SUM}};$$

$$\text{FaiR} = \frac{\beta_R}{d_R};$$

$$\text{FaiM} = \frac{\beta_M}{d_M};$$

$$\text{FaiRR} = \frac{\beta_{RR}}{d_{RR}};$$

$$\text{FaiRM} = \frac{\beta_{RM}}{d_{RM}};$$

$$\text{avef} = \frac{f_M + f_R}{2};$$

$$\text{FaiR2} = \frac{f_R}{2} * \text{FaiR};$$

$$\text{FaiM2} = \frac{\text{avef}}{2} * \text{FaiM};$$

$$\text{sMoran} = (\tilde{s}_{fM} / 2 + \tilde{s}_{\beta M} + \tilde{s}_{dM} + \tilde{s}_{\beta RM} + \tilde{s}_{dRM});$$

$$\text{sMoranAverage} = \frac{\text{sMoran}}{2};$$

We now check derivation of Eq. (B.7) from Eq. (B.6)

```
EqB6a2 = Limit[EqB6a, SUM → Infinity];
EqB6b2 = Limit[EqB6b, SUM → Infinity];
EqB7a = Simplify[(p̃MD * p̃RMH - d̃MH * d̃RMD) / (p̃MD * (p̃RMH + d̃MH)), 0 < dR];
EqB7b = Simplify[(p̃RMH * p̃MD - d̃MH * d̃RMD) / (p̃RMH * (p̃MD + d̃RMD)), 0 < dR];
Simplify[EqB7a - EqB6a2]
Simplify[EqB7b - EqB6b2]
```

0

0

These can also be written in the form given by Eq. (1):

```
Eq1a = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + FaiRM/2 * Sqrt[FaiR2 * FaiRR]);
Eq1b = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + 2 * FaiM2 * Sqrt[FaiR2 * FaiRR]);

FullSimplify[EqB7a - Eq1a, assumption]
Simplify[EqB7b - Eq1b, assumption]
```

0

0

Finally, we obtain Eq. (B.8) by assuming weak selection:

```
weakselection = {βRM → βRR * (1 + ϵ * s̃βRM), βMM → βRR * (1 + ϵ * s̃βMM), βM → βR * (1 + ϵ * s̃βM),
   fM → fR * (1 + ϵ * s̃fM), dRM → dRR * (1 - ϵ * s̃dRM), dMM → dRR * (1 - ϵ * s̃dMM), dM → dR * (1 - ϵ * s̃dM)};

Simplify[Normal[Series[Eq1a /. weakselection, {ϵ, 0, 1}]], assumption] /. ϵ → 1
Simplify[Normal[Series[Eq1b /. weakselection, {ϵ, 0, 1}]], assumption] /. ϵ → 1
```

$$\frac{f_R\,\beta_R\,(2\,\tilde{s}_{dM}+2\,\tilde{s}_{dRM}+\tilde{s}_{fM}+2\,\tilde{s}_{\beta M}+2\,\tilde{s}_{\beta RM})}{2\,f_R\,\beta_R+\sqrt{2}\sqrt{\frac{d_R\,f_R\,\beta_R\,\beta_{RR}}{d_{RR}}}}$$

$$\frac{\beta_{RR}\,(2\,\tilde{s}_{dM}+2\,\tilde{s}_{dRM}+\tilde{s}_{fM}+2\,\tilde{s}_{\beta M}+2\,\tilde{s}_{\beta RM})}{2\left(\beta_{RR}+\sqrt{2}\sqrt{\frac{d_{RR}\,f_R\,\beta_R\,\beta_{RR}}{d_R}}\right)}$$

which can be written as:

```
EqB8a = (4 * dRR * freqD) / (dR * freqH + 2 * dRR * freqD) * sMoranAverage;
EqB8b = (2 * dR * freqH) / (dR * freqH + 2 * dRR * freqD) * sMoranAverage;
Simplify[Normal[Series[Eq1a - EqB8a /. weakselection, {ϵ, 0, 1}]], assumption] /. ϵ → 1
Simplify[Normal[Series[Eq1b - EqB8b /. weakselection, {ϵ, 0, 1}]], assumption] /.
  ϵ → 1
```

0

0

- **Diffusion Approximation (First Moment)**

- **Diffusion Approximation (First Moment / Separation of Time Scale)**

We first derive the first moment of change in mutant allele frequency appying a separation of time scales.

Original Equations (Eqs.(A.7))

$$c_R = \beta_{RR} * x_{RR} + \frac{\beta_{RM}}{2} * x_{RM};$$

$$c_M = \frac{\beta_{RM}}{2} * x_{RM} + \beta_{MM} * x_{MM};$$

$$c_{RR} = \frac{f_R}{2} * \frac{\beta_R{}^2 * x_R{}^2}{\beta_R * x_R + \beta_M * x_M};$$

$$c_{RM} = \frac{f_R + f_M}{2} * \frac{\beta_R * \beta_M * x_R * x_M}{\beta_R * x_R + \beta_M * x_M};$$

$$c_{MM} = \frac{f_M}{2} * \frac{\beta_M{}^2 * x_M{}^2}{\beta_R * x_R + \beta_M * x_M};$$

$$\gamma = \frac{d_R * x_R + d_M * x_M + d_{RR} * x_{RR} + d_{RM} * x_{RM} + d_{MM} * x_{MM}}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}};$$

$$d_{mean} = \frac{d_R * x_R + d_M * x_M + d_{RR} * x_{RR} + d_{RM} * x_{RM} + d_{MM} * x_{MM}}{x_R + x_M + x_{RR} + x_{RM} + x_{MM}};$$

$$\text{eqHR} = \frac{\gamma * c_R - d_R * x_R}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$

$$\text{eqHM} = \frac{\gamma * c_M - d_M * x_M}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$

$$\text{eqDRR} = \frac{\gamma * c_{RR} - d_{RR} * x_{RR}}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$

$$\text{eqDRM} = \frac{\gamma * c_{RM} - d_{RM} * x_{RM}}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$

$$\text{eqDMM} = \frac{\gamma * c_{MM} - d_{MM} * x_{MM}}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$

$\text{HRnext} = \text{eqHR} + x_R;$
$\text{HMnext} = \text{eqHM} + x_M;$
$\text{DRRnext} = \text{eqDRR} + x_{RR};$
$\text{DRMnext} = \text{eqDRM} + x_{RM};$
$\text{DMMnext} = \text{eqDMM} + x_{MM};$

Transform variables
(Define: pM, δp, ρH, FD)
(SUM=N in the manuscript)

$$\text{set} = \Bigg\{ \text{SUM} == x_R + x_M + x_{RR} + x_{RM} + x_{MM},$$

$$p_M == \frac{d_{RR}}{d_R + d_{RR}} * \frac{x_M}{x_R + x_M} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR} + x_{RM} + x_{MM}},$$

$$\delta_p == \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR} + x_{RM} + x_{MM}} - \frac{x_M}{x_R + x_M},$$

$$\rho_H == \frac{x_R + x_M}{x_R + x_M + x_{RR} + x_{RM} + x_{MM}},$$

$$F_D == 1 - \frac{1}{2 * \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR} + x_{RM} + x_{MM}} * \left(1 - \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR} + x_{RM} + x_{MM}}\right)} * \frac{x_{RM}}{x_{RR} + x_{RM} + x_{MM}}$$

$$\Bigg\};$$

OldToNew: Transformation of old variables to new variabls
NewToOld: Transformation of new variables to old variabls

```
OldToNew = Solve[set, {xR, xM, xRR, xRM, xMM}] // Flatten
NewToOld = Solve[set, {SUM, pM, ρH, δp, FD}] // Flatten
```

$$\Big\{x_R \to -\frac{\mathrm{SUM}\,(-d_R - d_{RR} + d_R p_M + d_{RR} p_M - d_R \delta_p)\,\rho_H}{d_R + d_{RR}},$$
$$x_M \to \frac{\mathrm{SUM}\,(d_R p_M + d_{RR} p_M - d_R \delta_p)\,\rho_H}{d_R + d_{RR}},\ x_{RR} \to \frac{1}{(d_R + d_{RR})^2}\,\mathrm{SUM}\,(-d_R - d_{RR} + d_R p_M + d_{RR} p_M + d_{RR}\delta_p)$$
$$(d_R + d_{RR} - d_R p_M - d_{RR} p_M + d_R F_D p_M + d_{RR} F_D p_M - d_{RR}\delta_p + d_{RR} F_D \delta_p)\,(-1 + \rho_H),$$
$$x_{RM} \to -\frac{1}{(d_R + d_{RR})^2}\,2\,\mathrm{SUM}\,(-1 + F_D)\,\big(-d_R^2 p_M - 2 d_R d_{RR} p_M - d_{RR}^2 p_M + d_R^2 p_M^2 + 2 d_R d_{RR} p_M^2 + d_{RR}^2 p_M^2 - d_R d_{RR}\delta_p -$$
$$d_{RR}^2\delta_p + 2 d_R d_{RR} p_M \delta_p + 2 d_{RR}^2 p_M \delta_p + d_{RR}^2 \delta_p^2\big)\,(-1 + \rho_H),\ x_{MM} \to \frac{1}{(d_R + d_{RR})^2}\,\mathrm{SUM}\,(d_R p_M + d_{RR} p_M + d_{RR}\delta_p)$$
$$(-d_R F_D - d_{RR} F_D - d_R p_M - d_{RR} p_M + d_R F_D p_M + d_{RR} F_D p_M - d_{RR}\delta_p + d_{RR} F_D \delta_p)\,(-1 + \rho_H)\Big\}$$

$$\Big\{\mathrm{SUM} \to x_M + x_{MM} + x_R + x_{RM} + x_{RR},\ p_M \to \frac{d_{RR}\,x_M}{(d_R + d_{RR})\,(x_M + x_R)} + \frac{d_R\left(x_{MM} + \frac{x_{RM}}{2}\right)}{(d_R + d_{RR})\,(x_{MM} + x_{RM} + x_{RR})},$$
$$\rho_H \to \frac{x_M + x_R}{x_M + x_{MM} + x_R + x_{RM} + x_{RR}},\ \delta_p \to \frac{2 x_{MM} x_R - x_M x_{RM} + x_R x_{RM} - 2 x_M x_{RR}}{2\,(x_M + x_R)\,(x_{MM} + x_{RM} + x_{RR})},\ F_D \to \frac{-x_{RM}^2 + 4 x_{MM} x_{RR}}{(2 x_{MM} + x_{RM})\,(x_{RM} + 2 x_{RR})}\Big\}$$

Dynamics for the new variables

Dynamics pM

```
EqpMOLD = dRR/(dR + dRR) * HMnext/(HRnext + HMnext) + dR/(dR + dRR) * (DRMnext/2 + DMMnext)/(DRRnext + DRMnext + DMMnext) - pM;
EqpMNEW = EqpMOLD /. OldToNew;
```

Dynamics δp

```
EqδpOLD = ((DRMnext/2 + DMMnext)/(DMMnext + DRMnext + DRRnext) - HMnext/(HMnext + HRnext)) - δp;
EqδpNEW = EqδpOLD /. OldToNew;
```

Dynamics ρH

```
EqρHOLD = ((HMnext + HRnext)/SUM) - ρH;
EqρHNEW = EqρHOLD /. OldToNew;
```

Dynamics FD

```
EqFDOLD = (1 - (1/(2 * (DRMnext/2 + DMMnext)/(DRRnext + DRMnext + DMMnext) * (1 - (DRMnext/2 + DMMnext)/(DRRnext + DRMnext + DMMnext))) * DRMnext/(DRRnext + DRMnext + DMMnext))) -
   FD;
EqFDNEW = EqFDOLD /. OldToNew;
```

We next transform original equations to new variables system under the weak selection

Asuming weak selection (Eqs. (C.1))

```
weakselection = {βRM → βRR * (1 + ϵ * s̃βRM), βMM → βRR * (1 + ϵ * s̃βMM), βM → βR * (1 + ϵ * s̃βM),
   fM → fR * (1 + ϵ * s̃fM), dRM → dRR * (1 - ϵ * s̃dRM), dMM → dRR * (1 - ϵ * s̃dMM), dM → dR * (1 - ϵ * s̃dM)};
EqC1a = EqpMNEW /. weakselection;
EqC1b = EqδpNEW /. weakselection;
EqC1c = EqρHNEW /. weakselection;
EqC1d = EqFDNEW /. weakselection;
```

Eqs. C2
Order(1): ϵ=0

```
EqC2a = Simplify[EqC1a /. ϵ → 0, assumption && assumption2]
EqC2b = Simplify[EqC1b /. ϵ → 0, assumption && assumption2]
EqC2c = Simplify[EqC1c /. ϵ → 0, assumption && assumption2]
EqC2d = Simplify[EqC1d /. ϵ → 0, assumption && assumption2]
```

$$\begin{aligned}
&\Big(\delta_p\,(d_{RR}\,(-1+\rho_H)-d_R\,\rho_H)\\
&\quad\big(2\,d_{RR}\,\beta_{RR}\,(-1+\rho_H)^2-d_R\,f_R\,\beta_R\,\rho_H^2\big)\,\big(d_R\,\rho_H\,(-2\,\text{SUM}\,\beta_{RR}\,(-1+\rho_H)+f_R\,\beta_R\,(-1+\text{SUM}\,\rho_H))+\\
&\qquad d_{RR}\,(-1+\rho_H)\,(-\text{SUM}\,f_R\,\beta_R\,\rho_H+2\,\beta_{RR}\,(1-\text{SUM}+\text{SUM}\,\rho_H))\big)\Big)\Big/\\
&\Big((d_R+d_{RR})\,\big(d_R\,\rho_H^2\,(2\,\text{SUM}\,\beta_{RR}\,(-1+\rho_H)+f_R\,\beta_R\,(1-\text{SUM}\,\rho_H))-\\
&\qquad d_{RR}\,(-1+\rho_H)\,\big(-\text{SUM}\,f_R\,\beta_R\,\rho_H^2+2\,\beta_{RR}\,(-1+\rho_H)\,(1+\text{SUM}\,\rho_H)\big)\big)\\
&\quad\big(d_R\,\rho_H\,\big(2\,\text{SUM}\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,(1+\text{SUM}-\text{SUM}\,\rho_H)\big)-\\
&\qquad d_{RR}\,(-1+\rho_H)^2\,(-\text{SUM}\,f_R\,\beta_R\,\rho_H+2\,\beta_{RR}\,(1-\text{SUM}+\text{SUM}\,\rho_H))\big)\Big)
\end{aligned}$$

$$\begin{aligned}
&-\Big(\Big(\delta_p\,(d_{RR}\,(-1+\rho_H)-d_R\,\rho_H)\\
&\quad\big(d_R\,\rho_H\,\big(-4\,\text{SUM}\,\beta_{RR}^2\,(-1+\rho_H)^3-2\,(1+\text{SUM})\,f_R\,\beta_R\,\beta_{RR}\,(-1+\rho_H)\,\rho_H+f_R^2\,\beta_R^2\,\rho_H^2\,(-1+\text{SUM}\,\rho_H)\big)+d_{RR}\,(-1+\\
&\qquad\rho_H)\,\big(2\,(1+\text{SUM})\,f_R\,\beta_R\,\beta_{RR}\,(-1+\rho_H)\,\rho_H-\text{SUM}\,f_R^2\,\beta_R^2\,\rho_H^3+4\,\beta_{RR}^2\,(-1+\rho_H)^2\,(1-\text{SUM}+\text{SUM}\,\rho_H)\big)\big)\Big)\Big/\\
&\quad\Big(\big(d_R\,\rho_H^2\,(-2\,\text{SUM}\,\beta_{RR}\,(-1+\rho_H)+f_R\,\beta_R\,(-1+\text{SUM}\,\rho_H))+d_{RR}\,(-1+\rho_H)\\
&\qquad\big(-\text{SUM}\,f_R\,\beta_R\,\rho_H^2+2\,\beta_{RR}\,(-1+\rho_H)\,(1+\text{SUM}\,\rho_H)\big)\big)\\
&\quad\big(-d_R\,\rho_H\,\big(2\,\text{SUM}\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,(1+\text{SUM}-\text{SUM}\,\rho_H)\big)+\\
&\qquad d_{RR}\,(-1+\rho_H)^2\,(-\text{SUM}\,f_R\,\beta_R\,\rho_H+2\,\beta_{RR}\,(1-\text{SUM}+\text{SUM}\,\rho_H))\big)\Big)\Big)
\end{aligned}$$

$$\frac{2\,d_{RR}\,\beta_{RR}\,(-1+\rho_H)^2-d_R\,f_R\,\beta_R\,\rho_H^2}{\text{SUM}\,(d_{RR}\,(-1+\rho_H)-d_R\,\rho_H)\,(2\,\beta_{RR}\,(-1+\rho_H)-f_R\,\beta_R\,\rho_H)}$$

$$\begin{aligned}
&-\Big(\Big(f_R\,\beta_R\,\rho_H\,(-d_{RR}\,(-1+\rho_H)+d_R\,\rho_H)\\
&\quad\Big(d_R^3\,\rho_H\,\big(\text{SUM}\,\delta_p^2\,(-1+\rho_H)\,(2\,\beta_{RR}\,(-1+\rho_H)-f_R\,\beta_R\,\rho_H)+F_D\,\big(p_M^2\,\big(2\,\text{SUM}\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\\
&\qquad(1+\text{SUM}-\text{SUM}\,\rho_H)\big)-p_M\,(1+2\,\delta_p)\,\big(2\,\text{SUM}\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,(1+\text{SUM}-\text{SUM}\,\rho_H)\big)+\\
&\qquad\delta_p\,\big(2\,\text{SUM}\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,(1+\text{SUM}+\delta_p-\text{SUM}\,\rho_H)\big)\big)\big)-\\
&\quad d_{RR}^3\,(-1+\rho_H)\,\big(\delta_p^2\,(1-\text{SUM}+\text{SUM}\,\rho_H)\,(2\,\beta_{RR}\,(-1+\rho_H)-f_R\,\beta_R\,\rho_H)+\\
&\qquad F_D\,\big(-\delta_p^2\,(1-\text{SUM}+\text{SUM}\,\rho_H)\,(2\,\beta_{RR}\,(-1+\rho_H)-f_R\,\beta_R\,\rho_H)-p_M\,(-1+\rho_H)\,(-\text{SUM}\,f_R\,\beta_R\,\rho_H+\\
&\qquad 2\,\beta_{RR}\,(1-\text{SUM}+\text{SUM}\,\rho_H))+p_M^2\,(-1+\rho_H)\,(-\text{SUM}\,f_R\,\beta_R\,\rho_H+2\,\beta_{RR}\,(1-\text{SUM}+\text{SUM}\,\rho_H))\big)\big)+\\
&\quad d_R\,d_{RR}^2\,\big(-\delta_p^2\,(-1+\rho_H)\,(2-2\,\text{SUM}+\text{SUM}\,\rho_H)\,(2\,\beta_{RR}\,(-1+\rho_H)-f_R\,\beta_R\,\rho_H)+\\
&\qquad F_D\,\big(p_M^2\,\big(-2\,\beta_{RR}\,(-1+\rho_H)^2\,(2-2\,\text{SUM}+\text{SUM}\,\rho_H)+f_R\,\beta_R\,\rho_H\,\big(2\,\text{SUM}+(1-3\,\text{SUM})\,\rho_H+\text{SUM}\,\rho_H^2\big)\big)+\\
&\qquad\delta_p\,(-1+\rho_H)\,(2\,\beta_{RR}\,(-1+\rho_H)\,(-1+\text{SUM}-\text{SUM}\,\rho_H+\delta_p\,(2-2\,\text{SUM}+\text{SUM}\,\rho_H))-\\
&\qquad f_R\,\beta_R\,\rho_H\,(\text{SUM}-\text{SUM}\,\rho_H+\delta_p\,(2-2\,\text{SUM}+\text{SUM}\,\rho_H)))+\\
&\qquad p_M\,\big(-f_R\,\beta_R\,\rho_H\,\big(2\,\text{SUM}+2\,\text{SUM}\,\delta_p\,(-1+\rho_H)^2+(1-3\,\text{SUM})\,\rho_H+\text{SUM}\,\rho_H^2\big)+\\
&\qquad 2\,\beta_{RR}\,(-1+\rho_H)^2\,(2-2\,\text{SUM}+\text{SUM}\,\rho_H+2\,\delta_p\,(1-\text{SUM}+\text{SUM}\,\rho_H))\big)\big)\big)+\\
&\quad d_R^2\,d_{RR}\,\big(\delta_p^2\,(-1+\rho_H)\,(-1+\text{SUM}+\text{SUM}\,\rho_H)\,(2\,\beta_{RR}\,(-1+\rho_H)-f_R\,\beta_R\,\rho_H)+\\
&\qquad F_D\,\big(p_M^2\,\big(2\,\beta_{RR}\,(-1+\rho_H)^2\,(-1+\text{SUM}+\text{SUM}\,\rho_H)+f_R\,\beta_R\,\rho_H\,\big(\text{SUM}+2\,\rho_H-\text{SUM}\,\rho_H^2\big)\big)+\\
&\qquad p_M\,\big(-2\,\beta_{RR}\,(-1+\rho_H)^2\,(-1+\text{SUM}+2\,(-1+\text{SUM})\,\delta_p+\text{SUM}\,\rho_H)+f_R\,\beta_R\,\rho_H\,\big(-\text{SUM}-2\,\rho_H+\text{SUM}\,\rho_H^2+\\
&\qquad 2\,\delta_p\,(-\text{SUM}+(-1+\text{SUM})\,\rho_H)\big)\big)+\delta_p\,\big(-2\,\beta_{RR}\,(-1+\rho_H)^2\,(1-\text{SUM}+2\,\text{SUM}\,\delta_p\,\rho_H)+\\
&\qquad f_R\,\beta_R\,\rho_H\,(\text{SUM}-(-1+\text{SUM})\,\rho_H+\delta_p\,(-1+\rho_H)\,(-1+2\,\text{SUM}\,\rho_H))\big)\big)\big)\Big)\Big)\Big/\\
&\Big(\Big(d_R^2\,\rho_H\,\big(-f_R\,\beta_R\,\delta_p\,\rho_H+p_M\,\big(2\,\text{SUM}\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,(1+\text{SUM}-\text{SUM}\,\rho_H)\big)\big)+\\
&\quad d_R\,d_{RR}\,\big(p_M\,\big(2\,(-1+\text{SUM})\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,(\text{SUM}-(-1+\text{SUM})\,\rho_H)\big)-\\
&\qquad\delta_p\,(-1+\rho_H)\,\rho_H\,(-2\,\text{SUM}\,\beta_{RR}\,(-1+\rho_H)+f_R\,\beta_R\,(-1+\text{SUM}\,\rho_H))\big)-\\
&\quad d_{RR}^2\,(-1+\rho_H)\,(\delta_p\,(1-\text{SUM}+\text{SUM}\,\rho_H)\,(2\,\beta_{RR}\,(-1+\rho_H)-f_R\,\beta_R\,\rho_H)+\\
&\qquad p_M\,(-1+\rho_H)\,(-\text{SUM}\,f_R\,\beta_R\,\rho_H+2\,\beta_{RR}\,(1-\text{SUM}+\text{SUM}\,\rho_H)))\Big)\\
&\quad\Big(d_R\,d_{RR}\,\big(2\,\beta_{RR}\,(-1+\rho_H)^2\,(1-\text{SUM}+(-1+\text{SUM})\,p_M+\text{SUM}\,\delta_p\,\rho_H)-\\
&\qquad f_R\,\beta_R\,\rho_H\,(\text{SUM}+\rho_H-\text{SUM}\,\rho_H+p_M\,(-\text{SUM}+(-1+\text{SUM})\,\rho_H)+\delta_p\,(-1+\rho_H)\,(-1+\text{SUM}\,\rho_H))\big)+\\
&\quad d_R^2\,\rho_H\,\big(2\,\text{SUM}\,(-1+p_M)\,\beta_{RR}\,(-1+\rho_H)^2-f_R\,\beta_R\,\rho_H\,(1+\text{SUM}+\delta_p-\text{SUM}\,\rho_H+p_M\,(-1-\text{SUM}+\text{SUM}\,\rho_H))\big)-\\
&\quad d_{RR}^2\,(-1+\rho_H)\,(2\,\beta_{RR}\,(-1+p_M+\delta_p)\,(-1+\rho_H)\,(1-\text{SUM}+\text{SUM}\,\rho_H)-\\
&\qquad f_R\,\beta_R\,\rho_H\,(\text{SUM}+\text{SUM}\,p_M\,(-1+\rho_H)-\text{SUM}\,\rho_H+\delta_p\,(1-\text{SUM}+\text{SUM}\,\rho_H)))\Big)\Big)
\end{aligned}$$

The demographic equilibrium reached in the absence of selection is:

```
RapidEqOrder0 = Solve[{EqC2a == 0, EqC2b == 0, EqC2c == 0, EqC2d == 0}, {pM, ρH, δp, FD}]
```

Solve::svars : 方程式はすべての"solve"変数に対しては解を与えない可能性があります. ≫

$$\left\{\left\{\rho_H \to \frac{-\sqrt{2}\sqrt{d_R}\sqrt{d_{RR}}\sqrt{f_R}\sqrt{\beta_R}\sqrt{\beta_{RR}} - 2\, d_{RR}\,\beta_{RR}}{d_R\, f_R\,\beta_R - 2\, d_{RR}\,\beta_{RR}},\ \delta_p \to 0,\ F_D \to 0\right\},\ \left\{\rho_H \to \frac{\sqrt{2}\sqrt{d_R}\sqrt{d_{RR}}\sqrt{f_R}\sqrt{\beta_R}\sqrt{\beta_{RR}} - 2\, d_{RR}\,\beta_{RR}}{d_R\, f_R\,\beta_R - 2\, d_{RR}\,\beta_{RR}},\ \delta_p \to 0,\ F_D \to 0\right\}\right\}$$

Of the two roots for ρ_H, the relevant one is positive and equals:

```
assumption = 0 < SUM && 0 < βR && 0 < βRR && 0 < dR && 0 < dRR && 0 < fR && fR < 1;
assumption2 = 0 < pM && pM < 1 && 0 < ρH && ρH < 1;

RapidEqOrder0ρH = Solve[{EqC2c == 0}, {ρH}]
(*check*)
check = ρH /. RapidEqOrder0ρH[[2]] // Simplify;
```

$$\text{Simplify}\left[\text{check} - \frac{\sqrt{\beta_{RR} * d_{RR}}}{\sqrt{\frac{f_R*\beta_R*d_R}{2}} + \sqrt{\beta_{RR} * d_{RR}}},\ \text{assumption}\right]$$

$$\left\{\left\{\rho_H \to \frac{-\sqrt{2}\sqrt{d_R}\sqrt{d_{RR}}\sqrt{f_R}\sqrt{\beta_R}\sqrt{\beta_{RR}} - 2\, d_{RR}\,\beta_{RR}}{d_R\, f_R\,\beta_R - 2\, d_{RR}\,\beta_{RR}}\right\},\ \left\{\rho_H \to \frac{\sqrt{2}\sqrt{d_R}\sqrt{d_{RR}}\sqrt{f_R}\sqrt{\beta_R}\sqrt{\beta_{RR}} - 2\, d_{RR}\,\beta_{RR}}{d_R\, f_R\,\beta_R - 2\, d_{RR}\,\beta_{RR}}\right\}\right\}$$

0

Confirming that this is a solution to the O(1) equations:

$$\{\text{EqC2a}, \text{EqC2b}, \text{EqC2c}, \text{EqC2d}\}\ /.\ \left\{\rho_H \to \frac{\sqrt{\beta_{RR} * d_{RR}}}{\sqrt{\frac{f_R*\beta_R*d_R}{2}} + \sqrt{\beta_{RR} * d_{RR}}},\ F_D \to 0,\ \delta_p \to 0\right\}\ //\ \text{Factor}$$

{0, 0, 0, 0}

Order(ϵ): δp, ρH-EqρH, FD = (ϵ)
We'll now set δp, FD, and (ρH-EqρH) to 0 plus order ϵ terms (e.g., $\delta p \to \epsilon\ \delta p1$) and see what the change in allele frequency equals to next order, ϵ^1:

Dynamics pM

```
eqpMTrans = EqC1a /. δp → (ϵ * δ̃p) /. ρH → (ρ̂H + ϵ * ρ̃H) /. FD → (ϵ * F̃D);
TSeries = Series[eqpMTrans, {ϵ, 0, 1}];
Order1pM = Normal[Collect[TSeries, {ϵ, pM, δ̃p, ρ̃H, F̃D}, Simplify]];
```

Dynamics δp

```
eqδpTrans = EqC1b /. δp → (ϵ * δ̃p) /. ρH → (ρ̂H + ϵ * ρ̃H) /. FD → (ϵ * F̃D);
TSeries = Series[eqδpTrans, {ϵ, 0, 1}];
Order1δp = Normal[Collect[TSeries, {ϵ, pM, δ̃p, ρ̃H, F̃D}, Simplify]];
```

Dynamics ρH

```
eqρHTrans = EqC1c /. δp → (ϵ * δ̃p) /. ρH → (ρ̂H + ϵ * ρ̃H) /. FD → (ϵ * F̃D);
TSeries = Series[eqρHTrans, {ϵ, 0, 1}];
Order1ρH = Normal[Collect[TSeries, {ϵ, pM, δ̃p, ρ̃H, F̃D}, Simplify]];
```

Dynamics FD

```
eqFDTrans = EqC1d /. δp → (ϵ * δ̃p) /. ρH → (ρ̂H + ϵ * ρ̃H) /. FD → (ϵ * F̃D);
TSeries = Series[eqFDTrans, {ϵ, 0, 1}];
Order1FD = Normal[Collect[TSeries, {ϵ, pM, δ̃p, ρ̃H, F̃D}, Simplify]];
```

Conclusion:
EqOrder1pM (pM)
EqOrder1δp (pM, δp)
EqOrder1ρH (pM, ρH)
EqOrder1FD (pM, FD)

```
EqOrder1pM = Simplify[Factor[Order1pM /. Flatten[Solve[ρ̂H == Sqrt[βRR * dRR] / Sqrt[(fR*βR*dR)/2 + Sqrt[βRR * dRR]], βR]]],
  assumption && assumption2]
EqOrder1δp = Simplify[Factor[Order1δp /. Flatten[Solve[ρ̂H == Sqrt[βRR * dRR] / Sqrt[(fR*βR*dR)/2 + Sqrt[βRR * dRR]], βR]]],
  assumption && assumption2]
EqOrder1ρH = Simplify[Factor[Order1ρH /. Flatten[Solve[ρ̂H == Sqrt[βRR * dRR] / Sqrt[(fR*βR*dR)/2 + Sqrt[βRR * dRR]], βR]]],
  assumption && assumption2]
EqOrder1FD = Simplify[Factor[Order1FD /. Flatten[Solve[ρ̂H == Sqrt[βRR * dRR] / Sqrt[(fR*βR*dR)/2 + Sqrt[βRR * dRR]], βR]]],
  assumption && assumption2]
```

$$-\left(\epsilon\, d_R\, d_{RR}\, (-1+p_M)\, p_M\, \left(2\,\tilde{s}_{dM}+2\,\tilde{s}_{dRM}+\tilde{s}_{fM}+2\,\tilde{s}_{\beta M}+2\,p_M\,(\tilde{s}_{dMM}-2\,\tilde{s}_{dRM}+\tilde{s}_{\beta MM}-2\,\tilde{s}_{\beta RM})+2\,\tilde{s}_{\beta RM}\right)\right)\Big/\left(2\,\mathrm{SUM}\,(d_R+d_{RR})\left(-d_{RR}\left(-1+\hat{\rho}_H\right)+d_R\,\hat{\rho}_H\right)\right)$$

$$\left(\epsilon\left(2\,d_R\left(p_M^2\,(-\tilde{s}_{dM}+\tilde{s}_{\beta MM}-3\,\tilde{s}_{\beta RM})-p_M^3\,(\tilde{s}_{\beta MM}-2\,\tilde{s}_{\beta RM})+p_M\,(\tilde{s}_{dM}+\tilde{s}_{\beta RM})+\tilde{\delta}_p\right)+d_{RR}\left(2\,p_M^3\,(\tilde{s}_{dMM}-2\,\tilde{s}_{dRM})-p_M\,(2\,\tilde{s}_{dRM}+\tilde{s}_{fM}+2\,\tilde{s}_{\beta M})+p_M^2\,(-2\,\tilde{s}_{dMM}+6\,\tilde{s}_{dRM}+\tilde{s}_{fM}+2\,\tilde{s}_{\beta M})+2\,\tilde{\delta}_p\right)\right)\right)\Big/\left(2\,\mathrm{SUM}\left(d_{RR}\left(-1+\hat{\rho}_H\right)-d_R\,\hat{\rho}_H\right)\right)$$

$$\left(\epsilon\, d_R\, d_{RR}\left(-p_M\left(-1+\hat{\rho}_H\right)\hat{\rho}_H\,(\tilde{s}_{dM}-2\,\tilde{s}_{dRM}-\tilde{s}_{fM}-\tilde{s}_{\beta M}+2\,\tilde{s}_{\beta RM})+p_M^2\left(-1+\hat{\rho}_H\right)\hat{\rho}_H\,(\tilde{s}_{dMM}-2\,\tilde{s}_{dRM}-\tilde{s}_{\beta MM}+2\,\tilde{s}_{\beta RM})-2\,\tilde{\rho}_H\right)\right)\Big/\left(\mathrm{SUM}\left(d_{RR}\left(-1+\hat{\rho}_H\right)-d_R\,\hat{\rho}_H\right)^2\right)$$

$$\frac{\epsilon\, d_{RR}\left(\tilde{F}_D+(-1+p_M)\,p_M\,(\tilde{s}_{dMM}-2\,\tilde{s}_{dRM})\right)}{\mathrm{SUM}\left(d_{RR}\left(-1+\hat{\rho}_H\right)-d_R\,\hat{\rho}_H\right)}$$

From the first of these equations, we can see that the allele frequency change (pM), assuming weak selection, is only a function of pM. The other equations describe how selection alters the other variables, but these are not needed to know how pM changes. We thus have a closed one-variable model that describes the dynamics of selection acting on pM to order ϵ^1.
The variables are separated!!

```
EqOrder1pM2 =
 EqOrder1pM /. ϵ → 1 /. s̃fM → sfM /. s̃βM → sβM /. s̃βRM → sβRM /. s̃βMM → sβMM /. s̃dM → sdM /. s̃dRM → sdRM /.
  s̃dMM → sdMM
```

$$-\left(d_R\, d_{RR}\, (-1+p_M)\, p_M\, \left(2\,s_{dM}+2\,s_{dRM}+s_{fM}+2\,s_{\beta M}+2\,p_M\,(s_{dMM}-2\,s_{dRM}+s_{\beta MM}-2\,s_{\beta RM})+2\,s_{\beta RM}\right)\right)\Big/\left(2\,\mathrm{SUM}\,(d_R+d_{RR})\left(-d_{RR}\left(-1+\hat{\rho}_H\right)+d_R\,\hat{\rho}_H\right)\right)$$

We check derivation of Eq. (C.3)

$$s_{average} = \frac{1}{2} * \left(\frac{s_{fM}}{2} + s_{\beta M} + s_{dM}\right) + \frac{1}{2} * (s_{\beta RM} + s_{dRM});$$

$$d_{harmonic} = \frac{2}{\frac{1}{d_R} + \frac{1}{d_{RR}}};$$

$$\hat{d} = d_R * \hat{\rho}_H + d_{RR} * \left(1 - \hat{\rho}_H\right);$$

$$EqC3 = \frac{p_M * (1 - p_M)}{SUM} * \frac{2 * s_{average} + p_M * (s_{\beta MM} - 2 * s_{\beta RM} + s_{dMM} - 2 * s_{dRM})}{2} * \frac{d_{harmonic}}{\hat{d}};$$

```
Simplify[EqOrder1pM2 - EqC3]
```

0

- **Effective selection coefficient and degree of dominance**

We check the derivation of Eq. (9) from Eq. (C3)

$$Eq3b = SUM^2 * \frac{\hat{d}}{d_{harmonic}} * EqC3;$$

$$P0 = \text{Limit}\left[\frac{Eq3b}{SUM * p_M * (1 - p_M)}, p_M \to 0\right] \text{ // Simplify};$$

$$P1 = \text{Limit}\left[\frac{Eq3b}{SUM * p_M * (1 - p_M)}, p_M \to 1\right] \text{ // Simplify};$$

$$s_e = P0 + P1;$$

$$h_e = \frac{P0}{s_e};$$

$$Eq9a = \left(2 * s_{average} + \frac{(1 - 2 * h_\beta) * s_{\beta MM} + (1 - 2 * h_d) * s_{dMM}}{2}\right) \text{ /. } h_\beta \to \frac{s_{\beta RM}}{s_{\beta MM}} \text{ /. } h_d \to \frac{s_{dRM}}{s_{dMM}};$$

$$Eq9b = \frac{2 * s_{average}}{4 * s_{average} + ((1 - 2 * h_\beta) * s_{\beta MM} + (1 - 2 * h_d) * s_{dMM})} \text{ /. } h_\beta \to \frac{s_{\beta RM}}{s_{\beta MM}} \text{ /. } h_d \to \frac{s_{dRM}}{s_{dMM}};$$

```
Simplify[se - Eq9a]
Simplify[he - Eq9b]
```

0

0

- **Applicable weight for average allele frequency for the separation of time scale**

We show that the weighting scheme, $\frac{dRR}{dR+dRR} = \frac{lR}{lR+lRR}$, is the only scheme that allows for a separation of time scales in this model.

Original Equations (Eqs.(A.7))

$$c_R = \beta_{RR} * x_{RR} + \frac{\beta_{RM}}{2} * x_{RM};$$

$$c_M = \frac{\beta_{RM}}{2} * x_{RM} + \beta_{MM} * x_{MM};$$

$$c_{RR} = \frac{f_R}{2} * \frac{\beta_R{}^2 * x_R{}^2}{\beta_R * x_R + \beta_M * x_M};$$

$$c_{RM} = \frac{f_R + f_M}{2} * \frac{\beta_R * \beta_M * x_R * x_M}{\beta_R * x_R + \beta_M * x_M};$$

$$c_{MM} = \frac{f_M}{2} * \frac{\beta_M{}^2 * x_M{}^2}{\beta_R * x_R + \beta_M * x_M};$$

$$\gamma = \frac{d_R * x_R + d_M * x_M + d_{RR} * x_{RR} + d_{RM} * x_{RM} + d_{MM} * x_{MM}}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}};$$

$$d_{mean} = \frac{d_R * x_R + d_M * x_M + d_{RR} * x_{RR} + d_{RM} * x_{RM} + d_{MM} * x_{MM}}{x_R + x_M + x_{RR} + x_{RM} + x_{MM}};$$

$$eqHR = \frac{\gamma * c_R - d_R * x_R}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$

$$eqHM = \frac{\gamma * c_M - d_M * x_M}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$

$$eqDRR = \frac{\gamma * c_{RR} - d_{RR} * x_{RR}}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$

$$eqDRM = \frac{\gamma * c_{RM} - d_{RM} * x_{RM}}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$

$$eqDMM = \frac{\gamma * c_{MM} - d_{MM} * x_{MM}}{(x_R + x_M + x_{RR} + x_{RM} + x_{MM}) * d_{mean}};$$

$HRnext = eqHR + x_R;$
$HMnext = eqHM + x_M;$
$DRRnext = eqDRR + x_{RR};$
$DRMnext = eqDRM + x_{RM};$
$DMMnext = eqDMM + x_{MM};$

Transform variables
(Define: pM, δp, ρH, FD)
(SUM=N in the manuscript)

For the average allele frequency, here we consider all possible weightings of the allele frequency in haploids and diploids, using the arbitrary weight “a”.

$$set = \{$$

$$SUM == x_R + x_M + x_{RR} + x_{RM} + x_{MM},$$

$$p_M == a * \frac{x_M}{x_R + x_M} + (1 - a) * \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR} + x_{RM} + x_{MM}},$$

$$\delta_p == \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR} + x_{RM} + x_{MM}} - \frac{x_M}{x_R + x_M},$$

$$\rho_H == \frac{x_R + x_M}{x_R + x_M + x_{RR} + x_{RM} + x_{MM}},$$

$$F_D == 1 - \frac{1}{2 * \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR} + x_{RM} + x_{MM}} * \left(1 - \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR} + x_{RM} + x_{MM}}\right)} * \frac{x_{RM}}{x_{RR} + x_{RM} + x_{MM}}$$

$$\};$$

OldToNew: Transformation of old variables to new variabls
NewToOld: Transformation of new variables to old variabls

```
OldToNew = Solve[set, {xR, xM, xRR, xRM, xMM}] // Flatten
NewToOld = Solve[set, {SUM, pM, ρH, δp, FD}] // Flatten
```

$$\{x_R \to -\text{SUM}\,(-1 + p_M - \delta_p + a\,\delta_p)\,\rho_H,\ x_M \to \text{SUM}\,(p_M - \delta_p + a\,\delta_p)\,\rho_H,$$
$$x_{RR} \to \text{SUM}\,(-1 + p_M + a\,\delta_p)\,(1 - p_M + F_D\,p_M - a\,\delta_p + a\,F_D\,\delta_p)\,(-1 + \rho_H),$$
$$x_{RM} \to -2\,\text{SUM}\,(-1 + F_D)\,\left(-p_M + p_M^2 - a\,\delta_p + 2\,a\,p_M\,\delta_p + a^2\,\delta_p^2\right)\,(-1 + \rho_H),$$
$$x_{MM} \to \text{SUM}\,(p_M + a\,\delta_p)\,(-F_D - p_M + F_D\,p_M - a\,\delta_p + a\,F_D\,\delta_p)\,(-1 + \rho_H)\}$$

$$\left\{\text{SUM} \to x_M + x_{MM} + x_R + x_{RM} + x_{RR},\ p_M \to \frac{a\,x_M}{x_M + x_R} + \frac{(1-a)\left(x_{MM} + \frac{x_{RM}}{2}\right)}{x_{MM} + x_{RM} + x_{RR}},\ \rho_H \to \frac{x_M + x_R}{x_M + x_{MM} + x_R + x_{RM} + x_{RR}},\right.$$
$$\left.\delta_p \to \frac{2\,x_{MM}\,x_R - x_M\,x_{RM} + x_R\,x_{RM} - 2\,x_M\,x_{RR}}{2\,(x_M + x_R)\,(x_{MM} + x_{RM} + x_{RR})},\ F_D \to \frac{-x_{RM}^2 + 4\,x_{MM}\,x_{RR}}{(2\,x_{MM} + x_{RM})\,(x_{RM} + 2\,x_{RR})}\right\}$$

Dynamics for the new variables

Dynamics pM

$$\text{EqpMOLD} = a * \frac{\text{HMnext}}{\text{HRnext} + \text{HMnext}} + (1 - a) * \frac{\frac{\text{DRMnext}}{2} + \text{DMMnext}}{\text{DRRnext} + \text{DRMnext} + \text{DMMnext}} - p_M;$$

```
EqpMNEW = EqpMOLD /. OldToNew;
```

Dynamics δp

$$\text{Eq}\delta\text{pOLD} = \left(\frac{\frac{\text{DRMnext}}{2} + \text{DMMnext}}{\text{DMMnext} + \text{DRMnext} + \text{DRRnext}} - \frac{\text{HMnext}}{\text{HMnext} + \text{HRnext}}\right) - \delta_p;$$

```
EqδpNEW = EqδpOLD /. OldToNew;
```

Dynamics ρH

$$\text{Eq}\rho\text{HOLD} = \left(\frac{\text{HMnext} + \text{HRnext}}{\text{SUM}}\right) - \rho_H;$$

```
EqρHNEW = EqρHOLD /. OldToNew;
```

Dynamics FD

$$\text{EqFDOLD} = \left(1 - \left(\frac{1}{2 * \frac{\frac{\text{DRMnext}}{2} + \text{DMMnext}}{\text{DRRnext}+\text{DRMnext}+\text{DMMnext}} * \left(1 - \frac{\frac{\text{DRMnext}}{2} + \text{DMMnext}}{\text{DRRnext}+\text{DRMnext}+\text{DMMnext}}\right)} * \frac{\text{DRMnext}}{\text{DRRnext} + \text{DRMnext} + \text{DMMnext}}\right)\right) - F_D;$$

```
EqFDNEW = EqFDOLD /. OldToNew;
```

We next transform original equations to new variables system under the weak selection

Asuming weak selection (Eqs. (C.1))

$$\text{weakselection} = \{\beta_{RM} \to \beta_{RR} * (1 + \epsilon * \tilde{s}_{\beta RM}),\ \beta_{MM} \to \beta_{RR} * (1 + \epsilon * \tilde{s}_{\beta MM}),\ \beta_M \to \beta_R * (1 + \epsilon * \tilde{s}_{\beta M}),$$
$$f_M \to f_R * (1 + \epsilon * \tilde{s}_{fM}),\ d_{RM} \to d_{RR} * (1 - \epsilon * \tilde{s}_{dRM}),\ d_{MM} \to d_{RR} * (1 - \epsilon * \tilde{s}_{dMM}),\ d_M \to d_R * (1 - \epsilon * \tilde{s}_{dM})\};$$

```
EqC1a = EqpMNEW /. weakselection;
EqC1b = EqδpNEW /. weakselection;
EqC1c = EqρHNEW /. weakselection;
EqC1d = EqFDNEW /. weakselection;
```

Eqs. C2
Order(1): ϵ=0

```
assumption = 0 < SUM && 0 < β_R && 0 < β_RR && 0 < d_R && 0 < d_RR && 0 < f_R && f_R < 1;
assumption2 = 0 < p_M && p_M < 1 && 0 < ρ_H && ρ_H < 1;

EqC2a = Simplify[EqC1a /. ϵ → 0, assumption && assumption2]
EqC2b = Simplify[EqC1b /. ϵ → 0, assumption && assumption2]
EqC2c = Simplify[EqC1c /. ϵ → 0, assumption && assumption2]
EqC2d = Simplify[EqC1d /. ϵ → 0, assumption && assumption2]
```

$$\begin{aligned}
&\Big(\delta_p\,(d_{RR}\,(-1+\rho_H)-d_R\,\rho_H)\\
&\quad\Big(d_R\,\rho_H\,\Big(-4\,a\,SUM\,\beta_{RR}^2\,(-1+\rho_H)^3-2\,f_R\,\beta_R\,\beta_{RR}\,(-1+\rho_H)\,\rho_H\,(a+a\,SUM-SUM\,\rho_H)+(-1+a)\\
&\qquad f_R^2\,\beta_R^2\,\rho_H^2\,(-1+SUM\,\rho_H)\Big)+d_{RR}\,(-1+\rho_H)\,\Big(-(-1+a)\,SUM\,f_R^2\,\beta_R^2\,\rho_H^3+\\
&\qquad 4\,a\,\beta_{RR}^2\,(-1+\rho_H)^2\,(1-SUM+SUM\,\rho_H)-2\,f_R\,\beta_R\,\beta_{RR}\,(-1+\rho_H)\,\rho_H\,(1-a\,(1+SUM)+SUM\,\rho_H)\Big)\Big)\Big)\Big/\\
&\Big(\Big(d_R\,\rho_H^2\,(-2\,SUM\,\beta_{RR}\,(-1+\rho_H)+f_R\,\beta_R\,(-1+SUM\,\rho_H))+d_{RR}\,(-1+\rho_H)\\
&\qquad\Big(-SUM\,f_R\,\beta_R\,\rho_H^2+2\,\beta_{RR}\,(-1+\rho_H)\,(1+SUM\,\rho_H)\Big)\Big)\\
&\quad\Big(-d_R\,\rho_H\,\Big(2\,SUM\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,(1+SUM-SUM\,\rho_H)\Big)+\\
&\qquad d_{RR}\,(-1+\rho_H)^2\,(-SUM\,f_R\,\beta_R\,\rho_H+2\,\beta_{RR}\,(1-SUM+SUM\,\rho_H))\Big)\Big)
\end{aligned}$$

$$\begin{aligned}
&-\Big(\Big(\delta_p\,(d_{RR}\,(-1+\rho_H)-d_R\,\rho_H)\\
&\quad\Big(d_R\,\rho_H\,\Big(-4\,SUM\,\beta_{RR}^2\,(-1+\rho_H)^3-2\,(1+SUM)\,f_R\,\beta_R\,\beta_{RR}\,(-1+\rho_H)\,\rho_H+f_R^2\,\beta_R^2\,\rho_H^2\,(-1+SUM\,\rho_H)\Big)+d_{RR}\,(-1+\\
&\qquad\rho_H)\,\Big(2\,(1+SUM)\,f_R\,\beta_R\,\beta_{RR}\,(-1+\rho_H)\,\rho_H-SUM\,f_R^2\,\beta_R^2\,\rho_H^3+4\,\beta_{RR}^2\,(-1+\rho_H)^2\,(1-SUM+SUM\,\rho_H)\Big)\Big)\Big)\Big/\\
&\quad\Big(\Big(d_R\,\rho_H^2\,(-2\,SUM\,\beta_{RR}\,(-1+\rho_H)+f_R\,\beta_R\,(-1+SUM\,\rho_H))+d_{RR}\,(-1+\rho_H)\\
&\qquad\Big(-SUM\,f_R\,\beta_R\,\rho_H^2+2\,\beta_{RR}\,(-1+\rho_H)\,(1+SUM\,\rho_H)\Big)\Big)\\
&\quad\Big(-d_R\,\rho_H\,\Big(2\,SUM\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,(1+SUM-SUM\,\rho_H)\Big)+\\
&\qquad d_{RR}\,(-1+\rho_H)^2\,(-SUM\,f_R\,\beta_R\,\rho_H+2\,\beta_{RR}\,(1-SUM+SUM\,\rho_H))\Big)\Big)\Big)
\end{aligned}$$

$$\frac{2\,d_{RR}\,\beta_{RR}\,(-1+\rho_H)^2-d_R\,f_R\,\beta_R\,\rho_H^2}{SUM\,(d_{RR}\,(-1+\rho_H)-d_R\,\rho_H)\,(2\,\beta_{RR}\,(-1+\rho_H)-f_R\,\beta_R\,\rho_H)}$$

$$\begin{aligned}
&-\Big(\Big(f_R\,\beta_R\,\rho_H\,(d_{RR}\,(-1+\rho_H)-d_R\,\rho_H)\\
&\quad\Big(d_{RR}\,(-1+\rho_H)\,\Big(\delta_p^2\,(1-SUM+SUM\,\rho_H)\,(2\,\beta_{RR}\,(-1+\rho_H)-f_R\,\beta_R\,\rho_H)+F_D\,\Big(p_M^2\,(-1+\rho_H)\,(-SUM\,f_R\,\beta_R\,\rho_H+\\
&\qquad 2\,\beta_{RR}\,(1-SUM+SUM\,\rho_H))+p_M\,(-1+2\,(-1+a)\,\delta_p)\,(-1+\rho_H)\,(-SUM\,f_R\,\beta_R\,\rho_H+2\,\beta_{RR}\\
&\qquad (1-SUM+SUM\,\rho_H))+\delta_p\,\Big(2\,\beta_{RR}\,(1-a+(-2+a)\,a\,\delta_p)\,(-1+\rho_H)\,(1-SUM+SUM\,\rho_H)+\\
&\qquad f_R\,\beta_R\,\rho_H\,\Big((-1+a)\,SUM\,(-1+\rho_H)+\delta_p\,\Big(1-2\,a\,SUM+a^2\,SUM-(-2+a)\,a\,SUM\,\rho_H\Big)\Big)\Big)\Big)\Big)-\\
&\quad d_R\,\rho_H\,\Big(SUM\,\delta_p^2\,(-1+\rho_H)\,(2\,\beta_{RR}\,(-1+\rho_H)-f_R\,\beta_R\,\rho_H)+F_D\\
&\qquad\Big(p_M^2\,\Big(2\,SUM\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,(1+SUM-SUM\,\rho_H)\Big)+\\
&\qquad p_M\,(-1+2\,(-1+a)\,\delta_p)\,\Big(2\,SUM\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,(1+SUM-SUM\,\rho_H)\Big)+\\
&\qquad\delta_p\,\Big(2\,SUM\,\beta_{RR}\,(1-a+(-2+a)\,a\,\delta_p)\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,\Big((-1+a)\,(-1-SUM+SUM\,\rho_H)+\\
&\qquad\delta_p\,\Big(1-2\,a\,(1+SUM)+a^2\,(1+SUM)-(-2+a)\,a\,SUM\,\rho_H\Big)\Big)\Big)\Big)\Big)\Big)\Big)\Big/\\
&\quad\Big(\Big(d_{RR}\,(-1+\rho_H)\,(p_M\,(-1+\rho_H)\,(-SUM\,f_R\,\beta_R\,\rho_H+2\,\beta_{RR}\,(1-SUM+SUM\,\rho_H))+\\
&\qquad\delta_p\,(2\,a\,\beta_{RR}\,(-1+\rho_H)\,(1-SUM+SUM\,\rho_H)+f_R\,\beta_R\,\rho_H\,(-1+a\,SUM-a\,SUM\,\rho_H)))-\\
&\qquad d_R\,\rho_H\,\Big(p_M\,\Big(2\,SUM\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,(1+SUM-SUM\,\rho_H)\Big)+\\
&\qquad\delta_p\,\Big(2\,a\,SUM\,\beta_{RR}\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\,(-1+a+a\,SUM-a\,SUM\,\rho_H)\Big)\Big)\Big)\\
&\quad\Big(d_{RR}\,(-1+\rho_H)\,(2\,\beta_{RR}\,(-1+p_M+a\,\delta_p)\,(-1+\rho_H)\,(1-SUM+SUM\,\rho_H)-\\
&\qquad f_R\,\beta_R\,\rho_H\,(SUM+SUM\,p_M\,(-1+\rho_H)-SUM\,\rho_H+\delta_p\,(1-a\,SUM+a\,SUM\,\rho_H)))+\\
&\qquad d_R\,\rho_H\,\Big(-2\,SUM\,\beta_{RR}\,(-1+p_M+a\,\delta_p)\,(-1+\rho_H)^2+f_R\,\beta_R\,\rho_H\\
&\qquad (1+SUM-SUM\,\rho_H+p_M\,(-1-SUM+SUM\,\rho_H)+\delta_p\,(1-a\,(1+SUM)+a\,SUM\,\rho_H))\Big)\Big)\Big)\Big)
\end{aligned}$$

Order(ϵ): δp, ρH-EqρH, FD = (ϵ)
We'll now set δp, FD, and (ρH-EqρH) to 0 plus order ϵ terms (e.g., $\delta p \to \epsilon\, \delta p1$) and see what the change in allele frequency equals to next order, ϵ^1:

Dynamics pM

```
eqpMTrans = EqC1a /. δ_p → (ϵ * δ̃_p) /. ρ_H → (ρ̂_H + ϵ * ρ̃_H) /. F_D → (ϵ * F̃_D);
TSeries = Series[eqpMTrans, {ϵ, 0, 1}];
Order1pM = Normal[Collect[TSeries, {ϵ, p_M, δ̃_p, ρ̃_H, F̃_D}, Simplify]];
```

```
EqOrder1pM = Simplify[Factor[Order1pM /. Flatten[Solve[ρ̂H == Sqrt[βRR * dRR]/(Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]), βR]]],
  assumption && assumption2]
```

$$\left(\epsilon\left(2\,a\,d_R\left(p_M^3(\tilde{s}_{\beta MM}-2\tilde{s}_{\beta RM})-p_M(\tilde{s}_{dM}+\tilde{s}_{\beta RM})+p_M^2(\tilde{s}_{dM}-\tilde{s}_{\beta MM}+3\tilde{s}_{\beta RM})-\tilde{\delta}_p\right)-(-1+a)\,d_{RR}\left(2p_M^3(\tilde{s}_{dMM}-2\tilde{s}_{dRM})-p_M(2\tilde{s}_{dRM}+\tilde{s}_{fM}+2\tilde{s}_{\beta M})+p_M^2(-2\tilde{s}_{dMM}+6\tilde{s}_{dRM}+\tilde{s}_{fM}+2\tilde{s}_{\beta M})+2\tilde{\delta}_p\right)\right)\right)\Big/\left(2\,\mathrm{SUM}\left(d_{RR}(-1+\hat{\rho}_H)-d_R\hat{\rho}_H\right)\right)$$

Finally, we ask what the weighting term "a" must equal for the dynamics of the allele frequency to be independent from $\tilde{\delta}_p$:

```
Solve[D[EqOrder1pM, δ̃p] == 0, a]
```

$$\left\{\left\{a\to\frac{d_{RR}}{d_R+d_{RR}}\right\}\right\}$$

Recalling that the death rate is one over the longevity, this demonstrates that the allele frequency in haploids weighted by the average haploid longevity ($\frac{lR}{lR+lRR}$) plus the allele frequency in diploids times the average diploid longevity ($\frac{lRR}{lR+lRR}$) allows for a separation of time scales where the allele frequency dynamics can be separated from the dynamics of the remaining variables to this order.

- **Class reproductive value**

We show that the weighting scheme, $\frac{dRR}{dR+dRR}=\frac{lR}{lR+lRR}$, is similar as the class reproductive value of the Moran model.
Because the population dynamics of residents in the Moran model can be described as,

$$H' = H-\frac{dR*H}{dR*H+dRR*D}+\frac{\beta_{RR}*D}{\frac{fR*\beta_R}{2}*H+\beta_{RR}*D}=\left(1-\frac{dR}{dR*H+dRR*D}\right)H+\frac{\beta_{RR}}{\frac{fR*\beta_R}{2}*H+\beta_{RR}*D}D$$

$$D' = D-\frac{dRR*D}{dR*H+dRR*D}+\frac{\frac{fR*\beta_R}{2}*H}{\frac{fR*\beta_R}{2}*H+\beta_{RR}*D}=\frac{\frac{fR*\beta_R}{2}}{\frac{fR*\beta_R}{2}*H+\beta_{RR}*D}H+\left(1-\frac{dRR}{dR*H+dRR*D}\right)D$$

This dynamics of H/N and D/N can be described by matrix,

```
assumption = 0 < ρH && ρH < 1 && 0 < dR && 0 < dRR && 0 < βR && 0 < βRR && 0 < fR && fR < 1 && 0 < SUM;
```

$$W=\begin{pmatrix}1-\frac{d_R}{SUM*(d_R*hap+d_{RR}*dip)} & \frac{\beta_{RR}}{SUM*\left(\frac{f_R*\beta_R}{2}*hap+\beta_{RR}*dip\right)}\\ \frac{\frac{f_R*\beta_R}{2}}{SUM*\left(\frac{f_R*\beta_R}{2}*hap+\beta_{RR}*dip\right)} & 1-\frac{d_{RR}}{SUM*(d_R*hap+d_{RR}*dip)}\end{pmatrix}\ /.\ dip\to 1-hap;$$

At population equilibrium, the eigenvalue of matrix W should be one.

```
LambdaW = Eigenvalues[W];

Simplify[LambdaW /. hap → ρH /. Flatten[Solve[ρH == Sqrt[βRR * dRR]/(Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]), βR]], assumption]

right = Eigenvectors[W];

Simplify[right /. hap → ρH /. Flatten[Solve[ρH == Sqrt[βRR * dRR]/(Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]), βR]],
  assumption]
```

$$\left\{1,\ \frac{d_R(1-\mathrm{SUM}\,\rho_H)+d_{RR}(1-\mathrm{SUM}+\mathrm{SUM}\,\rho_H)}{\mathrm{SUM}\,(d_{RR}(-1+\rho_H)-d_R\,\rho_H)}\right\}$$

$$\left\{\left\{\frac{\rho_H}{1-\rho_H},1\right\},\left\{\frac{d_R\,\rho_H}{d_{RR}(-1+\rho_H)},1\right\}\right\}$$

The reproductive value of individual in the haploid-diploid model can be derived as the left eigenvector of the matrix W.

```
TransW = Eigenvalues[Transpose[W]];
```

$$\text{Simplify}\left[\text{TransW} \mathbin{/.} \text{hap} \to \rho_H \mathbin{/.} \text{Flatten}\left[\text{Solve}\left[\rho_H == \frac{\sqrt{\beta_{RR} * d_{RR}}}{\sqrt{\frac{f_R*\beta_R*d_R}{2}} + \sqrt{\beta_{RR} * d_{RR}}}, \beta_R\right]\right], \text{assumption}\right]$$

```
left = Eigenvectors[Transpose[W]];
```

$$\text{Simplify}\left[\text{left} \mathbin{/.} \text{hap} \to \rho_H \mathbin{/.} \text{Flatten}\left[\text{Solve}\left[\rho_H == \frac{\sqrt{\beta_{RR} * d_{RR}}}{\sqrt{\frac{f_R*\beta_R*d_R}{2}} + \sqrt{\beta_{RR} * d_{RR}}}, \beta_R\right]\right], \text{assumption}\right]$$

$$\left\{\frac{d_R\ (1 - \text{SUM}\ \rho_H) + d_{RR}\ (1 - \text{SUM} + \text{SUM}\ \rho_H)}{\text{SUM}\ (d_{RR}\ (-1 + \rho_H) - d_R\ \rho_H)}, 1\right\}$$

$$\left\{\left\{1 - \frac{1}{\rho_H}, 1\right\}, \left\{-\frac{d_{RR}\ (-1 + \rho_H)}{d_R\ \rho_H}, 1\right\}\right\}$$

We weight individual reproductive value by the population size, we have class reproductive value, c.

$$v = \text{Simplify}\left[\text{left}[[2]] \mathbin{/.} \text{hap} \to \rho_H \mathbin{/.} \text{Flatten}\left[\text{Solve}\left[\rho_H == \frac{\sqrt{\beta_{RR} * d_{RR}}}{\sqrt{\frac{f_R*\beta_R*d_R}{2}} + \sqrt{\beta_{RR} * d_{RR}}}, \beta_R\right]\right], \text{assumption}\right]$$

$$c = \text{Simplify}[\{v[[1]] * \rho_H, v[[2]] * \rho_D\} \mathbin{/.} \rho_H \to 1 - \rho_D, \text{assumption}]$$

$$\left\{-\frac{d_{RR}\ (-1 + \rho_H)}{d_R\ \rho_H}, 1\right\}$$

$$\left\{\frac{d_{RR}\ \rho_D}{d_R}, \rho_D\right\}$$

This is the weight to calculate the change in frequency of an allele under the selection, “class reproductive values” in Taylor (1990). And this weight certainly realizes one variable system by separation of time scale (Appendix B).

$$Nc = \text{Simplify}\left[\left\{\frac{c[[1]]}{c[[1]] + c[[2]]}, \frac{c[[2]]}{c[[1]] + c[[2]]}\right\}, \text{assumption}\right]$$

$$\left\{\frac{d_{RR}}{d_R + d_{RR}}, \frac{d_R}{d_R + d_{RR}}\right\}$$

- **Diffusion Approximation (Second and third Moment)**

- **The change in moments for pM**

We next derive the second moment of change in mutant allele frequency

Explicitly tracking the genotypes of the deaths, deaths occur in proportion to:

$$d_{tot} = d_R * x_R + d_M * x_M + d_{RR} * x_{RR} + d_{RM} * x_{RM} + d_{MM} * x_{MM};$$

$$\tilde{d}_R = \frac{d_R * x_R}{d_{tot}};$$

$$\tilde{d}_M = \frac{d_M * x_M}{d_{tot}};$$

$$\tilde{d}_{RR} = \frac{d_{RR} * x_{RR}}{d_{tot}};$$

$$\tilde{d}_{RM} = \frac{d_{RM} * x_{RM}}{d_{tot}};$$

$$\tilde{d}_{MM} = \frac{d_{MM} * x_{MM}}{d_{tot}};$$

Births then occur in proportion to c(GT)/totC.

$$c_R = \beta_{RR} * x_{RR} + \frac{\beta_{RM}}{2} * x_{RM};$$

$$c_M = \frac{\beta_{RM}}{2} * x_{RM} + \beta_{MM} * x_{MM};$$

$$c_{RR} = \frac{f_R}{2} * \frac{\beta_R{}^2 * x_R{}^2}{\beta_R * x_R + \beta_M * x_M};$$

$$c_{RM} = \frac{f_R + f_M}{2} * \frac{\beta_R * \beta_M * x_R * x_M}{\beta_R * x_R + \beta_M * x_M};$$

$$c_{MM} = \frac{f_M}{2} * \frac{\beta_M{}^2 * x_M{}^2}{\beta_R * x_R + \beta_M * x_M};$$

$$c_{tot} = (c_R + c_M + c_{RR} + c_{RM} + c_{MM});$$

$$\tilde{p}_R = \frac{c_R}{c_{tot}};$$

$$\tilde{p}_M = \frac{c_M}{c_{tot}};$$

$$\tilde{p}_{RR} = \frac{c_{RR}}{c_{tot}};$$

$$\tilde{p}_{RM} = \frac{c_{RM}}{c_{tot}};$$

$$\tilde{p}_{MM} = \frac{c_{MM}}{c_{tot}};$$

Transform variables
(Define: pM, δp, ρH, FD)
(SUM=N in the manuscript)

$$\text{set} = \Big\{$$

$$\text{SUM} == x_R + x_M + x_{RR} + x_{RM} + x_{MM},$$

$$p_M == \frac{d_{RR}}{d_R + d_{RR}} * \frac{x_M}{x_R + x_M} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR} + x_{RM} + x_{MM}},$$

$$\delta_p == \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR} + x_{RM} + x_{MM}} - \frac{x_M}{x_R + x_M},$$

$$\rho_H == \frac{x_R + x_M}{x_R + x_M + x_{RR} + x_{RM} + x_{MM}},$$

$$F_D == 1 - \frac{1}{2 * \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR}+x_{RM}+x_{MM}} * \left(1 - \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR}+x_{RM}+x_{MM}}\right)} * \frac{x_{RM}}{x_{RR} + x_{RM} + x_{MM}}$$

$$\Big\};$$

OldToNew: Transformation of old variables to new variabls
NewToOld: Transformation of new variables to old variabls

```
OldToNew = Solve[set, {xR, xM, xRR, xRM, xMM}] // Flatten
NewToOld = Solve[set, {SUM, pM, ρH, δp, FD}] // Flatten
```

$$\left\{x_R \to -\frac{SUM\,(-d_R - d_{RR} + d_R\,p_M + d_{RR}\,p_M - d_R\,\delta_p)\,\rho_H}{d_R + d_{RR}},\right.$$

$$x_M \to \frac{SUM\,(d_R\,p_M + d_{RR}\,p_M - d_R\,\delta_p)\,\rho_H}{d_R + d_{RR}},\ x_{RR} \to \frac{1}{(d_R + d_{RR})^2}\,SUM\,(-d_R - d_{RR} + d_R\,p_M + d_{RR}\,p_M + d_{RR}\,\delta_p)$$
$$(d_R + d_{RR} - d_R\,p_M - d_{RR}\,p_M + d_R\,F_D\,p_M + d_{RR}\,F_D\,p_M - d_{RR}\,\delta_p + d_{RR}\,F_D\,\delta_p)\,(-1 + \rho_H),$$

$$x_{RM} \to -\frac{1}{(d_R + d_{RR})^2}\,2\,SUM\,(-1 + F_D)\,\left(-d_R^2\,p_M - 2\,d_R\,d_{RR}\,p_M - d_{RR}^2\,p_M + d_R^2\,p_M^2 + 2\,d_R\,d_{RR}\,p_M^2 + d_{RR}^2\,p_M^2 - d_R\,d_{RR}\,\delta_p -\right.$$
$$\left.d_{RR}^2\,\delta_p + 2\,d_R\,d_{RR}\,p_M\,\delta_p + 2\,d_{RR}^2\,p_M\,\delta_p + d_{RR}^2\,\delta_p^2\right)\,(-1 + \rho_H),\ x_{MM} \to \frac{1}{(d_R + d_{RR})^2}\,SUM\,(d_R\,p_M + d_{RR}\,p_M + d_{RR}\,\delta_p)$$
$$\left.(-d_R\,F_D - d_{RR}\,F_D - d_R\,p_M - d_{RR}\,p_M + d_R\,F_D\,p_M + d_{RR}\,F_D\,p_M - d_{RR}\,\delta_p + d_{RR}\,F_D\,\delta_p)\,(-1 + \rho_H)\right\}$$

$$\left\{SUM \to x_M + x_{MM} + x_R + x_{RM} + x_{RR},\ p_M \to \frac{d_{RR}\,x_M}{(d_R + d_{RR})\,(x_M + x_R)} + \frac{d_R\left(x_{MM} + \frac{x_{RM}}{2}\right)}{(d_R + d_{RR})\,(x_{MM} + x_{RM} + x_{RR})},\right.$$

$$\left.\rho_H \to \frac{x_M + x_R}{x_M + x_{MM} + x_R + x_{RM} + x_{RR}},\ \delta_p \to \frac{2\,x_{MM}\,x_R - x_M\,x_{RM} + x_R\,x_{RM} - 2\,x_M\,x_{RR}}{2\,(x_M + x_R)\,(x_{MM} + x_{RM} + x_{RR})},\ F_D \to \frac{-x_{RM}^2 + 4\,x_{MM}\,x_{RR}}{(2\,x_{MM} + x_{RM})\,(x_{RM} + 2\,x_{RR})}\right\}$$

Here we focus only on changes in allele frequency, $pM == \frac{dR}{dR+dRR}\frac{\frac{DRM}{2}+DMM}{DRR+DRM+DMM} + \frac{dRR}{dR+dRR}\frac{HM}{HR+HM}$, across a time step, ignoring whether the allele is carried by a haploid or diploid (i.e., assuming that the allele frequencies will remain near each other) and ignoring the slight fluctuations that would occur around the equilibrium value of ρH.

In the following, we keep track of changes in the allele frequency, pM:

A single birth-death event could cause one of the following 20 changes to the kth moment of pM:

```
moment[k_] =
 (d̃R * p̃M *
   (dRR/(dR + dRR) * (xM + 1)/((xR - 1) + (xM + 1)) + dR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) - pM /. OldToNew /. δp → 0 /. FD → 0 /.
      ρH → ρ̂H)^k (* R dies, M born *)
  +
  d̃R * p̃RR *
   (dRR/(dR + dRR) * xM/((xR - 1) + xM) + dR/(dR + dRR) * (xRM/2 + xMM)/((xRR + 1) + xRM + xMM) - pM /. OldToNew /. δp → 0 /. FD → 0 /.
      ρH → ρ̂H)^k
   (* R dies, RR born*) +
  d̃R * p̃RM *
   (dRR/(dR + dRR) * xM/((xR - 1) + xM) + dR/(dR + dRR) * ((xRM+1)/2 + xMM)/(xRR + (xRM + 1) + xMM) - pM /. OldToNew /. δp → 0 /. FD → 0 /.
      ρH → ρ̂H)^k (* R dies, RM born *) +
  d̃R * p̃MM *
```

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{x_M}{(x_R - 1) + x_M} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{x_{RM}}{2} + (x_{MM} + 1)}{x_{RR} + x_{RM} + (x_{MM} + 1)} - p_M \;/.\; \text{OldToNew} \;/.\; \delta_p \to 0 \;/.\; F_D \to 0 \;/.\; \rho_H \to \hat{\rho}_H\right)^k$$

(* R dies, MM born *) +

$\tilde{d}_M * \tilde{p}_R *$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{(x_M - 1)}{(x_R + 1) + (x_M - 1)} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{x_{RM}}{2} + x_{MM}}{x_{RR} + x_{RM} + x_{MM}} - p_M \;/.\; \text{OldToNew} \;/.\; \delta_p \to 0 \;/.\; F_D \to 0 \;/.\; \rho_H \to \hat{\rho}_H\right)^k$$

(* M dies, R born *) +

$\tilde{d}_M * \tilde{p}_{RR} *$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{(x_M - 1)}{x_R + (x_M - 1)} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{x_{RM}}{2} + x_{MM}}{(x_{RR} + 1) + x_{RM} + x_{MM}} - p_M \;/.\; \text{OldToNew} \;/.\; \delta_p \to 0 \;/.\; F_D \to 0 \;/.\; \rho_H \to \hat{\rho}_H\right)^k$$

(* M dies, RR born *) +

$\tilde{d}_M * \tilde{p}_{RM} *$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{(x_M - 1)}{x_R + (x_M - 1)} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{(x_{RM}+1)}{2} + x_{MM}}{x_{RR} + (x_{RM} + 1) + x_{MM}} - p_M \;/.\; \text{OldToNew} \;/.\; \delta_p \to 0 \;/.\; F_D \to 0 \;/.\; \rho_H \to \hat{\rho}_H\right)^k$$

(* M dies, RM born *) +

$\tilde{d}_M * \tilde{p}_{MM} *$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{(x_M - 1)}{x_R + (x_M - 1)} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{x_{RM}}{2} + (x_{MM} + 1)}{x_{RR} + x_{RM} + (x_{MM} + 1)} - p_M \;/.\; \text{OldToNew} \;/.\; \delta_p \to 0 \;/.\; F_D \to 0 \;/.\; \rho_H \to \hat{\rho}_H\right)^k$$

(* M dies, MM born*) +

$\tilde{d}_{RR} * \tilde{p}_R *$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{x_M}{(x_R + 1) + x_M} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{x_{RM}}{2} + x_{MM}}{(x_{RR} - 1) + x_{RM} + x_{MM}} - p_M \;/.\; \text{OldToNew} \;/.\; \delta_p \to 0 \;/.\; F_D \to 0 \;/.\; \rho_H \to \hat{\rho}_H\right)^k$$

(* RR dies, R born*) +

$\tilde{d}_{RR} * \tilde{p}_M *$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{(x_M + 1)}{x_R + (x_M + 1)} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{x_{RM}}{2} + x_{MM}}{(x_{RR} - 1) + x_{RM} + x_{MM}} - p_M \;/.\; \text{OldToNew} \;/.\; \delta_p \to 0 \;/.\; F_D \to 0 \;/.\; \rho_H \to \hat{\rho}_H\right)^k$$

(* RR dies, M born *) +

$\tilde{d}_{RR} * \tilde{p}_{RM} *$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{x_M}{x_R + x_M} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{(x_{RM}+1)}{2} + x_{MM}}{(x_{RR} - 1) + (x_{RM} + 1) + x_{MM}} - p_M \text{ /. OldToNew /. } \delta_p \to 0 \text{ /. } F_D \to 0 \text{ /. } \rho_H \to \hat{\rho}_H\right)^k$$

(* RR dies, RM born *) +

$$\tilde{d}_{RR} * \tilde{p}_{MM} *$$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{x_M}{x_R + x_M} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{x_{RM}}{2} + (x_{MM} + 1)}{(x_{RR} - 1) + x_{RM} + (x_{MM} + 1)} - p_M \text{ /. OldToNew /. } \delta_p \to 0 \text{ /. } F_D \to 0 \text{ /. } \rho_H \to \hat{\rho}_H\right)^k$$

(* RR dies, MM born *) +

$$\tilde{d}_{RM} * \tilde{p}_R *$$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{x_M}{(x_R + 1) + x_M} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{(x_{RM}-1)}{2} + x_{MM}}{x_{RR} + (x_{RM} - 1) + x_{MM}} - p_M \text{ /. OldToNew /. } \delta_p \to 0 \text{ /. } F_D \to 0 \text{ /. } \rho_H \to \hat{\rho}_H\right)^k$$

(* RM dies, R born *) +

$$\tilde{d}_{RM} * \tilde{p}_M *$$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{(x_M + 1)}{x_R + (x_M + 1)} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{(x_{RM}-1)}{2} + x_{MM}}{x_{RR} + (x_{RM} - 1) + x_{MM}} - p_M \text{ /. OldToNew /. } \delta_p \to 0 \text{ /. } F_D \to 0 \text{ /. } \rho_H \to \hat{\rho}_H\right)^k$$

(* RM dies, M born *) +

$$\tilde{d}_{RM} * \tilde{p}_{RR} *$$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{x_M}{x_R + x_M} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{(x_{RM}-1)}{2} + x_{MM}}{(x_{RR} + 1) + (x_{RM} - 1) + x_{MM}} - p_M \text{ /. OldToNew /. } \delta_p \to 0 \text{ /. } F_D \to 0 \text{ /. } \rho_H \to \hat{\rho}_H\right)^k$$

(* RM dies, RR born *) +

$$\tilde{d}_{RM} * \tilde{p}_{MM} *$$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{x_M}{x_R + x_M} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{(x_{RM}-1)}{2} + (x_{MM} + 1)}{x_{RR} + (x_{RM} - 1) + (x_{MM} + 1)} - p_M \text{ /. OldToNew /. } \delta_p \to 0 \text{ /. } F_D \to 0 \text{ /. } \rho_H \to \hat{\rho}_H\right)^k$$

(* RM dies, MM born *)

+

$$\tilde{d}_{MM} * \tilde{p}_R *$$

$$\left(\frac{d_{RR}}{d_R + d_{RR}} * \frac{x_M}{(x_R + 1) + x_M} + \frac{d_R}{d_R + d_{RR}} * \frac{\frac{x_{RM}}{2} + (x_{MM} - 1)}{x_{RR} + x_{RM} + (x_{MM} - 1)} - p_M \text{ /. OldToNew /. } \delta_p \to 0 \text{ /. } F_D \to 0 \text{ /. } \rho_H \to \hat{\rho}_H\right)^k$$

(* MM dies, R born *)

+

$$\tilde{d}_{MM} * \tilde{p}_M *$$

```
(dRR/(dR + dRR) * (xM + 1)/(xR + (xM + 1)) + dR/(dR + dRR) * (xRM/2 + (xMM - 1))/(xRR + xRM + (xMM - 1)) - pM /. OldToNew /. δp → 0 /. FD → 0 /.
    ρH → ρ̂H)^k
  (* MM dies, M born *) +
  d̃MM * p̃RR *
   (dRR/(dR + dRR) * xM/(xR + xM) + dR/(dR + dRR) * (xRM/2 + (xMM - 1))/((xRR + 1) + xRM + (xMM - 1)) - pM /. OldToNew /. δp → 0 /. FD → 0 /.
    ρH → ρ̂H)^k (* MM dies, RR born *) +
  d̃MM * p̃RM *
   (dRR/(dR + dRR) * xM/(xR + xM) + dR/(dR + dRR) * ((xRM+1)/2 + (xMM - 1))/(xRR + (xRM + 1) + (xMM - 1)) - pM /. OldToNew /. δp → 0 /. FD → 0 /.
    ρH → ρ̂H)^k (* MM dies, RM born *)
 );
```

- **The second moment**

While the change in the second moment in p to O(1) is:

```
weakselection = {βRM → βRR * (1 + ϵ * s̃βRM), βMM → βRR * (1 + ϵ * s̃βMM), βM → βR * (1 + ϵ * s̃βM),
   fM → fR * (1 + ϵ * s̃fM), dRM → dRR * (1 - ϵ * s̃dRM), dMM → dRR * (1 - ϵ * s̃dMM), dM → dR * (1 - ϵ * s̃dM)};
Moment2WS = moment[2] /. OldToNew /. weakselection /. ϵ → 0 /. δp → 0 /. FD → 0 /. ρH → ρ̂H /.
   Flatten[Solve[ρ̂H == Sqrt[βRR * dRR]/(Sqrt[(fR*βR*dR)/2] + Sqrt[βRR * dRR]), βR]] // Simplify;
```

This is the change in pM, but scaling pM by N scales the variance by N^2 and taking limit of N -> Inf

We check derivation of Eq. (C6a) from Eq. (C4)

```
v = Limit[SUM^2 * Moment2WS * d̂/dharmonic, SUM → Infinity];
dharmonic = 2/(1/dR + 1/dRR);
d̂ = dR * ρ̂H + dRR * (1 - ρ̂H);
d̄ = (dR + dRR)/2;
EqC6a = pM * (1 - pM) * ((dR * ρ̂H + 2 * dRR * ρ̂D)/(ρ̂H * ρ̂D)) * 1/(4 * d̄) /. ρ̂D → 1 - ρ̂H // Simplify;
Simplify[v - EqC6a, assumption]
0
```

Setting this variance to $\frac{p(1-p)}{2\,\mathrm{Ne}}$, as in the standard diploid WF model, then defines an Ne.

That is Ne = $\frac{p(1-p)}{2\,\text{variance}}$ is a measure of the effective population size:

We check derivation of Eq. (8a) from Eq. (C6a)

$$\text{EffectiveN} = \frac{p_M * (1 - p_M)}{2 * \text{EqC6a}} \text{ // Simplify;}$$

$$\text{Eq8a} = \frac{2 * \hat{\rho}_H * \hat{\rho}_D}{\frac{\hat{\rho}_H}{\bar{d}} * d_R + 2 * \frac{\hat{\rho}_D}{\bar{d}} * d_{RR}} * \text{SUM} \text{ /. } \hat{\rho}_D \to 1 - \hat{\rho}_H\text{;}$$

$$\text{Simplify}\left[\text{EffectiveN} - \frac{\text{Eq8a}}{\text{SUM}}\right]$$

0

- **The third moment**

While the change in the third moment in p to O(1) is:

$$\text{Moment3WS} = \text{moment}[3] \text{ /. OldToNew /. weakselection /. } \epsilon \to 0 \text{ /. } \delta_p \to 0 \text{ /. } F_D \to 0 \text{ /. } \rho_H \to \hat{\rho}_H \text{ /. }$$
$$\text{Flatten}\left[\text{Solve}\left[\hat{\rho}_H == \frac{\sqrt{\beta_{RR} * d_{RR}}}{\sqrt{\frac{f_R * \beta_R * d_R}{2}} + \sqrt{\beta_{RR} * d_{RR}}}, \beta_R\right]\right] \text{ // Simplify;}$$

$$\text{Moment3} = \text{Limit}\left[\text{SUM}^2 * \text{Moment3WS}, \text{SUM} \to \text{Infinity}\right]$$

0

Thus, the third moment goes to zero, as required in a diffusion approximation.

Deriving the fixation probability in the haploid-diploid Wright-Fisher model

- **Frequency of haploids in the resident population**

Original Equations (Eq.(D.1))

$$c_R = w_{RR} * X_{RR} + \frac{w_{RM}}{2} * X_{RM};$$
$$c_M = \frac{w_{RM}}{2} * X_{RM} + w_{MM} * X_{MM};$$
$$c_{RR} = \frac{f_R}{2} * \frac{w_R^2 * X_R^2}{w_R * X_R + w_M * X_M};$$
$$c_{RM} = \frac{f_R + f_M}{2} * \frac{w_R * w_M * X_R * X_M}{w_R * X_R + w_M * X_M};$$
$$c_{MM} = \frac{f_M}{2} * \frac{w_M^2 * X_M^2}{w_R * X_R + w_M * X_M};$$
$$\text{eqHR} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_R}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_R;$$
$$\text{eqHM} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_M}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_M;$$
$$\text{eqDRR} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_{RR}}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_{RR};$$
$$\text{eqDRM} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_{RM}}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_{RM};$$
$$\text{eqDMM} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_{MM}}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_{MM};$$
$$\text{HRnext} = \text{eqHR} + X_R;$$
$$\text{HMnext} = \text{eqHM} + X_M;$$
$$\text{DRRnext} = \text{eqDRR} + X_{RR};$$
$$\text{DRMnext} = \text{eqDRM} + X_{RM};$$
$$\text{DMMnext} = \text{eqDMM} + X_{MM};$$

Calculating the frequency of haploids within a population composed entirely of residents (ρ_H), assuming the population size is large:

```
assumption = 0 < SUM && 0 < wR && 0 < wRR && 0 < fR && fR < 1;
assumption2 = 0 < pM && pM < 1 && 0 < ρH && ρH < 1;

Factor[Normal[Series[eqHR /. {XM → 0, XRM → 0, XMM → 0} /. XRR → N (1 - ρH) /. XR → N ρH /. N → N / ϵ,
    {ϵ, 0, 0}]]]
```

$$-\frac{N\left(2\,w_{RR}-4\,w_{RR}\,\rho_H-f_R\,w_R\,\rho_H^2+2\,w_{RR}\,\rho_H^2\right)}{\epsilon\,\left(-2\,w_{RR}-f_R\,w_R\,\rho_H+2\,w_{RR}\,\rho_H\right)}$$

```
Simplify[Solve[% == 0, ρH]]
```

$$\left\{\left\{\rho_H\to-\frac{\sqrt{2}\,\sqrt{f_R}\,\sqrt{w_R}\,\sqrt{w_{RR}}+2\,w_{RR}}{f_R\,w_R-2\,w_{RR}}\right\},\left\{\rho_H\to\frac{\sqrt{2}\,\sqrt{f_R}\,\sqrt{w_R}\,\sqrt{w_{RR}}-2\,w_{RR}}{f_R\,w_R-2\,w_{RR}}\right\}\right\}$$

The second of these can be rewritten as the strictly positive quantity (D.2)

$$\frac{\sqrt{w_{RR}}}{\sqrt{\tilde{w}_R}+\sqrt{w_{RR}}};$$

$$\text{Simplify}\left[\frac{\sqrt{2}\,\sqrt{f_R}\,\sqrt{w_R}\,\sqrt{w_{RR}}-2\,w_{RR}}{f_R\,w_R-2\,w_{RR}}-\%\;/.\;\tilde{w}_R\to\frac{f_R*w_R}{2},\ \text{assumption}\right]$$

0

Branching Process Approximation (WF-model)

Here we check the main calculations derived in Branching Process section of Appendix D.

Original Equations (Eq.(D.1))

$$c_R = w_{RR}*X_{RR}+\frac{w_{RM}}{2}*X_{RM};$$

$$c_M = \frac{w_{RM}}{2}*X_{RM}+w_{MM}*X_{MM};$$

$$c_{RR} = \frac{f_R}{2}*\frac{{w_R}^2*{X_R}^2}{w_R*X_R+w_M*X_M};$$

$$c_{RM} = \frac{f_R+f_M}{2}*\frac{w_R*w_M*X_R*X_M}{w_R*X_R+w_M*X_M};$$

$$c_{MM} = \frac{f_M}{2}*\frac{{w_M}^2*{X_M}^2}{w_R*X_R+w_M*X_M};$$

$$\text{eqHR} = (X_R+X_M+X_{RR}+X_{RM}+X_{MM})*\frac{c_R}{c_R+c_M+c_{RR}+c_{RM}+c_{MM}}-X_R;$$

$$\text{eqHM} = (X_R+X_M+X_{RR}+X_{RM}+X_{MM})*\frac{c_M}{c_R+c_M+c_{RR}+c_{RM}+c_{MM}}-X_M;$$

$$\text{eqDRR} = (X_R+X_M+X_{RR}+X_{RM}+X_{MM})*\frac{c_{RR}}{c_R+c_M+c_{RR}+c_{RM}+c_{MM}}-X_{RR};$$

$$\text{eqDRM} = (X_R+X_M+X_{RR}+X_{RM}+X_{MM})*\frac{c_{RM}}{c_R+c_M+c_{RR}+c_{RM}+c_{MM}}-X_{RM};$$

$$\text{eqDMM} = (X_R+X_M+X_{RR}+X_{RM}+X_{MM})*\frac{c_{MM}}{c_R+c_M+c_{RR}+c_{RM}+c_{MM}}-X_{MM};$$

```
HRnext = eqHR + XR;
HMnext = eqHM + XM;
DRRnext = eqDRR + XRR;
DRMnext = eqDRM + XRM;
DMMnext = eqDMM + XMM;
```

We define variables in the Appendx D

$$\texttt{assumption} = 0 < \texttt{SUM} \;\&\&\; 0 < w_R \;\&\&\; 0 < w_{RR} \;\&\&\; 0 < f_R \;\&\&\; f_R < 1 \;\&\&\; 0 < w_M \;\&\&\; 0 < w_{RM} \;\&\&\; 0 < f_M \;\&\&\; f_M < 1;$$

$$\tilde{w}_R = \frac{f_R * w_R}{2};$$

$$\texttt{freqH} = \frac{\sqrt{w_{RR}}}{\sqrt{\tilde{w}_R} + \sqrt{w_{RR}}};$$

$$\texttt{freqD} = \frac{\sqrt{\tilde{w}_R}}{\sqrt{\tilde{w}_R} + \sqrt{w_{RR}}};$$

$$s_{average} = \frac{1}{2} * \left(\frac{s_{fM}}{2} + s_{wM}\right) + \frac{1}{2} * s_{wRM};$$

$$\texttt{avef} = \frac{f_R + f_M}{2};$$

$$\texttt{weakselection} = \{w_{RM} \to w_{RR} * (1 + \epsilon * \tilde{s}_{wRM}),\ w_{MM} \to w_{RR} * (1 + \epsilon * \tilde{s}_{wMM}),\ w_M \to w_R * (1 + \epsilon * \tilde{s}_{wM}),\ f_M \to f_R * (1 + \epsilon * \tilde{s}_{fM})\};$$

$$\texttt{weakselection2} = \{s_{wRM} \to \epsilon * \tilde{s}_{wRM},\ s_{wMM} \to \epsilon * \tilde{s}_{wMM},\ s_{wM} \to \epsilon * \tilde{s}_{wM},\ s_{fM} \to \epsilon * \tilde{s}_{fM}\};$$

When the mutation first appears in haploids, the expected number of offspring is N*$q_{RM|H}$, which is given by:

$$N \frac{c_{RM}}{c_R + c_{RR}} \;/.\; X_M \to 1 \;/.\; X_{RM} \to 0 \;/.\; (w_M + w_R X_R) \texttt{->} (w_R X_R) \;/.\; X_R \to \hat{\rho}_H N \;/.\; X_{RR} \to \hat{\rho}_D N \;//\; \texttt{Simplify}$$

$$\frac{(f_M + f_R)\, w_M}{2\, w_{RR}\, \hat{\rho}_D + f_R\, w_R\, \hat{\rho}_H}$$

$$\frac{\texttt{avef}\ w_M}{\hat{w}} \;/.\; \hat{w} \to \frac{f_R * w_R}{2} * \hat{\rho}_H + w_{RR} * \hat{\rho}_D;$$

```
% - %% // Factor
```

0

When the mutation first appears in diploids, the expected number of offspring is N*$q_{M|D}$, which is given by:

$$N \frac{c_M}{c_R + c_{RR}} \;/.\; X_M \to 0 \;/.\; X_{RM} \to 1 \;/.\; X_{MM} \to 0 \;/.\; \left(\frac{w_{RM}}{2} + \frac{1}{2} f_R\, w_R\, X_R + w_{RR}\, X_{RR}\right) \texttt{->} \left(\frac{1}{2} f_R\, w_R\, X_R + w_{RR}\, X_{RR}\right) \;/.\; X_R \to \hat{\rho}_H N \;/.\; X_{RR} \to \hat{\rho}_D N \;//\; \texttt{Simplify}$$

$$\frac{w_{RM}}{2\, w_{RR}\, \hat{\rho}_D + f_R\, w_R\, \hat{\rho}_H}$$

$$\frac{w_{RM}}{2\, \hat{w}} \;/.\; \hat{w} \to \frac{f_R * w_R}{2} * \hat{\rho}_H + w_{RR} * \hat{\rho}_D;$$

```
% - %% // Factor
```

0

Using a weak selection approximation, we derive Eq. (D.5) from Eq. (D.4)

$$\texttt{EqD4a} = \texttt{Exp}\left[-\pi D * \frac{\texttt{avef} * w_M}{\hat{w}}\right] \;/.\; \texttt{weakselection} \;/.\; \pi D \to \tilde{\pi D} * \epsilon;$$

$$\texttt{EqD4b} = \texttt{Exp}\left[-\pi H * \frac{w_{RM}}{2 * \hat{w}}\right] \;/.\; \texttt{weakselection} \;/.\; \pi H \to \tilde{\pi H} * \epsilon;$$

$$\texttt{tempa} = \texttt{Normal}[\texttt{Series}[\texttt{EqD4a}, \{\epsilon, 0, 2\}]] \;/.\; \epsilon \to 1 \;/.\; \tilde{\pi D} \to \pi D \;/.\; \tilde{s}_{fM} \to s_{fM} \;/.\; \tilde{s}_{wM} \to s_{wM} \;/.\; \tilde{s}_{wRM} \to s_{wRM}$$

$$\texttt{tempb} = \texttt{Normal}[\texttt{Series}[\texttt{EqD4b}, \{\epsilon, 0, 2\}]] \;/.\; \epsilon \to 1 \;/.\; \tilde{\pi H} \to \pi H \;/.\; \tilde{s}_{fM} \to s_{fM} \;/.\; \tilde{s}_{wM} \to s_{wM} \;/.\; \tilde{s}_{wRM} \to s_{wRM}$$

$$1 - \frac{\pi D\, f_R\, w_R}{\hat{w}} + \frac{-\pi D\, \hat{w}\, f_R\, s_{fM}\, w_R - 2\, \pi D\, \hat{w}\, f_R\, s_{wM}\, w_R + \pi D^2\, f_R^2\, w_R^2}{2\, \hat{w}^2}$$

$$1 - \frac{\pi H\, w_{RR}}{2\, \hat{w}} + \frac{-4\, \pi H\, \hat{w}\, s_{wRM}\, w_{RR} + \pi H^2\, w_{RR}^2}{8\, \hat{w}^2}$$

Calculating the probability of fixation from these probabilities of loss, we obtain equations (D.5):

$$EqD5a = \frac{f_R * w_R}{\hat{w}} * \pi D + \frac{2 * f_R * s_{wM} * w_R + f_R * s_{fM} * w_R}{2 * \hat{w}} * \pi D - \frac{f_R^2 * w_R^2}{2 * \hat{w}^2} * \pi D^2;$$

$$EqD5b = \frac{w_{RR}}{2 * \hat{w}} * \pi H + \frac{s_{wRM} * w_{RR}}{2 * \hat{w}} * \pi H - \frac{w_{RR}^2}{8 * \hat{w}^2} * \pi H^2;$$

```
EqD5a - (1 - tempa) // Factor
EqD5b - (1 - tempb) // Factor
```

0

0

To obtain the fixation probabilities as given in equations (5b) and (5c), we substitute πD (as given by EqD5b) into EqD5a and solve for πH under weak selection. To simplify the result, we use the definition $\hat{w} \to \frac{f_R * w_R}{2} * \hat{\rho}_H + w_{RR} * (1 - \hat{\rho}_H)$ and replace for w_{RR} using its relationship to $\hat{\rho}_H$:

```
EqWSπH =
  Factor[
      Normal[
       Series[
        πH - EqD5a /. πD -> EqD5b /. ŵ → (f_R * w_R)/2 * ρ̂_H + w_RR * (1 - ρ̂_H) /.
           Flatten[Solve[freqH == ρ̂_H, w_RR]] /. πD -> πD * ϵ /. πH -> πH * ϵ /. weakselection2,
        {ϵ, 0, 2}]]] /. s̃_fM → s_fM /. s̃_wM → s_wM /. s̃_wRM → s_wRM;
solπH = Solve[EqWSπH == 0, πH]
```

$$\left\{\{\pi H \to 0\}, \left\{\pi H \to \frac{2\left(-s_{fM} - 2\, s_{wM} - 2\, s_{wRM} + s_{fM}\, \hat{\rho}_H + 2\, s_{wM}\, \hat{\rho}_H + 2\, s_{wRM}\, \hat{\rho}_H\right)}{-2 + \hat{\rho}_H}\right\}\right\}$$

The second solution is the relevant one, and it can be written as equation (5a): $\frac{4\,\hat{\rho}_D}{\hat{\rho}_H + 2\,\hat{\rho}_D} * 2 * S_{average}$

$$WF\pi H = \frac{4 * \hat{\rho}_D}{\hat{\rho}_H + 2 * \hat{\rho}_D} * 2 * S_{average};$$

```
WFπH - πH /. solπH[[2]] /. ρ̂_D → 1 - ρ̂_H // Factor
```

0

Following the same procedure to solve for πD

```
EqWSπD =
  Factor[
      Normal[
       Series[
        πD - EqD5b /. πH -> EqD5a /. ŵ → (f_R * w_R)/2 * ρ̂_H + w_RR * (1 - ρ̂_H) /.
           Flatten[Solve[freqH == ρ̂_H, w_RR]] /. πD -> πD * ϵ /. πH -> πH * ϵ /. weakselection2,
        {ϵ, 0, 2}]]] /. s̃_fM → s_fM /. s̃_wM → s_wM /. s̃_wRM → s_wRM;
solπD = Solve[EqWSπD == 0, πD]
```

$$\left\{\{\pi D \to 0\}, \left\{\pi D \to \frac{-s_{fM}\, \hat{\rho}_H - 2\, s_{wM}\, \hat{\rho}_H - 2\, s_{wRM}\, \hat{\rho}_H}{-2 + \hat{\rho}_H}\right\}\right\}$$

The second solution is the relevant one, and it can be written as equation (5a): $\frac{4\,\hat{\rho}_D}{\hat{\rho}_H + 2\,\hat{\rho}_D} * 2 * S_{average}$

$$WF\pi D = \frac{2 * \hat{\rho}_H}{\hat{\rho}_H + 2 * \hat{\rho}_D} * 2 * S_{average};$$

```
WFπD - πD /. solπD[[2]] /. ρ̂_D → 1 - ρ̂_H // Factor
```

0

Fixation probability (Eqs. (5))

$$\text{Eq5b} = \frac{2 * \tilde{w}_R * w_{RR}}{\tilde{w}_R * w_{RR} + \frac{w_{RR}}{2} * \sqrt{\tilde{w}_R * w_{RR}}} * 2 * S_{average};$$

$$\text{Eq5c} = \frac{2 * \tilde{w}_R * w_{RR}}{\tilde{w}_R * w_{RR} + 2 * \tilde{w}_R * \sqrt{\tilde{w}_R * w_{RR}}} * 2 * S_{average};$$

$$\text{Simplify}\left[\text{Eq5b} - \text{WF}\pi\text{H} \;/.\; \hat{w} \to \frac{f_R * w_R}{2} * \hat{\rho}_H + w_{RR} * \hat{\rho}_D \;/.\; \hat{\rho}_H \to \text{freqH} \;/.\; \hat{\rho}_D \to \text{freqD}, \text{assumption}\right]$$

$$\text{Simplify}\left[\text{Eq5c} - \text{WF}\pi\text{D} \;/.\; \hat{w} \to \frac{f_R * w_R}{2} * \hat{\rho}_H + w_{RR} * \hat{\rho}_D \;/.\; \hat{\rho}_H \to \text{freqH} \;/.\; \hat{\rho}_D \to \text{freqD}, \text{assumption}\right]$$

0

0

Fixation probability (Eqs. (6))

$$\text{Eq5a} = \frac{\hat{\rho}_H}{\hat{\rho}_H + 2 * \hat{\rho}_D} * \text{Eq5b} + \frac{2 * \hat{\rho}_D}{\hat{\rho}_H + 2 * \hat{\rho}_D} * \text{Eq5c};$$

$$\text{Eq6} = \frac{8 * \hat{\rho}_H * \hat{\rho}_D}{\left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)^2} * 2 * S_{average};$$

$$\text{Simplify}\left[\text{Eq5a} - \text{Eq6} \;/.\; \hat{\rho}_H \to \text{freqH} \;/.\; \hat{\rho}_D \to \text{freqD}, \text{assumption}\right]$$

0

- **Diffusion Approximation (First Moment)**

- **Diffusion Approximation (First Moment / Separation of Time Scale)**

Original Equations (Eq.(D.1))

$$c_R = w_{RR} * X_{RR} + \frac{w_{RM}}{2} * X_{RM};$$

$$c_M = \frac{w_{RM}}{2} * X_{RM} + w_{MM} * X_{MM};$$

$$c_{RR} = \frac{f_R}{2} * \frac{w_R{}^2 * X_R{}^2}{w_R * X_R + w_M * X_M};$$

$$c_{RM} = \frac{f_R + f_M}{2} * \frac{w_R * w_M * X_R * X_M}{w_R * X_R + w_M * X_M};$$

$$c_{MM} = \frac{f_M}{2} * \frac{w_M{}^2 * X_M{}^2}{w_R * X_R + w_M * X_M};$$

$$\text{eqHR} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_R}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_R;$$

$$\text{eqHM} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_M}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_M;$$

$$\text{eqDRR} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_{RR}}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_{RR};$$

$$\text{eqDRM} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_{RM}}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_{RM};$$

$$\text{eqDMM} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_{MM}}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_{MM};$$

$$\text{HRnext} = \text{eqHR} + X_R;$$
$$\text{HMnext} = \text{eqHM} + X_M;$$
$$\text{DRRnext} = \text{eqDRR} + X_{RR};$$
$$\text{DRMnext} = \text{eqDRM} + X_{RM};$$
$$\text{DMMnext} = \text{eqDMM} + X_{MM};$$

Transform variables
(Define: pM, δp, ρH, FD)
(SUM=N in the manuscript)

$$\texttt{set} = \Bigg\{ \mathrm{SUM} == X_R + X_M + X_{RR} + X_{RM} + X_{MM},$$

$$p_M == \frac{1}{2} * \frac{X_M}{X_R + X_M} + \frac{1}{2} * \frac{\frac{X_{RM}}{2} + X_{MM}}{X_{RR} + X_{RM} + X_{MM}},$$

$$\delta_p == \frac{\frac{X_{RM}}{2} + X_{MM}}{X_{RR} + X_{RM} + X_{MM}} - \frac{X_M}{X_R + X_M},$$

$$\rho_H == \frac{X_R + X_M}{X_R + X_M + X_{RR} + X_{RM} + X_{MM}},$$

$$F_D == 1 - \frac{1}{2 * \frac{\frac{X_{RM}}{2} + X_{MM}}{X_{RR} + X_{RM} + X_{MM}} * \left(1 - \frac{\frac{X_{RM}}{2} + X_{MM}}{X_{RR} + X_{RM} + X_{MM}}\right)} * \frac{X_{RM}}{X_{RR} + X_{RM} + X_{MM}}$$

$$\Bigg\};$$

OldToNew: Transformation of old variables to new variabls
NewToOld: Transformation of new variables to old variabls

```
OldToNew = Solve[set, {XR, XM, XRR, XRM, XMM}] // Flatten
NewToOld = Solve[set, {SUM, pM, ρH, δp, FD}] // Flatten
```

$$\Big\{X_R \to -\frac{1}{2} \, \mathrm{SUM} \, (-2 + 2 \, p_M - \delta_p) \, \rho_H, \; X_M \to \frac{1}{2} \, \mathrm{SUM} \, (2 \, p_M - \delta_p) \, \rho_H,$$

$$X_{RR} \to \frac{1}{4} \, \mathrm{SUM} \, (-2 + 2 \, p_M + \delta_p) \, (2 - 2 \, p_M + 2 \, F_D \, p_M - \delta_p + F_D \, \delta_p) \, (-1 + \rho_H),$$

$$X_{RM} \to -\frac{1}{2} \, \mathrm{SUM} \, (-1 + F_D) \, \left(-4 \, p_M + 4 \, p_M^2 - 2 \, \delta_p + 4 \, p_M \, \delta_p + \delta_p^2\right) \, (-1 + \rho_H),$$

$$X_{MM} \to \frac{1}{4} \, \mathrm{SUM} \, (2 \, p_M + \delta_p) \, (-2 \, F_D - 2 \, p_M + 2 \, F_D \, p_M - \delta_p + F_D \, \delta_p) \, (-1 + \rho_H)\Big\}$$

$$\Big\{\mathrm{SUM} \to X_M + X_{MM} + X_R + X_{RM} + X_{RR}, \; p_M \to \frac{4 \, X_M \, X_{MM} + 2 \, X_{MM} \, X_R + 3 \, X_M \, X_{RM} + X_R \, X_{RM} + 2 \, X_M \, X_{RR}}{4 \, (X_M + X_R) \, (X_{MM} + X_{RM} + X_{RR})},$$

$$\rho_H \to \frac{X_M + X_R}{X_M + X_{MM} + X_R + X_{RM} + X_{RR}}, \; \delta_p \to \frac{2 \, X_{MM} \, X_R - X_M \, X_{RM} + X_R \, X_{RM} - 2 \, X_M \, X_{RR}}{2 \, (X_M + X_R) \, (X_{MM} + X_{RM} + X_{RR})}, \; F_D \to \frac{-X_{RM}^2 + 4 \, X_{MM} \, X_{RR}}{(2 \, X_{MM} + X_{RM}) \, (X_{RM} + 2 \, X_{RR})}\Big\}$$

Dynamics of new variables

Dynamics pM

$$\mathrm{EqpMOLD} = \frac{1}{2} * \frac{\mathrm{HMnext}}{\mathrm{HRnext} + \mathrm{HMnext}} + \frac{1}{2} * \frac{\frac{\mathrm{DRMnext}}{2} + \mathrm{DMMnext}}{\mathrm{DRRnext} + \mathrm{DRMnext} + \mathrm{DMMnext}} - p_M;$$

```
EqpMNEW = EqpMOLD /. OldToNew;
```

Dynamics δp

$$\mathrm{Eq\delta pOLD} = \left(\frac{\frac{\mathrm{DRMnext}}{2} + \mathrm{DMMnext}}{\mathrm{DMMnext} + \mathrm{DRMnext} + \mathrm{DRRnext}} - \frac{\mathrm{HMnext}}{\mathrm{HMnext} + \mathrm{HRnext}}\right) - \delta_p;$$

```
EqδpNEW = EqδpOLD /. OldToNew;
```

Dynamics ρH

$$\mathrm{Eq\rho HOLD} = \left(\frac{\mathrm{HMnext} + \mathrm{HRnext}}{\mathrm{SUM}}\right) - \rho_H;$$

```
EqρHNEW = EqρHOLD /. OldToNew;
```

Dynamics FD

$$\textbf{EqFDOLD} = \left(1 - \left(\frac{1}{2 * \frac{\frac{\text{DRMnext}}{2}+\text{DMMnext}}{\text{DRRnext}+\text{DRMnext}+\text{DMMnext}} * \left(1 - \frac{\frac{\text{DRMnext}}{2}+\text{DMMnext}}{\text{DRRnext}+\text{DRMnext}+\text{DMMnext}}\right)} * \frac{\text{DRMnext}}{\text{DRRnext} + \text{DRMnext} + \text{DMMnext}}\right)\right) - F_D;$$

```
EqFDNEW = EqFDOLD /. OldToNew;
```

Transforming differential equations to new variable system

Asuming weak selection

$$\text{weakselection} = \{w_{RM} \to w_{RR} * (1 + \epsilon * \tilde{s}_{wRM}),\ w_{MM} \to w_{RR} * (1 + \epsilon * \tilde{s}_{wMM}),\ w_M \to w_R * (1 + \epsilon * \tilde{s}_{wM}),\ f_M \to f_R * (1 + \epsilon * \tilde{s}_{fM})\};\ \text{pMWS} = \text{EqpMNEW} \mathbin{/.} \text{weakselection} \mathbin{//} \text{Simplify}$$

$$\delta\text{pWS} = \text{Eq}\delta\text{pNEW} \mathbin{/.} \text{weakselection} \mathbin{//} \text{Simplify}$$

$$\rho\text{HWS} = \text{Eq}\rho\text{HNEW} \mathbin{/.} \text{weakselection} \mathbin{//} \text{Simplify}$$

$$\text{FDWS} = \text{EqFDNEW} \mathbin{/.} \text{weakselection} \mathbin{//} \text{Simplify}$$

$$\Big(\epsilon \Big(64 \epsilon^2 (-1 + F_D)\, p_M^5\, \tilde{s}_{wM} (\tilde{s}_{wM} + \tilde{s}_{fM} (1 + \epsilon \tilde{s}_{wM})) (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) -$$
$$16 \epsilon p_M^4 (2 \tilde{s}_{wM} ((3 - 2 \epsilon \tilde{s}_{wM} + F_D (-3 + 4 \epsilon \tilde{s}_{wM})) \tilde{s}_{wMM} - (-1 + F_D) (-6 + 7 \epsilon \tilde{s}_{wM}) \tilde{s}_{wRM}) +$$
$$\tilde{s}_{fM} (1 + \epsilon \tilde{s}_{wM}) ((3 - 4 \epsilon \tilde{s}_{wM} + F_D (-3 + 8 \epsilon \tilde{s}_{wM})) \tilde{s}_{wMM} - 2 (-1 + F_D) (-3 + 7 \epsilon \tilde{s}_{wM}) \tilde{s}_{wRM})) -$$
$$16 p_M^3 \Big(\epsilon \tilde{s}_{fM} (1 + \epsilon \tilde{s}_{wM}) \Big((-3 - 2 F_D (-3 + \delta_p) + 2 \delta_p) \tilde{s}_{wMM} + 2 (-1 + F_D) (-5 + 2 \delta_p) \tilde{s}_{wRM} +$$
$$2 \tilde{s}_{wM} \Big(1 - \epsilon \delta_p^2 (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) - 3 \epsilon \tilde{s}_{wRM} - 2 \epsilon \delta_p \tilde{s}_{wRM} +$$
$$\epsilon F_D \Big(\big(-2 - \delta_p + \delta_p^2\big) \tilde{s}_{wMM} + \big(3 + 2 \delta_p - 2 \delta_p^2\big) \tilde{s}_{wRM}\Big)\Big)\Big) +$$
$$2 \Big(-(-1 + F_D) (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) + \epsilon \tilde{s}_{wM} ((-3 - 2 F_D (-3 + \delta_p) + 2 \delta_p) \tilde{s}_{wMM} +$$
$$2 (-1 + F_D) (-5 + 2 \delta_p) \tilde{s}_{wRM}) + \epsilon \tilde{s}_{wM}^2 \Big(1 - \epsilon \delta_p^2 (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) - 3 \epsilon \tilde{s}_{wRM} - 2 \epsilon \delta_p \tilde{s}_{wRM} +$$
$$\epsilon F_D \Big(\big(-2 - \delta_p + \delta_p^2\big) \tilde{s}_{wMM} + \big(3 + 2 \delta_p - 2 \delta_p^2\big) \tilde{s}_{wRM}\Big)\Big)\Big)\Big) + \delta_p \Big(-2 (-2 + \epsilon \delta_p \tilde{s}_{wM})$$
$$\Big(\tilde{s}_{wM} \Big(-4 + \epsilon (-1 + F_D) \delta_p^2 (2 \tilde{s}_{wMM} - 3 \tilde{s}_{wRM}) - 2 \delta_p (1 + \epsilon F_D (2 \tilde{s}_{wMM} - 3 \tilde{s}_{wRM}) + 3 \epsilon \tilde{s}_{wRM})\Big) +$$
$$(-2 + \delta_p) (-\delta_p (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) - 2 \tilde{s}_{wRM} + F_D ((-2 + \delta_p) \tilde{s}_{wMM} - 2 (-1 + \delta_p) \tilde{s}_{wRM}))\Big) -$$
$$\tilde{s}_{fM} (1 + \epsilon \tilde{s}_{wM}) \Big(8 + \epsilon (-1 + F_D) \delta_p^3 ((1 + 4 \epsilon \tilde{s}_{wM}) \tilde{s}_{wMM} - 2 (1 + 3 \epsilon \tilde{s}_{wM}) \tilde{s}_{wRM}) -$$
$$4 \delta_p (-1 + 2 \epsilon \tilde{s}_{wM} - 4 \epsilon \tilde{s}_{wRM} + F_D (-3 \epsilon \tilde{s}_{wMM} + 4 \epsilon \tilde{s}_{wRM})) -$$
$$2 \epsilon \delta_p^2 ((-3 + 4 F_D) \tilde{s}_{wMM} - 6 (-1 + F_D) \tilde{s}_{wRM} + \tilde{s}_{wM} (2 + 6 \epsilon \tilde{s}_{wRM} + F_D (4 \epsilon \tilde{s}_{wMM} - 6 \epsilon \tilde{s}_{wRM})))\Big)\Big) +$$
$$4 p_M \Big(\epsilon \delta_p \tilde{s}_{wM}^2 \Big(-8 + \epsilon (-1 + F_D) \delta_p^3 (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) - 2 \epsilon \delta_p^2 (F_D (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) + 2 \tilde{s}_{wRM}) -$$
$$2 \delta_p (3 + \epsilon F_D (2 \tilde{s}_{wMM} - 3 \tilde{s}_{wRM}) + 3 \epsilon \tilde{s}_{wRM})\Big) +$$
$$\tilde{s}_{wM} \Big(8 + 8 \delta_p - 4 \epsilon (-1 + F_D) \delta_p^3 (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) + 2 \epsilon \delta_p^2 ((-3 + 6 F_D) \tilde{s}_{wMM} - 10 (-1 + F_D) \tilde{s}_{wRM})\Big) +$$
$$2 \Big(4 \delta_p (\tilde{s}_{wMM} - 3 \tilde{s}_{wRM}) - 3 \delta_p^2 (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) + 4 \tilde{s}_{wRM} +$$
$$F_D \Big(\big(4 - 8 \delta_p + 3 \delta_p^2\big) \tilde{s}_{wMM} - 2 \big(2 - 6 \delta_p + 3 \delta_p^2\big) \tilde{s}_{wRM}\Big)\Big) +$$
$$\tilde{s}_{fM} (1 + \epsilon \tilde{s}_{wM}) \Big(4 + \delta_p (4 - 8 \epsilon \tilde{s}_{wM}) + \epsilon^2 (-1 + F_D) \delta_p^4 \tilde{s}_{wM} (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) -$$
$$2 \epsilon \delta_p^3 ((-1 + F_D (1 + \epsilon \tilde{s}_{wM})) \tilde{s}_{wMM} - 2 (-1 + F_D) (1 + \epsilon \tilde{s}_{wM}) \tilde{s}_{wRM}) -$$
$$\epsilon \delta_p^2 ((3 - 6 F_D) \tilde{s}_{wMM} + 10 (-1 + F_D) \tilde{s}_{wRM} + \tilde{s}_{wM} (6 + 6 \epsilon \tilde{s}_{wRM} + F_D (4 \epsilon \tilde{s}_{wMM} - 6 \epsilon \tilde{s}_{wRM})))\Big)\Big) +$$
$$8 p_M^2 \Big(\tilde{s}_{fM} (1 + \epsilon \tilde{s}_{wM}) \Big(-2 + 3 \epsilon \delta_p \tilde{s}_{wMM} + \epsilon \delta_p^2 \tilde{s}_{wMM} + 8 \epsilon \tilde{s}_{wRM} - 6 \epsilon \delta_p \tilde{s}_{wRM} - 2 \epsilon \delta_p^2 \tilde{s}_{wRM} +$$
$$\epsilon F_D \Big(-\big(-6 + 4 \delta_p + \delta_p^2\big) \tilde{s}_{wMM} + 2 \big(-4 + 3 \delta_p + \delta_p^2\big) \tilde{s}_{wRM}\Big) + 2 \epsilon \tilde{s}_{wM}$$
$$\Big(2 + \epsilon \delta_p^2 ((-2 + 3 F_D) \tilde{s}_{wMM} - 5 (-1 + F_D) \tilde{s}_{wRM}) + \delta_p (3 - 3 \epsilon \tilde{s}_{wRM} + F_D (-2 \epsilon \tilde{s}_{wMM} + 3 \epsilon \tilde{s}_{wRM}))\Big)\Big) +$$
$$2 \Big((2 - 3 \delta_p + F_D (-4 + 3 \delta_p)) \tilde{s}_{wMM} - 6 (-1 + F_D) (-1 + \delta_p) \tilde{s}_{wRM} +$$
$$\tilde{s}_{wM} \Big(-2 + 3 \epsilon \delta_p (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) + \epsilon \delta_p^2 (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) + 8 \epsilon \tilde{s}_{wRM} +$$
$$\epsilon F_D \Big(-\big(-6 + 4 \delta_p + \delta_p^2\big) \tilde{s}_{wMM} + 2 \big(-4 + 3 \delta_p + \delta_p^2\big) \tilde{s}_{wRM}\Big)\Big) + \epsilon \tilde{s}_{wM}^2 \Big(2 + \epsilon \delta_p^2$$
$$((-2 + 3 F_D) \tilde{s}_{wMM} - 5 (-1 + F_D) \tilde{s}_{wRM}) + \delta_p (3 - 3 \epsilon \tilde{s}_{wRM} + F_D (-2 \epsilon \tilde{s}_{wMM} + 3 \epsilon \tilde{s}_{wRM}))\Big)\Big)\Big)\Big)\Big) \Big/$$
$$\Big(4 (2 + 2 \epsilon p_M \tilde{s}_{wM} - \epsilon \delta_p \tilde{s}_{wM}) (2 + 2 \epsilon p_M (\tilde{s}_{wM} + \tilde{s}_{fM} (1 + \epsilon \tilde{s}_{wM})) -$$
$$\epsilon \delta_p$$
$$(\tilde{s}_{wM} + \tilde{s}_{fM} (1 + \epsilon \tilde{s}_{wM})))$$
$$\Big(4 + \epsilon \delta_p^2 \tilde{s}_{wMM} + 4 \epsilon p_M^2 (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) - \epsilon F_D \Big(4 p_M^2 + 4 p_M (-1 + \delta_p) + (-2 + \delta_p) \delta_p\Big)$$
$$(\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) + 4 \epsilon \delta_p \tilde{s}_{wRM} -$$
$$2 \epsilon \delta_p^2 \tilde{s}_{wRM} + 4 \epsilon p_M (\delta_p (\tilde{s}_{wMM} - 2 \tilde{s}_{wRM}) + 2 \tilde{s}_{wRM})\Big)\Big)$$

$$
\begin{aligned}
&-\Big(\Big(-2\,\epsilon^{3}\,(-1+F_{D})\,\delta_{p}^{5}\,\tilde{s}_{wM}\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\\
&\quad\epsilon^{2}\,\delta_{p}^{4}\,(\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM})\,((-5+F_{D}\,(5+4\,\epsilon\,\tilde{s}_{wM}))\,\tilde{s}_{wMM}-10\,(-1+F_{D})\,(1+\epsilon\,\tilde{s}_{wM})\,\tilde{s}_{wRM})+\\
&\qquad 2\,\tilde{s}_{wM}\,((-5+F_{D}\,(5+2\,\epsilon\,\tilde{s}_{wM}))\,\tilde{s}_{wMM}-5\,(-1+F_{D})\,(2+\epsilon\,\tilde{s}_{wM})\,\tilde{s}_{wRM}))-\\
&\quad 16\,\epsilon\,(-1+p_{M})\,p_{M}\,\Big(2\,(1+\epsilon\,p_{M}\,\tilde{s}_{wM})\,(p_{M}\,\tilde{s}_{wMM}+\tilde{s}_{wRM}-2\,p_{M}\,\tilde{s}_{wRM}+\tilde{s}_{wM}\,(-1+\epsilon\,(-1+F_{D})\,p_{M}\,\tilde{s}_{wRM})+\\
&\qquad\quad F_{D}\,(-(-1+p_{M})\,\tilde{s}_{wMM}+(-1+2\,p_{M})\,\tilde{s}_{wRM}))+\\
&\qquad \tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM})\,\Big(-1+p_{M}\,(-2\,\epsilon\,\tilde{s}_{wM}+\epsilon\,F_{D}\,\tilde{s}_{wMM})+\epsilon\,(-1+F_{D})\,p_{M}^{2}\,(-\tilde{s}_{wMM}+2\,(1+\epsilon\,\tilde{s}_{wM})\,\tilde{s}_{wRM})\Big)\Big)+\\
&\quad 2\,\epsilon\,\delta_{p}^{3}\,\Big(\epsilon\,\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM})\,\Big((-1+4\,F_{D}\,(-1+p_{M})-4\,p_{M})\,\tilde{s}_{wMM}-2\,(-1+F_{D})\,(-5+4\,p_{M})\,\tilde{s}_{wRM}+\\
&\qquad\quad 2\,\tilde{s}_{wM}\,\Big(3-4\,\epsilon\,p_{M}^{2}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-\epsilon\,\tilde{s}_{wRM}-4\,\epsilon\,p_{M}\,\tilde{s}_{wRM}+\\
&\qquad\qquad \epsilon\,F_{D}\,\Big(-2\,p_{M}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+4\,p_{M}^{2}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\tilde{s}_{wRM}\Big)\Big)\Big)+\\
&\qquad 2\,\Big(-3\,(-1+F_{D})\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\epsilon\,\tilde{s}_{wM}\,((-1+4\,F_{D}\,(-1+p_{M})-4\,p_{M})\,\tilde{s}_{wMM}-\\
&\qquad\qquad 2\,(-1+F_{D})\,(-5+4\,p_{M})\,\tilde{s}_{wRM})+\epsilon\,\tilde{s}_{wM}^{2}\,\Big(3-4\,\epsilon\,p_{M}^{2}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-\epsilon\,\tilde{s}_{wRM}-\\
&\qquad\qquad 4\,\epsilon\,p_{M}\,\tilde{s}_{wRM}+\epsilon\,F_{D}\,\Big(-2\,p_{M}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+4\,p_{M}^{2}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\tilde{s}_{wRM}\Big)\Big)\Big)\Big)-\\
&\quad 8\,\delta_{p}\,\Big(2\,\Big(-4-2\,\epsilon\,p_{M}\,\tilde{s}_{wMM}-\epsilon\,p_{M}^{2}\,\tilde{s}_{wMM}-\epsilon\,\tilde{s}_{wRM}-2\,\epsilon\,p_{M}\,\tilde{s}_{wRM}+2\,\epsilon\,p_{M}^{2}\,\tilde{s}_{wRM}+\epsilon\,F_{D}\\
&\qquad\quad \Big(\Big(-1+p_{M}^{2}\Big)\,\tilde{s}_{wMM}+\Big(1+2\,p_{M}-2\,p_{M}^{2}\Big)\,\tilde{s}_{wRM}\Big)+\epsilon\,\tilde{s}_{wM}\,\Big(-1-6\,p_{M}+4\,\epsilon\,(-1+F_{D})\,p_{M}^{3}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-\\
&\qquad\qquad \epsilon\,p_{M}^{2}\,((1+4\,F_{D})\,\tilde{s}_{wMM}-10\,(-1+F_{D})\,\tilde{s}_{wRM})\Big)+\epsilon^{2}\,p_{M}\,\tilde{s}_{wM}^{2}\,\Big(-2+2\,\epsilon\,(-1+F_{D})\,p_{M}^{3}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-\\
&\qquad\qquad 2\,\epsilon\,p_{M}^{2}\,(F_{D}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+2\,\tilde{s}_{wRM})+p_{M}\,(-1+\epsilon\,(-1+F_{D})\,\tilde{s}_{wRM})\Big)\Big)+\\
&\qquad \epsilon\,\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM})\,\Big(-1-2\,p_{M}\,(3+2\,\epsilon\,\tilde{s}_{wM})+4\,\epsilon^{2}\,(-1+F_{D})\,p_{M}^{4}\,\tilde{s}_{wM}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-\\
&\qquad\quad 4\,\epsilon\,p_{M}^{3}\,((1+F_{D}\,(-1+\epsilon\,\tilde{s}_{wM}))\,\tilde{s}_{wMM}-2\,(-1+F_{D})\,(-1+\epsilon\,\tilde{s}_{wM})\,\tilde{s}_{wRM})+\\
&\qquad\quad \epsilon\,p_{M}^{2}\,(-(1+4\,F_{D})\,\tilde{s}_{wMM}+10\,(-1+F_{D})\,\tilde{s}_{wRM}+2\,\tilde{s}_{wM}\,(-1+\epsilon\,(-1+F_{D})\,\tilde{s}_{wRM}))\Big)\Big)-\\
&\quad 4\,\epsilon\,\delta_{p}^{2}\,\Big(\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM})\,\Big(7+\epsilon\,p_{M}\,\tilde{s}_{wMM}-6\,\epsilon\,p_{M}^{2}\,\tilde{s}_{wMM}-2\,\epsilon\,p_{M}\,\tilde{s}_{wRM}+12\,\epsilon\,p_{M}^{2}\,\tilde{s}_{wRM}+\\
&\qquad\quad \epsilon\,F_{D}\,\Big(\Big(1-2\,p_{M}+6\,p_{M}^{2}\Big)\,\tilde{s}_{wMM}+2\,(1-6\,p_{M})\,p_{M}\,\tilde{s}_{wRM}\Big)+\\
&\qquad\quad 2\,\epsilon\,\tilde{s}_{wM}\,\Big(1+2\,\epsilon\,p_{M}^{2}\,(F_{D}\,(\tilde{s}_{wMM}-3\,\tilde{s}_{wRM})+3\,\tilde{s}_{wRM})+p_{M}\,(5+\epsilon\,(-1+F_{D})\,\tilde{s}_{wRM})\Big)\Big)+\\
&\qquad 2\,\Big((-1-5\,p_{M}+F_{D}\,(-2+5\,p_{M}))\,\tilde{s}_{wMM}-5\,(-1+F_{D})\,(-1+2\,p_{M})\,\tilde{s}_{wRM}+\\
&\qquad\quad \epsilon\,\tilde{s}_{wM}^{2}\,\Big(1+2\,\epsilon\,p_{M}^{2}\,(F_{D}\,(\tilde{s}_{wMM}-3\,\tilde{s}_{wRM})+3\,\tilde{s}_{wRM})+p_{M}\,(5+\epsilon\,(-1+F_{D})\,\tilde{s}_{wRM})\Big)+\tilde{s}_{wM}\,\Big(7+\epsilon\,p_{M}\,(\tilde{s}_{wMM}-\\
&\qquad\qquad 2\,\tilde{s}_{wRM})-6\,\epsilon\,p_{M}^{2}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\epsilon\,F_{D}\,\Big(\Big(1-2\,p_{M}+6\,p_{M}^{2}\Big)\,\tilde{s}_{wMM}+2\,(1-6\,p_{M})\,p_{M}\,\tilde{s}_{wRM}\Big)\Big)\Big)\Big)\Big)\Big/\\
&\quad\Big(2\,(2+2\,\epsilon\,p_{M}\,\tilde{s}_{wM}-\epsilon\,\delta_{p}\,\tilde{s}_{wM})\,(2+2\,\epsilon\,p_{M}\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))-\epsilon\,\delta_{p}\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM})))\\
&\qquad\Big(4+\epsilon\,\delta_{p}^{2}\,\tilde{s}_{wMM}+4\,\epsilon\,p_{M}^{2}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-\\
&\qquad\quad \epsilon\,F_{D}\,\Big(4\,p_{M}^{2}+4\,p_{M}\,(-1+\delta_{p})+(-2+\delta_{p})\,\delta_{p}\Big)\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\\
&\qquad\quad 4\,\epsilon\,\delta_{p}\,\tilde{s}_{wRM}-2\,\epsilon\,\delta_{p}^{2}\,\tilde{s}_{wRM}+4\,\epsilon\,p_{M}\,(\delta_{p}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+2\,\tilde{s}_{wRM})\Big)\Big)\Big)\\
&\Big(f_{R}\,w_{R}\,\rho_{H}^{2}\,(-2-2\,\epsilon\,p_{M}\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))+\epsilon\,\delta_{p}\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM})))-\\
&\quad w_{RR}\,(-1+\rho_{H})^{2}\,\Big(-4-\epsilon\,\delta_{p}^{2}\,\tilde{s}_{wMM}-4\,\epsilon\,p_{M}^{2}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\epsilon\,F_{D}\,\Big(4\,p_{M}^{2}+4\,p_{M}\,(-1+\delta_{p})+(-2+\delta_{p})\,\delta_{p}\Big)\\
&\qquad (\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-4\,\epsilon\,\delta_{p}\,\tilde{s}_{wRM}+2\,\epsilon\,\delta_{p}^{2}\,\tilde{s}_{wRM}-4\,\epsilon\,p_{M}\,(\delta_{p}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+2\,\tilde{s}_{wRM})\Big)\Big)\Big/\\
&\Big(f_{R}\,w_{R}\,\rho_{H}\,(2+2\,\epsilon\,p_{M}\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))-\epsilon\,\delta_{p}\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM})))+w_{RR}\,(-1+\rho_{H})\\
&\quad\Big(-4-\epsilon\,\delta_{p}^{2}\,\tilde{s}_{wMM}-4\,\epsilon\,p_{M}^{2}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\epsilon\,F_{D}\,\Big(4\,p_{M}^{2}+4\,p_{M}\,(-1+\delta_{p})+(-2+\delta_{p})\,\delta_{p}\Big)\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-\\
&\qquad 4\,\epsilon\,\delta_{p}\,\tilde{s}_{wRM}+2\,\epsilon\,\delta_{p}^{2}\,\tilde{s}_{wRM}-4\,\epsilon\,p_{M}\,(\delta_{p}\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+2\,\tilde{s}_{wRM})\Big)\Big)\\
&\Big(\epsilon^{2}\,\Big(4\,p_{M}^{2}-4\,p_{M}\,(1+\delta_{p})+\delta_{p}\,(2+\delta_{p})\Big)\,\tilde{s}_{fM}^{2}\,(1+\epsilon\,\tilde{s}_{wM})-\\
&\quad F_{D}\,\Big(4\,(2+2\,\epsilon\,p_{M}\,\tilde{s}_{wM}-\epsilon\,\delta_{p}\,\tilde{s}_{wM})^{2}+\epsilon^{2}\,(2\,p_{M}-\delta_{p})\,\tilde{s}_{fM}^{2}\,(1+\epsilon\,\tilde{s}_{wM})\,(2-\delta_{p}\,(1+2\,\epsilon\,\tilde{s}_{wM})+p_{M}\,(2+4\,\epsilon\,\tilde{s}_{wM}))+\\
&\qquad 2\,\epsilon\,\tilde{s}_{fM}\,\Big(4-4\,\delta_{p}\,(1+2\,\epsilon\,\tilde{s}_{wM})+4\,\epsilon\,p_{M}^{2}\,\tilde{s}_{wM}\,(2+3\,\epsilon\,\tilde{s}_{wM})+\\
&\qquad\quad \epsilon\,\delta_{p}^{2}\,\tilde{s}_{wM}\,(2+3\,\epsilon\,\tilde{s}_{wM})-4\,p_{M}\,\Big(-2+2\,\epsilon\,(-2+\delta_{p})\,\tilde{s}_{wM}+3\,\epsilon^{2}\,\delta_{p}\,\tilde{s}_{wM}^{2}\Big)\Big)\Big)\Big)\Big/\\
&\quad((4+2\,\epsilon\,p_{M}\,(2\,\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))-\epsilon\,\delta_{p}\,(2\,\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM})))\\
&\qquad(4+4\,\epsilon\,p_{M}\,\tilde{s}_{wM}-2\,\epsilon\,\delta_{p}\,\tilde{s}_{wM}+\epsilon\,\tilde{s}_{fM}\,(2-\delta_{p}\,(1+2\,\epsilon\,\tilde{s}_{wM})+p_{M}\,(2+4\,\epsilon\,\tilde{s}_{wM}))))
\end{aligned}
$$

We derive Eqs. D6

Order(1): ϵ=0

```
assumption = 0 < SUM && 0 < wR && 0 < wRR && 0 < fR && fR < 1;
assumption2 = 0 < pM && pM < 1 && 0 < ρH && ρH < 1;

EqD6a = Simplify[pMWS /. ϵ → 0, assumption && assumption2]
EqD6b = Simplify[δpWS /. ϵ → 0, assumption && assumption2]
EqD6c = Simplify[ρHWS /. ϵ → 0, assumption && assumption2]
EqD6d = Simplify[FDWS /. ϵ → 0, assumption && assumption2]
```

0

$-2\,\delta_p$

$$\frac{-2\,w_{RR}\,(-1+\rho_H)^2 + f_R\,w_R\,\rho_H^2}{2\,w_{RR}\,(-1+\rho_H) - f_R\,w_R\,\rho_H}$$

$-F_D$

The demographic equilibrium reached in the absence of selection is:

```
RapidEqOrder0 = Solve[{EqD6a == 0, EqD6b == 0, EqD6c == 0, EqD6d == 0}, {pM, ρH, δp, FD}]
```

Solve::svars : 方程式はすべての"solve"変数に対しては解を与えない可能性があります． ≫

$$\left\{\left\{\rho_H \to \frac{-\sqrt{2}\,\sqrt{f_R}\,\sqrt{w_R}\,\sqrt{w_{RR}} - 2\,w_{RR}}{f_R\,w_R - 2\,w_{RR}},\ \delta_p \to 0,\ F_D \to 0\right\},\ \left\{\rho_H \to \frac{\sqrt{2}\,\sqrt{f_R}\,\sqrt{w_R}\,\sqrt{w_{RR}} - 2\,w_{RR}}{f_R\,w_R - 2\,w_{RR}},\ \delta_p \to 0,\ F_D \to 0\right\}\right\}$$

Of the two roots for fracH, the relevant one is positive and equals:

```
RapidEqOrder0ρH = Solve[{EqD6c == 0}, {ρH}]
(*check*)
check = ρH /. RapidEqOrder0ρH[[2]] // Simplify;
```

$$\text{Simplify}\left[\text{check} - \frac{\sqrt{w_{RR}}}{\sqrt{\frac{f_R * w_R}{2} + \sqrt{w_{RR}}}},\ \text{assumption}\right]$$

$$\left\{\left\{\rho_H \to \frac{-\sqrt{2}\,\sqrt{f_R}\,\sqrt{w_R}\,\sqrt{w_{RR}} - 2\,w_{RR}}{f_R\,w_R - 2\,w_{RR}}\right\},\ \left\{\rho_H \to \frac{\sqrt{2}\,\sqrt{f_R}\,\sqrt{w_R}\,\sqrt{w_{RR}} - 2\,w_{RR}}{f_R\,w_R - 2\,w_{RR}}\right\}\right\}$$

0

Confirming that this is a solution to the O(1) equations:

$$\{\text{EqD6a},\ \text{EqD6b},\ \text{EqD6c},\ \text{EqD6d}\}\ /.\ \left\{\rho_H \to \frac{\sqrt{w_{RR}}}{\sqrt{\frac{f_R * w_R}{2} + \sqrt{w_{RR}}}},\ F_D \to 0,\ \delta_p \to 0\right\}\ //\ \text{Factor}$$

$\{0,\ 0,\ 0,\ 0\}$

Order(ϵ): δp, ρH-EqρH, FD = (ϵ)
We'll now set δp, FD, and (ρH-EqρH) to 0 plus order ϵ terms (e.g., δp→ϵ δp1) and see what the change in allele frequency equals to next order, ϵ^1:

Dynamics pM

```
eqpMTrans = pMWS /. δp → (ϵ * δ̃p) /. ρH → (ρ̂H + ϵ * ρ̃H) /. FD → (ϵ * F̃D);
TSeries = Series[eqpMTrans, {ϵ, 0, 1}];
Order1pM = Normal[Collect[TSeries, {ϵ, pM, δ̃p, ρ̃H, F̃D}, Simplify]];
```

Dynamics δp

```
eqδpTrans = δpWS /. δp → (ϵ * δ̃p) /. ρH → (ρ̂H + ϵ * ρ̃H) /. FD → (ϵ * F̃D);
TSeries = Series[eqδpTrans, {ϵ, 0, 1}];
Order1δp = Normal[Collect[TSeries, {ϵ, pM, δ̃p, ρ̃H, F̃D}, Simplify]];
```

Dynamics ρH

```
eqρHTrans = ρHWS /. δp → (ϵ * δ̃p) /. ρH → (ρ̂H + ϵ * ρ̃H) /. FD → (ϵ * F̃D);
TSeries = Series[eqρHTrans, {ϵ, 0, 1}];
Order1ρH = Normal[Collect[TSeries, {ϵ, pM, δ̃p, ρ̃H, F̃D}, Simplify]];
```

Dynamics FD

```
eqFDTrans = FDWS /. δp → (ϵ * δ̃p) /. ρH → (ρ̂H + ϵ * ρ̃H) /. FD → (ϵ * F̃D);
TSeries = Series[eqFDTrans, {ϵ, 0, 1}];
Order1FD = Normal[Collect[TSeries, {ϵ, pM, δ̃p, ρ̃H, F̃D}, Simplify]];
```

Conclusion:
EqOrder1pM(p, δp)
EqOrder1δp(p, δp)
EqOrder1ρH(p, ρH)
EqOrder1FD(FD)

```
EqOrder1pM = Simplify[Factor[Order1pM /. Flatten[Solve[ρ̂H == Sqrt[wRR]/Sqrt[fR*wR/2 + Sqrt[wRR]], wR]]],
  assumption && assumption2]

EqOrder1δp = Simplify[Factor[Order1δp /. Flatten[Solve[ρ̂H == Sqrt[wRR]/Sqrt[fR*wR/2 + Sqrt[wRR]], wR]]],
  assumption && assumption2]

EqOrder1ρH = Simplify[Factor[Order1ρH /. Flatten[Solve[ρ̂H == Sqrt[wRR]/Sqrt[fR*wR/2 + Sqrt[wRR]], wR]]],
  assumption && assumption2]

EqOrder1FD = Simplify[Factor[Order1FD /. Flatten[Solve[ρ̂H == Sqrt[wRR]/Sqrt[fR*wR/2 + Sqrt[wRR]], wR]]],
  assumption && assumption2]
```

$$-\frac{1}{4}\epsilon\,(-1+p_M)\,p_M\,(\tilde{s}_{fM}+2\,(\tilde{s}_{wM}+p_M\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\tilde{s}_{wRM}))$$

$$\frac{1}{2}\epsilon\left(-p_M^2\,(\tilde{s}_{fM}+2\,(\tilde{s}_{wM}+\tilde{s}_{wMM}-3\,\tilde{s}_{wRM}))+p_M\,(\tilde{s}_{fM}+2\,\tilde{s}_{wM}-2\,\tilde{s}_{wRM})+2\,p_M^3\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-4\,\tilde{\delta}_p\right)$$

$$\epsilon\left(p_M\left(-1+\hat{\rho}_H\right)\hat{\rho}_H\,(\tilde{s}_{fM}+\tilde{s}_{wM}-2\,\tilde{s}_{wRM})-p_M^2\left(-1+\hat{\rho}_H\right)\hat{\rho}_H\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-2\,\tilde{\rho}_H\right)$$

$$-\epsilon\,\tilde{F}_D$$

From the first of these equations, we can see that the allele frequency change (pM), assuming weak selection, is only a function of pM. The other equations describe how selection alters the other variables, but these are not needed to know how pM changes. We thus have a closed one-variable model that describes the dynamics of selection acting on pM to order ϵ^1.
The variables are separated!!

Check: Eq. D.7

We check derivation of Eq. (D.7)

```
EqOrder1pM2 = EqOrder1pM /. ϵ → 1 /. s̃fM → sfM /. s̃wM → swM /. s̃wRM → swRM /. s̃wMM → swMM;

saverage = 1/2 * (sfM/2 + swM) + 1/2 * swRM;

EqD7 = pM * (1 - pM) * (2 * saverage + pM * ((1 - 2 * hw) * swMM))/2 /. hw → swRM/swMM;
Simplify[EqOrder1pM2 - EqD7, assumption]
```

0

- **Effective selection coefficient and degree of dominance**

We check derivation of Eq. (11) from Eq. (7)

```
Eq7a = SUM * EqD7;
```

$$\mathtt{P0} = \mathtt{Limit}\left[\frac{\mathtt{Eq7a}}{\mathtt{SUM} * p_M * (1 - p_M)},\ p_M \to 0\right] \mathbin{//} \mathtt{Simplify};$$

$$\mathtt{P1} = \mathtt{Limit}\left[\frac{\mathtt{Eq7a}}{\mathtt{SUM} * p_M * (1 - p_M)},\ p_M \to 1\right] \mathbin{//} \mathtt{Simplify};$$

$$s_e = \mathtt{P0} + \mathtt{P1};$$

$$h_e = \frac{\mathtt{P0}}{s_e};$$

$$\mathtt{Eq10a} = \left(2 * S_{average} + \frac{(1 - 2 * h_w) * s_{wMM}}{2}\right) /.\ h_w \to \frac{s_{wRM}}{s_{wMM}};$$

$$\mathtt{Eq10b} = \frac{2 * S_{average}}{4 * S_{average} + ((1 - 2 * h_w) * s_{wMM})} /.\ h_w \to \frac{s_{wRM}}{s_{wMM}};$$

$$\mathtt{Simplify}[s_e - \mathtt{Eq10a}]$$
$$\mathtt{Simplify}[h_e - \mathtt{Eq10b}]$$

0

0

- **Applicable weight for average allele frequency for the separation of time scale**

We show that the weighting scheme, $\left\{\frac{1}{2}, \frac{1}{2}\right\}$, is the only scheme that allows for a separation of time scales in this model.

$$c_R = w_{RR} * X_{RR} + \frac{w_{RM}}{2} * X_{RM};$$

$$c_M = \frac{w_{RM}}{2} * X_{RM} + w_{MM} * X_{MM};$$

$$c_{RR} = \frac{f_R}{2} * \frac{w_R{}^2 * X_R{}^2}{w_R * X_R + w_M * X_M};$$

$$c_{RM} = \frac{f_R + f_M}{2} * \frac{w_R * w_M * X_R * X_M}{w_R * X_R + w_M * X_M};$$

$$c_{MM} = \frac{f_M}{2} * \frac{w_M{}^2 * X_M{}^2}{w_R * X_R + w_M * X_M};$$

$$\mathtt{eqHR} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_R}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_R;$$

$$\mathtt{eqHM} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_M}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_M;$$

$$\mathtt{eqDRR} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_{RR}}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_{RR};$$

$$\mathtt{eqDRM} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_{RM}}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_{RM};$$

$$\mathtt{eqDMM} = (X_R + X_M + X_{RR} + X_{RM} + X_{MM}) * \frac{c_{MM}}{c_R + c_M + c_{RR} + c_{RM} + c_{MM}} - X_{MM};$$

$$\mathtt{HRnext} = \mathtt{eqHR} + X_R;$$
$$\mathtt{HMnext} = \mathtt{eqHM} + X_M;$$
$$\mathtt{DRRnext} = \mathtt{eqDRR} + X_{RR};$$
$$\mathtt{DRMnext} = \mathtt{eqDRM} + X_{RM};$$
$$\mathtt{DMMnext} = \mathtt{eqDMM} + X_{MM};$$

Transform variables
(Define: pM, δp, ρH, FD)
(SUM=N in the manuscript)

$$\texttt{set} = \Big\{ \\
\texttt{SUM} == X_R + X_M + X_{RR} + X_{RM} + X_{MM}, \\
p_M == a * \frac{X_M}{X_R + X_M} + (1-a) * \frac{\frac{X_{RM}}{2} + X_{MM}}{X_{RR} + X_{RM} + X_{MM}}, \\
\delta_p == \frac{\frac{X_{RM}}{2} + X_{MM}}{X_{RR} + X_{RM} + X_{MM}} - \frac{X_M}{X_R + X_M}, \\
\rho_H == \frac{X_R + X_M}{X_R + X_M + X_{RR} + X_{RM} + X_{MM}}, \\
F_D == 1 - \frac{1}{2 * \frac{\frac{X_{RM}}{2}+X_{MM}}{X_{RR}+X_{RM}+X_{MM}} * \left(1 - \frac{\frac{X_{RM}}{2}+X_{MM}}{X_{RR}+X_{RM}+X_{MM}}\right)} * \frac{X_{RM}}{X_{RR} + X_{RM} + X_{MM}} \\
\Big\};$$

OldToNew: Transformation of old variables to new variabls
NewToOld: Transformation of new variables to old variabls

$$\texttt{OldToNew = Solve[set, \{}X_R, X_M, X_{RR}, X_{RM}, X_{MM}\texttt{\}] // Flatten}$$
$$\texttt{NewToOld = Solve[set, \{SUM, }p_M, \rho_H, \delta_p, F_D\texttt{\}] // Flatten}$$

$$\Big\{X_R \to -\text{SUM}\,(-1 + p_M - \delta_p + a\,\delta_p)\,\rho_H,\; X_M \to \text{SUM}\,(p_M - \delta_p + a\,\delta_p)\,\rho_H, \\
X_{RR} \to \text{SUM}\,(-1 + p_M + a\,\delta_p)\,(1 - p_M + F_D\,p_M - a\,\delta_p + a\,F_D\,\delta_p)\,(-1 + \rho_H), \\
X_{RM} \to -2\,\text{SUM}\,(-1 + F_D)\,\left(-p_M + p_M^2 - a\,\delta_p + 2\,a\,p_M\,\delta_p + a^2\,\delta_p^2\right)\,(-1 + \rho_H), \\
X_{MM} \to \text{SUM}\,(p_M + a\,\delta_p)\,(-F_D - p_M + F_D\,p_M - a\,\delta_p + a\,F_D\,\delta_p)\,(-1 + \rho_H)\Big\}$$

$$\Big\{\text{SUM} \to X_M + X_{MM} + X_R + X_{RM} + X_{RR},\; p_M \to \frac{a\,X_M}{X_M + X_R} + \frac{(1-a)\left(X_{MM} + \frac{X_{RM}}{2}\right)}{X_{MM} + X_{RM} + X_{RR}},\; \rho_H \to \frac{X_M + X_R}{X_M + X_{MM} + X_R + X_{RM} + X_{RR}}, \\
\delta_p \to \frac{2\,X_{MM}\,X_R - X_M\,X_{RM} + X_R\,X_{RM} - 2\,X_M\,X_{RR}}{2\,(X_M + X_R)\,(X_{MM} + X_{RM} + X_{RR})},\; F_D \to \frac{-X_{RM}^2 + 4\,X_{MM}\,X_{RR}}{(2\,X_{MM} + X_{RM})\,(X_{RM} + 2\,X_{RR})}\Big\}$$

Dynamics of new variables

Dynamics pM

$$\texttt{EqpMOLD} = a * \frac{\texttt{HMnext}}{\texttt{HRnext + HMnext}} + (1-a) * \frac{\frac{\texttt{DRMnext}}{2} + \texttt{DMMnext}}{\texttt{DRRnext + DRMnext + DMMnext}} - p_M;$$
$$\texttt{EqpMNEW = EqpMOLD /. OldToNew;}$$

Dynamics δp

$$\texttt{Eq}\delta\texttt{pOLD} = \left(\frac{\frac{\texttt{DRMnext}}{2} + \texttt{DMMnext}}{\texttt{DMMnext + DRMnext + DRRnext}} - \frac{\texttt{HMnext}}{\texttt{HMnext + HRnext}}\right) - \delta_p;$$
$$\texttt{Eq}\delta\texttt{pNEW = Eq}\delta\texttt{pOLD /. OldToNew;}$$

Dynamics ρH

$$\texttt{Eq}\rho\texttt{HOLD} = \left(\frac{\texttt{HMnext + HRnext}}{\texttt{SUM}}\right) - \rho_H;$$
$$\texttt{Eq}\rho\texttt{HNEW = Eq}\rho\texttt{HOLD /. OldToNew;}$$

Dynamics FD

$$\texttt{EqFDOLD} = \left(1 - \left(\frac{1}{2 * \frac{\frac{\texttt{DRMnext}}{2}+\texttt{DMMnext}}{\texttt{DRRnext+DRMnext+DMMnext}} * \left(1 - \frac{\frac{\texttt{DRMnext}}{2}+\texttt{DMMnext}}{\texttt{DRRnext+DRMnext+DMMnext}}\right)} * \frac{\texttt{DRMnext}}{\texttt{DRRnext + DRMnext + DMMnext}}\right)\right) - F_D;$$
$$\texttt{EqFDNEW = EqFDOLD /. OldToNew;}$$

Transforming differential equations to new variable system

Asuming weak selection

```
weakselection = {wRM → wRR * (1 + ϵ * s̃wRM), wMM → wRR * (1 + ϵ * s̃wMM), wM → wR * (1 + ϵ * s̃wM),
   fM → fR * (1 + ϵ * s̃fM)}; pMWS = EqpMNEW /. weakselection // Simplify
δpWS = EqδpNEW /. weakselection // Simplify
ρHWS = EqρHNEW /. weakselection // Simplify
FDWS = EqFDNEW /. weakselection // Simplify
```

$$-p_M - \Big((-1+a)\,(p_M+(-1+a)\,\delta_p)\,(1+\epsilon\,\tilde{s}_{wM})\,\big(2\,(1+\epsilon\,p_M\,\tilde{s}_{wM}+(-1+a)\,\epsilon\,\delta_p\,\tilde{s}_{wM})+\epsilon\,\tilde{s}_{fM}\,(1+p_M\,(1+2\,\epsilon\,\tilde{s}_{wM})+(-1+a)\,\delta_p\,(1+2\,\epsilon\,\tilde{s}_{wM}))\big)\Big)\Big/\Big(2\,(1+\epsilon\,p_M\,\tilde{s}_{wM}+(-1+a)\,\epsilon\,\delta_p\,\tilde{s}_{wM})\,\big(1+\epsilon\,p_M\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))+(-1+a)\,\epsilon\,\delta_p\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))\big)\Big) - \Big(a\,(p_M+a\,\delta_p)\,\big(-1-a\,\epsilon\,\delta_p\,\tilde{s}_{wMM}+\epsilon\,F_D\,(-1+p_M+a\,\delta_p)\,(\tilde{s}_{wMM}-\tilde{s}_{wRM})-\epsilon\,\tilde{s}_{wRM}+a\,\epsilon\,\delta_p\,\tilde{s}_{wRM}+\epsilon\,p_M\,(-\tilde{s}_{wMM}+\tilde{s}_{wRM})\big)\Big)\Big/\Big(1+a^2\,\epsilon\,\delta_p^2\,\tilde{s}_{wMM}+\epsilon\,p_M^2\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-\epsilon\,F_D\,\big(p_M^2+a\,\delta_p\,(-1+a\,\delta_p)+p_M\,(-1+2\,a\,\delta_p)\big)\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+2\,a\,\epsilon\,\delta_p\,\tilde{s}_{wRM}-2\,a^2\,\epsilon\,\delta_p^2\,\tilde{s}_{wRM}+2\,\epsilon\,p_M\,(a\,\delta_p\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\tilde{s}_{wRM})\Big)$$

$$-\delta_p + \Big((p_M+(-1+a)\,\delta_p)\,(1+\epsilon\,\tilde{s}_{wM})\,\big(2\,(1+\epsilon\,p_M\,\tilde{s}_{wM}+(-1+a)\,\epsilon\,\delta_p\,\tilde{s}_{wM})+\epsilon\,\tilde{s}_{fM}\,(1+p_M\,(1+2\,\epsilon\,\tilde{s}_{wM})+(-1+a)\,\delta_p\,(1+2\,\epsilon\,\tilde{s}_{wM}))\big)\Big)\Big/\Big(2\,(1+\epsilon\,p_M\,\tilde{s}_{wM}+(-1+a)\,\epsilon\,\delta_p\,\tilde{s}_{wM})\,\big(1+\epsilon\,p_M\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))+(-1+a)\,\epsilon\,\delta_p\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))\big)\Big) + \Big((p_M+a\,\delta_p)\,\big(-1-a\,\epsilon\,\delta_p\,\tilde{s}_{wMM}+\epsilon\,F_D\,(-1+p_M+a\,\delta_p)\,(\tilde{s}_{wMM}-\tilde{s}_{wRM})-\epsilon\,\tilde{s}_{wRM}+a\,\epsilon\,\delta_p\,\tilde{s}_{wRM}+\epsilon\,p_M\,(-\tilde{s}_{wMM}+\tilde{s}_{wRM})\big)\Big)\Big/\Big(1+a^2\,\epsilon\,\delta_p^2\,\tilde{s}_{wMM}+\epsilon\,p_M^2\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-\epsilon\,F_D\,\big(p_M^2+a\,\delta_p\,(-1+a\,\delta_p)+p_M\,(-1+2\,a\,\delta_p)\big)\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+2\,a\,\epsilon\,\delta_p\,\tilde{s}_{wRM}-2\,a^2\,\epsilon\,\delta_p^2\,\tilde{s}_{wRM}+2\,\epsilon\,p_M\,(a\,\delta_p\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\tilde{s}_{wRM})\Big)$$

$$-\Big(\Big(f_R\,w_R\,\rho_H^2\,\big(1+\epsilon\,p_M\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))+(-1+a)\,\epsilon\,\delta_p\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))\big)+2\,w_{RR}\,(-1+\rho_H)^2\,\Big(-1-a^2\,\epsilon\,\delta_p^2\,\tilde{s}_{wMM}-\epsilon\,p_M^2\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\epsilon\,F_D\,\big(p_M^2+a\,\delta_p\,(-1+a\,\delta_p)+p_M\,(-1+2\,a\,\delta_p)\big)\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-2\,a\,\epsilon\,\delta_p\,\tilde{s}_{wRM}+2\,a^2\,\epsilon\,\delta_p^2\,\tilde{s}_{wRM}-2\,\epsilon\,p_M\,(a\,\delta_p\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\tilde{s}_{wRM})\Big)\Big)\Big/\Big(f_R\,w_R\,\rho_H\,\big(1+\epsilon\,p_M\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))+(-1+a)\,\epsilon\,\delta_p\,(\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))\big)+2\,w_{RR}\,(-1+\rho_H)\,\Big(-1-a^2\,\epsilon\,\delta_p^2\,\tilde{s}_{wMM}-\epsilon\,p_M^2\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\epsilon\,F_D\,\big(p_M^2+a\,\delta_p\,(-1+a\,\delta_p)+p_M\,(-1+2\,a\,\delta_p)\big)\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})-2\,a\,\epsilon\,\delta_p\,\tilde{s}_{wRM}+2\,a^2\,\epsilon\,\delta_p^2\,\tilde{s}_{wRM}-2\,\epsilon\,p_M\,(a\,\delta_p\,(\tilde{s}_{wMM}-2\,\tilde{s}_{wRM})+\tilde{s}_{wRM})\Big)\Big)\Big)$$

$$\Big(\epsilon^2\,\big(p_M^2+(-1+a)\,\delta_p\,(-1+(-1+a)\,\delta_p)+p_M\,(-1+2\,(-1+a)\,\delta_p)\big)\,\tilde{s}_{fM}^2\,(1+\epsilon\,\tilde{s}_{wM})-F_D\,\Big(4\,(1+\epsilon\,p_M\,\tilde{s}_{wM}+(-1+a)\,\epsilon\,\delta_p\,\tilde{s}_{wM})^2+\epsilon^2\,(p_M+(-1+a)\,\delta_p)\,\tilde{s}_{fM}^2\,(1+\epsilon\,\tilde{s}_{wM})\,(1+p_M\,(1+2\,\epsilon\,\tilde{s}_{wM})+(-1+a)\,\delta_p\,(1+2\,\epsilon\,\tilde{s}_{wM}))+2\,\epsilon\,\tilde{s}_{fM}\,\big(1+2\,(-1+a)\,\delta_p\,(1+2\,\epsilon\,\tilde{s}_{wM})+\epsilon\,p_M^2\,\tilde{s}_{wM}\,(2+3\,\epsilon\,\tilde{s}_{wM})+(-1+a)^2\,\epsilon\,\delta_p^2\,\tilde{s}_{wM}\,(2+3\,\epsilon\,\tilde{s}_{wM})+p_M\,(2+4\,\epsilon\,(1+(-1+a)\,\delta_p)\,\tilde{s}_{wM}+6\,(-1+a)\,\epsilon^2\,\delta_p\,\tilde{s}_{wM}^2)\big)\Big)\Big)\Big/\Big((2+\epsilon\,p_M\,(2\,\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM}))+(-1+a)\,\epsilon\,\delta_p\,(2\,\tilde{s}_{wM}+\tilde{s}_{fM}\,(1+\epsilon\,\tilde{s}_{wM})))\,\big(2\,(1+\epsilon\,p_M\,\tilde{s}_{wM}+(-1+a)\,\epsilon\,\delta_p\,\tilde{s}_{wM})+\epsilon\,\tilde{s}_{fM}\,(1+p_M\,(1+2\,\epsilon\,\tilde{s}_{wM})+(-1+a)\,\delta_p\,(1+2\,\epsilon\,\tilde{s}_{wM}))\big)\Big)$$

We derive Eqs. D5
Order(1): ϵ=0

```
assumption = 0 < SUM && 0 < wR && 0 < wRR && 0 < fR && fR < 1;
assumption2 = 0 < pM && pM < 1 && 0 < ρH && ρH < 1;

EqD5a = Simplify[pMWS /. ϵ → 0, assumption && assumption2]
EqD5b = Simplify[δpWS /. ϵ → 0, assumption && assumption2]
EqD5c = Simplify[ρHWS /. ϵ → 0, assumption && assumption2]
EqD5d = Simplify[FDWS /. ϵ → 0, assumption && assumption2]
```

$$(-1+2\,a)\,\delta_p$$

$$-2\,\delta_p$$

$$\frac{-2\,w_{RR}\,(-1+\rho_H)^2+f_R\,w_R\,\rho_H^2}{2\,w_{RR}\,(-1+\rho_H)-f_R\,w_R\,\rho_H}$$

$$-F_D$$

Order(ϵ): δp, ρH-EqρH, FD = (ϵ)
We’ll now set δp, FD, and (ρH-EqρH) to 0 plus order ϵ terms (e.g., δp→ϵ δp1) and see what the change in allele frequency equals to next order, ϵ^1:

Dynamics pM

```
eqpMTrans = pMWS /. δp → (ϵ * δ̃p) /. ρH → (ρ̂H + ϵ * ρ̃H) /. FD → (ϵ * F̃D);
TSeries = Series[eqpMTrans, {ϵ, 0, 1}];
Order1pM = Normal[Collect[TSeries, {ϵ, pM, δ̃p, ρ̃H, F̃D}, Simplify]];
```

Dynamics δp

```
eqδpTrans = δpWS /. δp → (ϵ * δ̃p) /. ρH → (ρ̂H + ϵ * ρ̃H) /. FD → (ϵ * F̃D);
TSeries = Series[eqδpTrans, {ϵ, 0, 1}];
Order1δp = Normal[Collect[TSeries, {ϵ, pM, δ̃p, ρ̃H, F̃D}, Simplify]];
```

Dynamics ρH

```
eqρHTrans = ρHWS /. δp → (ϵ * δ̃p) /. ρH → (ρ̂H + ϵ * ρ̃H) /. FD → (ϵ * F̃D);
TSeries = Series[eqρHTrans, {ϵ, 0, 1}];
Order1ρH = Normal[Collect[TSeries, {ϵ, pM, δ̃p, ρ̃H, F̃D}, Simplify]];
```

Dynamics FD

```
eqFDTrans = FDWS /. δp → (ϵ * δ̃p) /. ρH → (ρ̂H + ϵ * ρ̃H) /. FD → (ϵ * F̃D);
TSeries = Series[eqFDTrans, {ϵ, 0, 1}];
Order1FD = Normal[Collect[TSeries, {ϵ, pM, δ̃p, ρ̃H, F̃D}, Simplify]];
```

Conclusion:
EqOrder1pM(p, δp)
EqOrder1δp(p, δp)
EqOrder1ρH(p, ρH)
EqOrder1FD(FD)

```
EqOrder1pM = Simplify[Factor[Order1pM /. Flatten[Solve[ρ̂H == Sqrt[wRR]/Sqrt[fR*wR/2 + Sqrt[wRR]], wR]]],
  assumption && assumption2]
EqOrder1δp = Simplify[Factor[Order1δp /. Flatten[Solve[ρ̂H == Sqrt[wRR]/Sqrt[fR*wR/2 + Sqrt[wRR]], wR]]],
  assumption && assumption2]
EqOrder1ρH = Simplify[Factor[Order1ρH /. Flatten[Solve[ρ̂H == Sqrt[wRR]/Sqrt[fR*wR/2 + Sqrt[wRR]], wR]]],
  assumption && assumption2]
EqOrder1FD = Simplify[Factor[Order1FD /. Flatten[Solve[ρ̂H == Sqrt[wRR]/Sqrt[fR*wR/2 + Sqrt[wRR]], wR]]],
  assumption && assumption2]
```

$$-\frac{1}{2}\epsilon\left(p_M^2\left(-(-1+a)\,\tilde{s}_{fM}-2(-1+a)\,\tilde{s}_{wM}-2a\left(\tilde{s}_{wMM}-3\tilde{s}_{wRM}\right)\right)+2a\,p_M^3\left(\tilde{s}_{wMM}-2\tilde{s}_{wRM}\right)+p_M\left((-1+a)\,\tilde{s}_{fM}+2(-1+a)\,\tilde{s}_{wM}-2a\,\tilde{s}_{wRM}\right)+2(1-2a)\,\tilde{\delta}_p\right)$$

$$\frac{1}{2}\epsilon\left(-p_M^2\left(\tilde{s}_{fM}+2\left(\tilde{s}_{wM}+\tilde{s}_{wMM}-3\tilde{s}_{wRM}\right)\right)+p_M\left(\tilde{s}_{fM}+2\tilde{s}_{wM}-2\tilde{s}_{wRM}\right)+2p_M^3\left(\tilde{s}_{wMM}-2\tilde{s}_{wRM}\right)-4\tilde{\delta}_p\right)$$

$$\epsilon\left(p_M\left(-1+\hat{\rho}_H\right)\hat{\rho}_H\left(\tilde{s}_{fM}+\tilde{s}_{wM}-2\tilde{s}_{wRM}\right)-p_M^2\left(-1+\hat{\rho}_H\right)\hat{\rho}_H\left(\tilde{s}_{wMM}-2\tilde{s}_{wRM}\right)-2\tilde{\rho}_H\right)$$

$$-\epsilon\,\tilde{F}_D$$

Finally, we ask what the weighting term “a” must equal for the dynamics of the allele frequency to be independent from $\tilde{\delta}_p$:

```
Solve[D[EqOrder1pM, δ̃p] == 0, a]
```

$\{\{a \to \frac{1}{2}\}\}$

Recalling that the death rate is one over the longevity, this demonstrates that the allele frequency in haploids weighted by the average haploid longevity ($\frac{lR}{lR+lRR}$) plus the allele frequency in diploids times the average diploid longevity ($\frac{lRR}{lR+lRR}$) allows for a separation of time scales where the allele frequency dynamics can be separated from the dynamics of the remaining variables to this order.

Diffusion Approximation (Second Moment)

The change in moments for pM

```
assumption = 0 < SUM && 0 < wR && 0 < wRR && 0 < fR && fR < 1;
assumption2 = 0 < pM && pM < 1 && 0 < ρH && ρH < 1;
```

Births then occur in proportion to c(GT)/totC.

$$c_R = w_{RR} * X_{RR} + \frac{w_{RM}}{2} * X_{RM};$$
$$c_M = \frac{w_{RM}}{2} * X_{RM} + w_{MM} * X_{MM};$$
$$c_{RR} = \frac{f_R}{2} * \frac{{w_R}^2 * {X_R}^2}{w_R * X_R + w_M * X_M};$$
$$c_{RM} = \frac{f_R + f_M}{2} * \frac{w_R * w_M * X_R * X_M}{w_R * X_R + w_M * X_M};$$
$$c_{MM} = \frac{f_M}{2} * \frac{{w_M}^2 * {X_M}^2}{w_R * X_R + w_M * X_M};$$
$$c_{tot} = (c_R + c_M + c_{RR} + c_{RM} + c_{MM});$$
$$\tilde{p}_R = \frac{c_R}{c_{tot}};$$
$$\tilde{p}_M = \frac{c_M}{c_{tot}};$$
$$\tilde{p}_{RR} = \frac{c_{RR}}{c_{tot}};$$
$$\tilde{p}_{RM} = \frac{c_{RM}}{c_{tot}};$$
$$\tilde{p}_{MM} = \frac{c_{MM}}{c_{tot}};$$

The second moment

Here, we ignore the small fluctuations that occur in the ratio of haploids to diploids within the population. That is, we assume that the haploid population is replaced by mutant M spores(produced by diploid) with probability $\tilde{p}_{MH} = \frac{c_M}{c_R + c_M}$, and the diploid population is replaced by mutant RM zygotes (produced by haploids) with probability $\tilde{p}_{RMD} = \frac{c_{RM}}{c_{RR} + c_{RM} + c_{MM}}$. This assumes that both haploid and diploid populations are large and drift in their ratio can be ignored. Drift in allele frequencies can, however, still be substantial, especially when alleles are rare.

$$\tilde{p}_{MH} = \frac{c_M}{c_R + c_M};$$
$$\tilde{p}_{RMD} = \frac{c_{RM}}{c_{RR} + c_{RM} + c_{MM}};$$
$$\tilde{p}_{MMD} = \frac{c_{MM}}{c_{RR} + c_{RM} + c_{MM}};$$
$$mM = \tilde{p}_{MH} * SUM * \hat{\rho}_H;$$
$$vM = \tilde{p}_{MH} * (1 - \tilde{p}_{MH}) * SUM * \hat{\rho}_H;$$
$$mRM = \tilde{p}_{RMD} * SUM * \hat{\rho}_D;$$
$$vRM = \tilde{p}_{RMD} * (1 - \tilde{p}_{RMD}) * SUM * \hat{\rho}_D;$$
$$mMM = \tilde{p}_{MMD} * SUM * \hat{\rho}_D;$$
$$vMM = \tilde{p}_{MMD} * (1 - \tilde{p}_{MMD}) * SUM * \hat{\rho}_D;$$
$$covRMMM = -\tilde{p}_{RMD} * \tilde{p}_{MMD} * SUM * \hat{\rho}_D;$$

we have

E[△pH] = E[pH'] - E[pH]

$E\left[\triangle pH^2\right]$ = E[pH'^2] - 2 E[pH'] pH + pH^2.

Transform variables
(Define: pM, δp, ρH, FD)
(SUM=N in the manuscript)

$$set = \Big\{ SUM == X_R + X_M + X_{RR} + X_{RM} + X_{MM},$$
$$p_M == \frac{1}{2} * \frac{X_M}{X_R + X_M} + \frac{1}{2} * \frac{\frac{X_{RM}}{2} + X_{MM}}{X_{RR} + X_{RM} + X_{MM}},$$
$$\delta_p == \frac{\frac{X_{RM}}{2} + X_{MM}}{X_{RR} + X_{RM} + X_{MM}} - \frac{X_M}{X_R + X_M},$$
$$\rho_H == \frac{X_R + X_M}{X_R + X_M + X_{RR} + X_{RM} + X_{MM}},$$
$$F_D == 1 - \frac{1}{2 * \frac{\frac{X_{RM}}{2}+X_{MM}}{X_{RR}+X_{RM}+X_{MM}} * \left(1 - \frac{\frac{X_{RM}}{2}+X_{MM}}{X_{RR}+X_{RM}+X_{MM}}\right)} * \frac{X_{RM}}{X_{RR} + X_{RM} + X_{MM}}$$
$$\Big\};$$

OldToNew: Transformation of old variables to new variabls
NewToOld: Transformation of new variables to old variabls

```
OldToNew = Solve[set, {X_R, X_M, X_RR, X_RM, X_MM}] // Flatten
NewToOld = Solve[set, {SUM, p_M, ρ_H, δ_p, F_D}] // Flatten
```

$$\Big\{X_R \to -\frac{1}{2} SUM (-2 + 2 p_M - \delta_p) \rho_H,\ X_M \to \frac{1}{2} SUM (2 p_M - \delta_p) \rho_H,$$
$$X_{RR} \to \frac{1}{4} SUM (-2 + 2 p_M + \delta_p) (2 - 2 p_M + 2 F_D p_M - \delta_p + F_D \delta_p) (-1 + \rho_H),$$
$$X_{RM} \to -\frac{1}{2} SUM (-1 + F_D) \left(-4 p_M + 4 p_M^2 - 2 \delta_p + 4 p_M \delta_p + \delta_p^2\right) (-1 + \rho_H),$$
$$X_{MM} \to \frac{1}{4} SUM (2 p_M + \delta_p) (-2 F_D - 2 p_M + 2 F_D p_M - \delta_p + F_D \delta_p) (-1 + \rho_H)\Big\}$$

$$\Big\{SUM \to X_M + X_{MM} + X_R + X_{RM} + X_{RR},\ p_M \to \frac{4 X_M X_{MM} + 2 X_{MM} X_R + 3 X_M X_{RM} + X_R X_{RM} + 2 X_M X_{RR}}{4 (X_M + X_R) (X_{MM} + X_{RM} + X_{RR})},$$
$$\rho_H \to \frac{X_M + X_R}{X_M + X_{MM} + X_R + X_{RM} + X_{RR}},\ \delta_p \to \frac{2 X_{MM} X_R - X_M X_{RM} + X_R X_{RM} - 2 X_M X_{RR}}{2 (X_M + X_R) (X_{MM} + X_{RM} + X_{RR})},\ F_D \to \frac{-X_{RM}^2 + 4 X_{MM} X_{RR}}{(2 X_{MM} + X_{RM}) (X_{RM} + 2 X_{RR})}\Big\}$$

```
ExpectpH1 = mM/(SUM * ρ̂_H);
ExpectpH2 = (vM + mM^2)/(SUM * ρ̂_H)^2;
ExpectΔpH1 = ExpectpH1 - X_M/(SUM * ρ̂_H);
ExpectΔpH2 = ExpectpH2 - 2 * ExpectpH1 * (X_M/(SUM * ρ̂_H)) + (X_M/(SUM * ρ̂_H))^2;
```

we have
E[△pD] = E[pD'] - E[pD]
E[△pD^2] = E[pD'^2] - 2 E[pD'] pD + pD^2

```
ExpectpD1 = (mRM/2 + mMM)/(SUM * ρ̂_D);
ExpectpD2 = (vRM + mRM^2)/(2 * SUM * ρ̂_D)^2 + 2 * (covRMMM + mRM * mMM)/((2 * SUM * ρ̂_D) * (SUM * ρ̂_D)) + (vMM + mMM^2)/(SUM * ρ̂_D)^2;
ExpectΔpD1 = ExpectpD1 - (X_RM/2 + X_MM)/(SUM * ρ̂_D);
ExpectΔpD2 = ExpectpD2 - 2 * ExpectpD1 * ((X_RM/2 + X_MM)/(SUM * ρ̂_D)) + ((X_RM/2 + X_MM)/(SUM * ρ̂_D))^2;
```

Using these, we have $\triangle pM^2 = \frac{1}{2^2}\left(\triangle pH^2 + 2\,\triangle pH\triangle pD + \triangle pD^2\right)$

This is the change in pM, but scaling pM by N scales the variance by *N* and taking limit of N -> Inf
We have Eq. (7b)

```
weakselection = {w_RM → w_RR * (1 + ϵ * s̃_wRM), w_MM → w_RR * (1 + ϵ * s̃_wMM), w_M → w_R * (1 + ϵ * s̃_wM),
    f_M → f_R * (1 + ϵ * s̃_fM)};
Moment2WS =
  (ExpectΔpH2 + 2 * ExpectΔpH1 * ExpectΔpD1 + ExpectΔpD2)/4 /. OldToNew /. weakselection /. ϵ → 0 /.
        ρ_H → ρ̂_H /. δ_p → 0 /. F_D → 0 /. ρ̂_D → 1 - ρ̂_H /. Flatten[Solve[ρ̂_H == Sqrt[w_RR]/Sqrt[(f_R*w_R)/2 + Sqrt[w_RR]], w_R]] //
    Simplify;
v = Limit[SUM * Moment2WS, SUM → Infinity];
Eq7b = (p_M * (1 - p_M) * (ρ̂_H + 2 * ρ̂_D))/(8 * ρ̂_H * ρ̂_D);
Simplify[Eq7b - v /. ρ̂_D → 1 - ρ̂_H]
```

0

Examples of the computer simulations of the figures.

Here we copied the results of computer simulations by *Mathematica*.
We simulated by Wolfram *Mathematica* 10 and figures are prepared by adobe illustrator.

Fig.1(a)

Parameter values

```
Clear[x]

Site = 100;
fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 2 * x;
SELβMM = 0;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0.5;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

```
MinV = -0.02;
MaxV = 0.31;
```

$$s_{average} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SEL}\beta\text{M} + \text{SELdM}\right) + \frac{1}{2} * (\text{SEL}\beta\text{MM} * \text{HETERO}\beta + \text{SELdM} * \text{HETEROd})$$

```
0. + x
```

Out put

```
SampleMax = 10 000;
ChangeMax = 6;
OutputData = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulation

```
VectpM = Table[0, {SampleMax}];

Resultparameter = Table[0, {ChangeMax + 1}];
Result = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.015];
  If[change == 2, x = -0.01];
  If[change == 3, x = 0];
  If[change == 4, x = 0.05];
  If[change == 5, x = 0.1];
  If[change == 6, x = 0.2];
  If[change == 7, x = 0.3];
```

```
sample = 1;
For[sample = 1, sample ≤ SampleMax, sample = sample + 1,

  EqρH = Sqrt[βRR * dRR] / Sqrt[(fR*βR*dR)/2 + Sqrt[βRR * dRR]];
  xR = Round[Site * EqρH];
  xM = 0;
  xRR = Site - xR;
  xRM = 0;
  xMM = 0;
  sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];
  xR = xR - sampleMutation[[1]];
  xM = xM + sampleMutation[[1]];
  xRR = xRR - sampleMutation[[2]];
  xRM = xRM + sampleMutation[[2]];
  xMM = 0;

  If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
  If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
  If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

  step = 2;
  While[0 < pM && pM < 1,
   pDeathR = (dR * xR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   pDeathM = (dM * xM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   pDeathRR = (dRR * xRR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   pDeathRM = (dRM * xRM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   pDeathMM = (dMM * xMM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   sampleDeath =
    RandomVariate[MultinomialDistribution[1,
      {pDeathR, pDeathM, pDeathRR, pDeathRM, pDeathMM}]];
   xR = xR - sampleDeath[[1]];
   xM = xM - sampleDeath[[2]];
   xRR = xRR - sampleDeath[[3]];
   xRM = xRM - sampleDeath[[4]];
   xMM = xMM - sampleDeath[[5]];

   cR = βRR * xRR + βRM/2 * xRM;
   cM = βMM * xMM + βRM/2 * xRM;
   If[xR == 0 && xM == 0, cRR = 0];
```

```
      If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (βR^2 * xR^2)/(βR * xR + βM * xM)];
      If[xR == 0 && xM == 0, cRM = 0];
      If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (βR * βM * xR * xM)/(βR * xR + βM * xM)];
      If[xR == 0 && xM == 0, cMM = 0];
      If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (βM^2 * xM^2)/(βR * xR + βM * xM)];
      pCellR = cR/(cR + cM + cRR + cRM + cMM);
      pCellM = cM/(cR + cM + cRR + cRM + cMM);
      pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
      pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
      pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
      sampleBirth =
       RandomVariate[MultinomialDistribution[1,
         {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

      xR = xR + sampleBirth[[1]];
      xM = xM + sampleBirth[[2]];
      xRR = xRR + sampleBirth[[3]];
      xRM = xRM + sampleBirth[[4]];
      xMM = xMM + sampleBirth[[5]];

      If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
      If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
      If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

      step = step + 1;
     ];

    VectpM[[sample]] = pM;

    OutputData[[countData, 1]] = x;
    OutputData[[countData, 2]] = pM;
    countData = countData + 1;

   ];

  Resultparameter[[change]] = x;
  Result[[change]] = Total[VectpM];

  OutputResult[[countResult, 1]] = x;
  OutputResult[[countResult, 2]] = Total[VectpM];
  countResult = countResult + 1;

  Print[change]

 ];
```

1

2

3

4

5

6

7

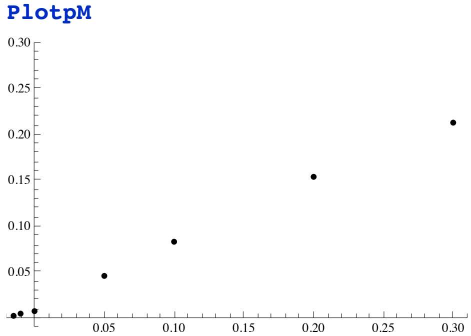
Output of the data

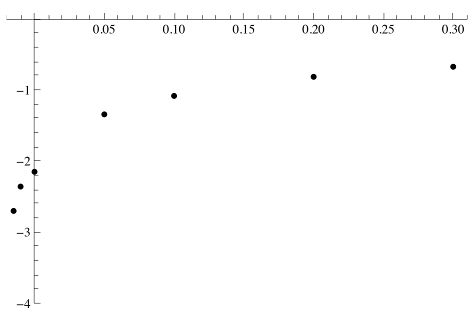
```
Resultparameter
Result
Export["/Users/Bessho/Desktop/workstation/OutputDataFig1a.txt", OutputData, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig1a.txt", OutputResult, "Table"];
```

```
{-0.015, -0.01, 0, 0.05, 0.1, 0.2, 0.3}
{20., 44., 71., 455., 827., 1534., 2124.}
```

Plot

```
PlotpM = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM[[change, 1]] = Resultparameter[[change]];
  PlotpM[[change, 2]] = Result[[change]] / SampleMax;
  PlotpMlog[[change, 1]] = Resultparameter[[change]];
  PlotpMlog[[change, 2]] = Log[10, Result[[change]] / SampleMax];
 ];
gNumerical = ListPlot[PlotpM, PlotRange → {{MinV, MaxV}, {-0.01, 0.3}},
  PlotStyle → {Thickness[0.005], Black}]
gLogNumerical = ListPlot[PlotpMlog, PlotRange → {{MinV, MaxV}, {-4, 0}},
  PlotStyle → {Thickness[0.005], Black}]
PlotpM
```






```
{{-0.015, 0.002}, {-0.01, 0.0044}, {0, 0.0071},
 {0.05, 0.0455}, {0.1, 0.0827}, {0.2, 0.1534}, {0.3, 0.2124}}
```

Wilson score interval (95% Confidence interval)

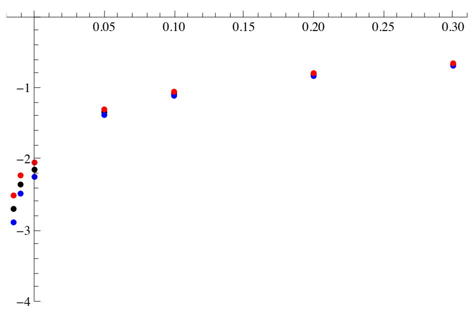
Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \overline{z}] \sim 0.95$.

```
z = 1.96;
```

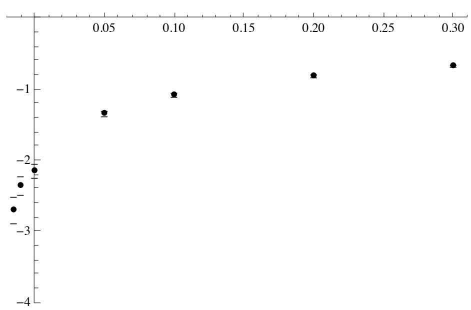
$$\underline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /.\ n \to \text{SampleMax} /.\ \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

$$\overline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /.\ n \to \text{SampleMax} /.\ \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

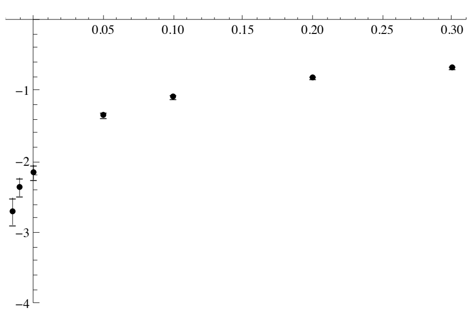
```
{0.0012951, 0.00327942, 0.00563309, 0.041587, 0.0774605, 0.14647, 0.204495}
{0.00308737, 0.00590121, 0.00894547, 0.0497621, 0.08826, 0.160596, 0.220526}
PlotCIunder = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder[[change, 1]] = Resultparameter[[change]];
  PlotCIupper[[change, 1]] = Resultparameter[[change]];
  PlotCIunder[[change, 2]] = Log[10, z̲[[change]]];
  PlotCIupper[[change, 2]] = Log[10, z̄[[change]]];
 ];
gCIupperDot = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Red}];
gCIunderDot = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Blue}];
Show[gLogNumerical, gCIunderDot, gCIupperDot]
```




```
Log[10, z̄]
N[Log[10, PlotpM[[All, 2]]]]
Log[10, z̲]
gCIupper = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
Show[gLogNumerical, gCIunder, gCIupper]

{-2.51041, -2.22906, -2.0484, -1.3031, -1.05424, -0.794264, -0.65654}
{-2.69897, -2.35655, -2.14874, -1.34199, -1.08249, -0.814175, -0.672845}
{-2.8877, -2.4842, -2.24925, -1.38104, -1.11092, -0.834252, -0.689318}
```



```
PlotCI1 = Table[0, {i, 2}, {j, 2}];
PlotCI2 = Table[0, {i, 2}, {j, 2}];
PlotCI3 = Table[0, {i, 2}, {j, 2}];
PlotCI4 = Table[0, {i, 2}, {j, 2}];
PlotCI5 = Table[0, {i, 2}, {j, 2}];
PlotCI6 = Table[0, {i, 2}, {j, 2}];
PlotCI7 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1[[1, 1]] = Resultparameter[[1]];
  PlotCI1[[1, 2]] = Log[10, z̲[[1]]];
  PlotCI1[[2, 1]] = Resultparameter[[1]];
  PlotCI1[[2, 2]] = Log[10, z̄[[1]]];
  PlotCI2[[1, 1]] = Resultparameter[[2]];
  PlotCI2[[1, 2]] = Log[10, z̲[[2]]];
  PlotCI2[[2, 1]] = Resultparameter[[2]];
  PlotCI2[[2, 2]] = Log[10, z̄[[2]]];
  PlotCI3[[1, 1]] = Resultparameter[[3]];
  PlotCI3[[1, 2]] = Log[10, z̲[[3]]];
  PlotCI3[[2, 1]] = Resultparameter[[3]];
  PlotCI3[[2, 2]] = Log[10, z̄[[3]]];
  PlotCI4[[1, 1]] = Resultparameter[[4]];
  PlotCI4[[1, 2]] = Log[10, z̲[[4]]];
  PlotCI4[[2, 1]] = Resultparameter[[4]];
  PlotCI4[[2, 2]] = Log[10, z̄[[4]]];
  PlotCI5[[1, 1]] = Resultparameter[[5]];
  PlotCI5[[1, 2]] = Log[10, z̲[[5]]];
  PlotCI5[[2, 1]] = Resultparameter[[5]];
  PlotCI5[[2, 2]] = Log[10, z̄[[5]]];
  PlotCI6[[1, 1]] = Resultparameter[[6]];
  PlotCI6[[1, 2]] = Log[10, z̲[[6]]];
  PlotCI6[[2, 1]] = Resultparameter[[6]];
  PlotCI6[[2, 2]] = Log[10, z̄[[6]]];
  PlotCI7[[1, 1]] = Resultparameter[[7]];
  PlotCI7[[1, 2]] = Log[10, z̲[[7]]];
  PlotCI7[[2, 1]] = Resultparameter[[7]];
  PlotCI7[[2, 2]] = Log[10, z̄[[7]]];
 ];
gCI1 = ListPlot[PlotCI1, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2 = ListPlot[PlotCI2, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3 = ListPlot[PlotCI3, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4 = ListPlot[PlotCI4, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5 = ListPlot[PlotCI5, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI6 = ListPlot[PlotCI6, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI7 = ListPlot[PlotCI7, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper]
```



Analytical result (Branching process approximation)

```
Clear[y]

fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 2 * y;
SELβMM = 0;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SEL}\beta\text{M} + \text{SELdM}\right) + \frac{1}{2} * (\text{SEL}\beta\text{MM} * \text{HETERO}\beta + \text{SELdM} * \text{HETEROd})$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\beta\text{RR} * \text{dRR}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

$$\text{FaiR} = \frac{\beta\text{R}}{\text{dR}};$$

$$\text{FaiM} = \frac{\beta\text{M}}{\text{dM}};$$

$$\text{FaiRR} = \frac{\beta\text{RR}}{\text{dRR}};$$

$$\text{FaiRM} = \frac{\beta\text{RM}}{\text{dRM}};$$

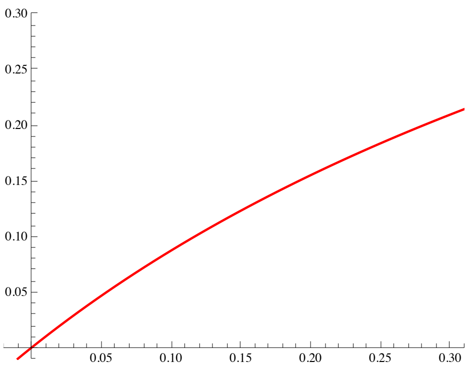
$$\text{avef} = \frac{\text{fM} + \text{fR}}{2};$$

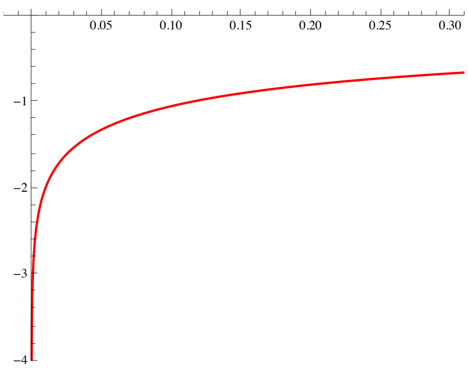
$$\text{FaiR2} = \frac{\text{fR}}{2} * \text{FaiR};$$

$$\text{FaiM2} = \frac{\text{avef}}{2} * \text{FaiM};$$

```
BPPaiH = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + FaiRM/2 * Sqrt[FaiR2 * FaiRR]);
BPPaiD = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + 2 * FaiM2 * Sqrt[FaiR2 * FaiRR]);
BPfix = EqρH / (EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD) / (EqρH + 2 * EqρD) * BPPaiD;
gBPfix = Plot[BPfix, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.3}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gLogBPfix = Plot[Log[10, BPfix], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-4, 0}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
```

0. + y





Analytical result (Diffusion approximation)

$$\text{Parameters} = \left\{\text{SUM} \to \text{Site},\ d_R \to \text{dR},\ d_{RR} \to \text{dRR},\ \hat{\rho}_H \to \frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}},\right.$$

$$\hat{\rho}_D \to \frac{\sqrt{\frac{fR*\beta R*dR}{2}}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}},\ \text{sfM} \to \text{SELfM},\ s\beta M \to \text{SEL}\beta M,\ s\beta RM \to \text{SEL}\beta RM,\ s\beta MM \to \text{SEL}\beta MM,$$

$$\left.\text{sdM} \to \text{SELdM},\ \text{sdRM} \to \text{SELdRM},\ \text{sdMM} \to \text{SELdMM}\right\}$$

$$\text{Additive} = s\beta MM - 2 * s\beta RM + \text{sdMM} - 2 * \text{sdRM};$$

$$d_{harmonic} = \frac{2}{\frac{1}{d_R} + \frac{1}{d_{RR}}};$$

$$\hat{d} = d_R * \hat{\rho}_H + d_{RR} * \left(1 - \hat{\rho}_H\right);$$

$$\overline{d} = \frac{d_R + d_{RR}}{2};$$

$$m = \text{SUM} * p * (1 - p) * \frac{2 * s_{average} + p * \text{Additive}}{2};$$

$$v = \frac{p * (1 - p) * \left(\frac{d_R}{\hat{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\hat{d}} * \hat{\rho}_D\right)}{4 * \hat{\rho}_H * \hat{\rho}_D};$$

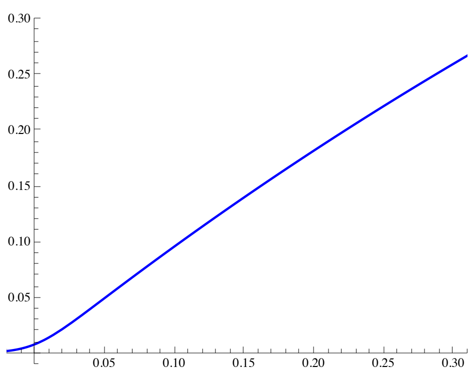
$$Q = \text{Integrate}\left[\frac{m}{v} \text{ /. Parameters},\ p\right];$$

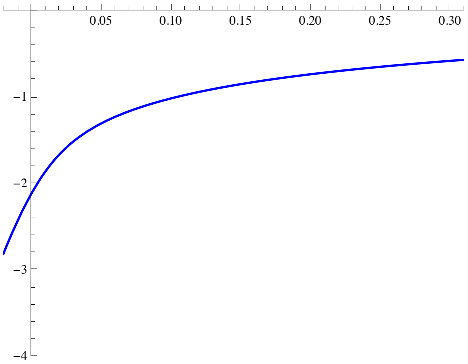
$$\text{INI} = \frac{1}{\text{SUM} * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)} \text{ /. Parameters};$$

$$\text{DFfixWS} = \frac{\text{Integrate}[\text{Exp}[-2 * Q],\ \{p,\ 0,\ \text{INI}\}]}{\text{Integrate}[\text{Exp}[-2 * Q],\ \{p,\ 0,\ 1\}]};$$

```
gDFfixWS = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.3}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS = Plot[Log[10, DFfixWS], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-4, 0}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

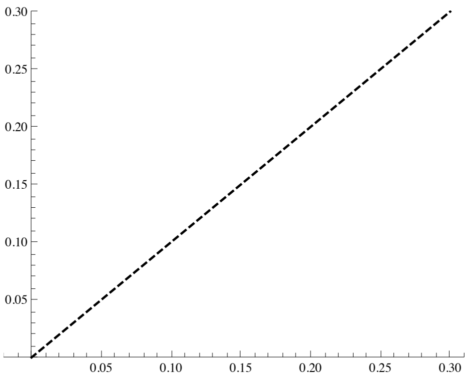
$$\{\text{SUM} \to 100,\ d_R \to 0.005,\ d_{RR} \to 0.005,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,$$
$$\text{sfM} \to 0,\ s\beta M \to 2\,y,\ s\beta RM \to 0.,\ s\beta MM \to 0,\ \text{sdM} \to 0,\ \text{sdRM} \to 0.,\ \text{sdMM} \to 0\}$$

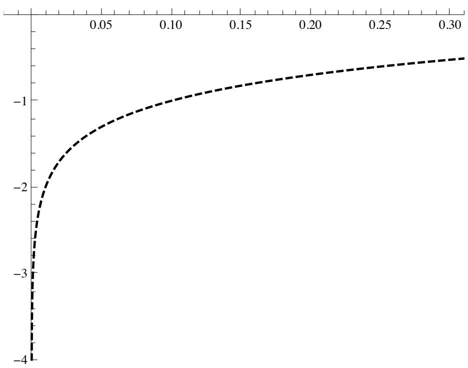




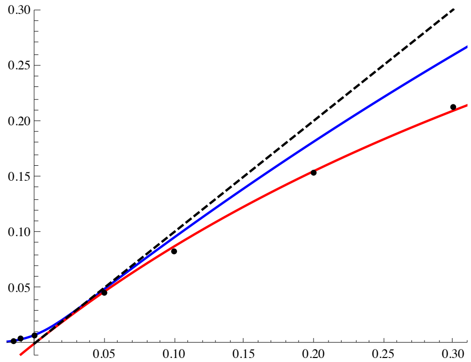
Analytical result (Lineary approximation)

```
FixLinear = (8 * ρ̂H * ρ̂D) / ((ρ̂H + 2 * ρ̂D) * (dR/d̄ * ρ̂H + 2 * dRR/d̄ * ρ̂D)) * saverage;
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.3}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
gLogFixLinear = Plot[Log[10, FixLinear] /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-4, 0}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```





```
Show[gDFfixWS, gBPfix, gFixLinear, gNumerical]
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical]
```



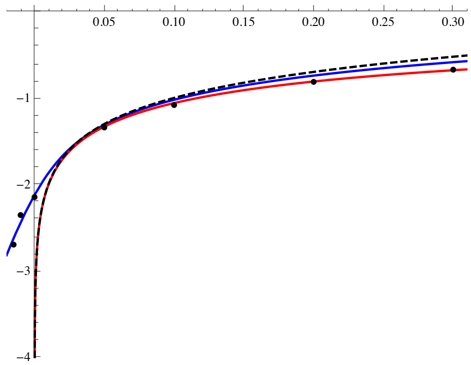
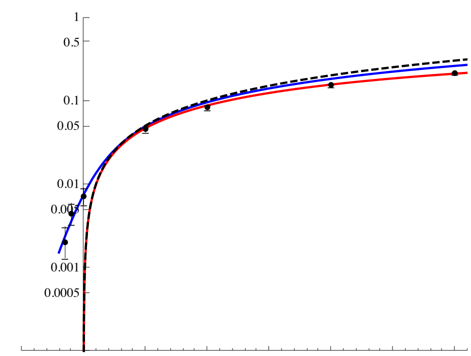


Figure for output

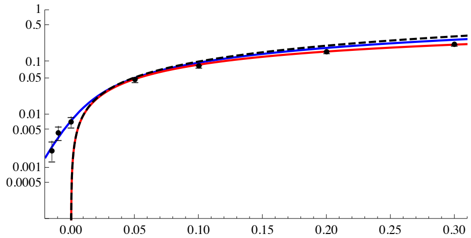
```
axeslabels = {{-5, "0.00001"}, {Log[10, 0.00005], "0.00005"}, {-4, "0.0001"},
   {Log[10, 0.0005], "0.0005"}, {-3, "0.001"}, {Log[10, 0.005], "0.005"},
   {-2, "0.01"}, {Log[10, 0.05], "0.05"}, {-1, "0.1"}, {Log[10, 0.5], "0.5"}, {0, "1"}};
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{-0.05, MaxV}, {-4, 0}},
 AxesOrigin → {0, -4}, Ticks → {Automatic, axeslabels}]
```

$$\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR * \beta R * dR}{2}} + \sqrt{\beta RR * dRR}}$$

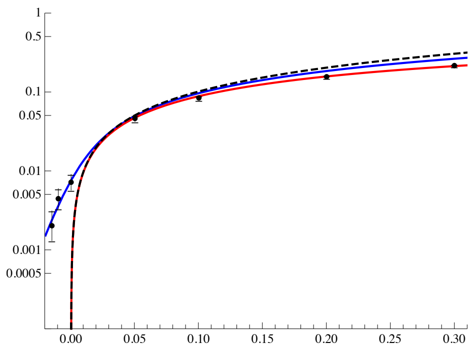


0.666667

```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
 AxesOrigin → {-0.02, -4}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.5]
```




```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
 AxesOrigin → {-0.02, -4}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



- **Fig.1(b)**

Parameter values

```
Clear[x]

Site = 100;
fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 0;
SELβMM = 4 * x;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0.5;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$SEL\beta RM = \frac{\beta RM - \beta RR}{\beta RR};$$

$$SELdRM = \frac{dRR - dRM}{dRR};$$

```
MinV = -0.02;
MaxV = 0.31;
```

$$s_{average} = \frac{1}{2} * \left(\frac{SELfM}{2} + SEL\beta M + SELdM\right) + \frac{1}{2} * (SEL\beta MM * HETERO\beta + SELdM * HETEROd)$$

$$\frac{1}{2}\ (0. + 2.\ x)$$

Out put

```
SampleMax = 10000;
ChangeMax = 6;
OutputData = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulation

```
VectpM = Table[0, {SampleMax}];

Resultparameter = Table[0, {ChangeMax + 1}];
Result = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.015];
  If[change == 2, x = -0.01];
  If[change == 3, x = 0];
  If[change == 4, x = 0.05];
  If[change == 5, x = 0.1];
  If[change == 6, x = 0.2];
```

```
If[change == 7, x = 0.3];

sample = 1;
For[sample = 1, sample ≤ SampleMax, sample = sample + 1,

  EqρH = Sqrt[βRR * dRR] / (Sqrt[(fR*βR*dR)/2] + Sqrt[βRR * dRR]);

  xR = Round[Site * EqρH];
  xM = 0;
  xRR = Site - xR;
  xRM = 0;
  xMM = 0;
  sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];
  xR = xR - sampleMutation[[1]];
  xM = xM + sampleMutation[[1]];
  xRR = xRR - sampleMutation[[2]];
  xRM = xRM + sampleMutation[[2]];
  xMM = 0;

  If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
  If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
  If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

  step = 2;
  While[0 < pM && pM < 1,
   pDeathR = (dR * xR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   pDeathM = (dM * xM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   pDeathRR = (dRR * xRR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   pDeathRM = (dRM * xRM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   pDeathMM = (dMM * xMM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   sampleDeath =
    RandomVariate[MultinomialDistribution[1,
      {pDeathR, pDeathM, pDeathRR, pDeathRM, pDeathMM}]];
   xR = xR - sampleDeath[[1]];
   xM = xM - sampleDeath[[2]];
   xRR = xRR - sampleDeath[[3]];
   xRM = xRM - sampleDeath[[4]];
   xMM = xMM - sampleDeath[[5]];

   cR = βRR * xRR + βRM/2 * xRM;
   cM = βMM * xMM + βRM/2 * xRM;
   If[xR == 0 && xM == 0, cRR = 0];
```

```
      If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (βR^2 * xR^2)/(βR * xR + βM * xM)];
      If[xR == 0 && xM == 0, cRM = 0];
      If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (βR * βM * xR * xM)/(βR * xR + βM * xM)];
      If[xR == 0 && xM == 0, cMM = 0];
      If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (βM^2 * xM^2)/(βR * xR + βM * xM)];
      pCellR = cR/(cR + cM + cRR + cRM + cMM);
      pCellM = cM/(cR + cM + cRR + cRM + cMM);
      pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
      pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
      pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
      sampleBirth =
       RandomVariate[MultinomialDistribution[1,
          {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

      xR = xR + sampleBirth[[1]];
      xM = xM + sampleBirth[[2]];
      xRR = xRR + sampleBirth[[3]];
      xRM = xRM + sampleBirth[[4]];
      xMM = xMM + sampleBirth[[5]];


      If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
      If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
      If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

      step = step + 1;
     ];

    VectpM[[sample]] = pM;

    OutputData[[countData, 1]] = x;
    OutputData[[countData, 2]] = pM;
    countData = countData + 1;

   ];

  Resultparameter[[change]] = x;
  Result[[change]] = Total[VectpM];

  OutputResult[[countResult, 1]] = x;
  OutputResult[[countResult, 2]] = Total[VectpM];
  countResult = countResult + 1;

  Print[change]

 ];
```

1

2

3

4

5

6

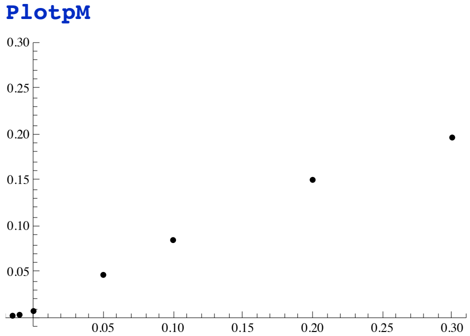
7

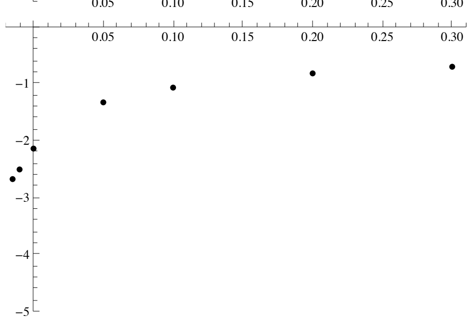
Output of the data

```
Resultparameter
Result
Export["/Users/Bessho/Desktop/workstation/OutputDataFig1b.txt", OutputData, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig1b.txt", OutputResult, "Table"];
{-0.015, -0.01, 0, 0.05, 0.1, 0.2, 0.3}
{21., 31., 72., 466., 845., 1501., 1962.}
```

Plot

```
PlotpM = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM[[change, 1]] = Resultparameter[[change]];
  PlotpM[[change, 2]] = Result[[change]] / SampleMax;
  PlotpMlog[[change, 1]] = Resultparameter[[change]];
  PlotpMlog[[change, 2]] = Log[10, Result[[change]] / SampleMax];
 ];
gNumerical = ListPlot[PlotpM, PlotRange → {{MinV, MaxV}, {-0.01, 0.3}},
  PlotStyle → {Thickness[0.005], Black}]
gLogNumerical = ListPlot[PlotpMlog, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Black}]
PlotpM
```






```
{{-0.015, 0.0021}, {-0.01, 0.0031}, {0, 0.0072},
 {0.05, 0.0466}, {0.1, 0.0845}, {0.2, 0.1501}, {0.3, 0.1962}}
```

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u}=\frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \overline{z}] \sim 0.95$.

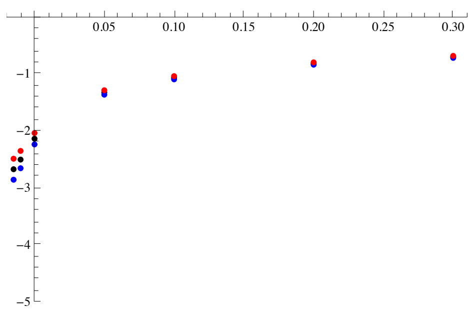
```
z = 1.96;
```

$$\underline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /. \; n \to \text{SampleMax} \; /. \; \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

$$\overline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /. \; n \to \text{SampleMax} \; /. \; \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

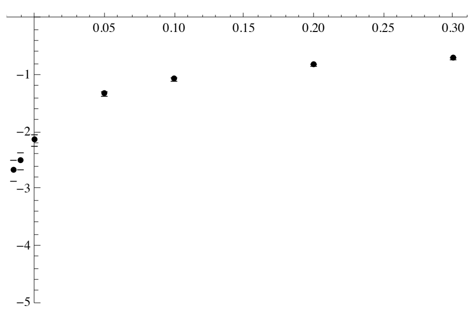
```
{0.00137398, 0.00218485, 0.00572167, 0.0426399, 0.0792068, 0.143234, 0.188534}
{0.00320842, 0.00439678, 0.00905681, 0.0509083, 0.0901123, 0.157235, 0.2041}
```

```
PlotCIunder = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder[[change, 1]] = Resultparameter[[change]];
  PlotCIupper[[change, 1]] = Resultparameter[[change]];
  PlotCIunder[[change, 2]] = Log[10, z̲[[change]]];
  PlotCIupper[[change, 2]] = Log[10, z̄[[change]]];
 ];
gCIupperDot = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Red}];
gCIunderDot = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Blue}];
Show[gLogNumerical, gCIunderDot, gCIupperDot]
```

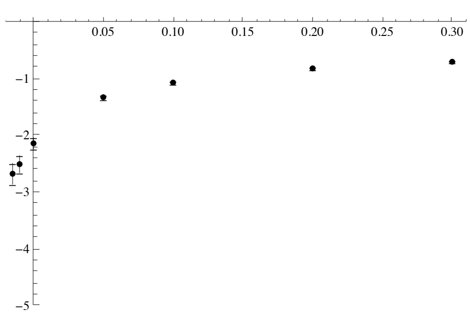



```
Log[10, z̄]
N[Log[10, PlotpM[[All, 2]]]]
Log[10, z̲]
gCIupper = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
Show[gLogNumerical, gCIunder, gCIupper]
```

```
{-2.49371, -2.35687, -2.04302, -1.29321, -1.04522, -0.803451, -0.690158}
{-2.67778, -2.50864, -2.14267, -1.33161, -1.07314, -0.823619, -0.707301}
{-2.86202, -2.66058, -2.24248, -1.37018, -1.10124, -0.843954, -0.724611}
```



```
PlotCI1 = Table[0, {i, 2}, {j, 2}];
PlotCI2 = Table[0, {i, 2}, {j, 2}];
PlotCI3 = Table[0, {i, 2}, {j, 2}];
PlotCI4 = Table[0, {i, 2}, {j, 2}];
PlotCI5 = Table[0, {i, 2}, {j, 2}];
PlotCI6 = Table[0, {i, 2}, {j, 2}];
PlotCI7 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1[[1, 1]] = Resultparameter[[1]];
  PlotCI1[[1, 2]] = Log[10, z̲[[1]]];
  PlotCI1[[2, 1]] = Resultparameter[[1]];
  PlotCI1[[2, 2]] = Log[10, z̅[[1]]];
  PlotCI2[[1, 1]] = Resultparameter[[2]];
  PlotCI2[[1, 2]] = Log[10, z̲[[2]]];
  PlotCI2[[2, 1]] = Resultparameter[[2]];
  PlotCI2[[2, 2]] = Log[10, z̅[[2]]];
  PlotCI3[[1, 1]] = Resultparameter[[3]];
  PlotCI3[[1, 2]] = Log[10, z̲[[3]]];
  PlotCI3[[2, 1]] = Resultparameter[[3]];
  PlotCI3[[2, 2]] = Log[10, z̅[[3]]];
  PlotCI4[[1, 1]] = Resultparameter[[4]];
  PlotCI4[[1, 2]] = Log[10, z̲[[4]]];
  PlotCI4[[2, 1]] = Resultparameter[[4]];
  PlotCI4[[2, 2]] = Log[10, z̅[[4]]];
  PlotCI5[[1, 1]] = Resultparameter[[5]];
  PlotCI5[[1, 2]] = Log[10, z̲[[5]]];
  PlotCI5[[2, 1]] = Resultparameter[[5]];
  PlotCI5[[2, 2]] = Log[10, z̅[[5]]];
  PlotCI6[[1, 1]] = Resultparameter[[6]];
  PlotCI6[[1, 2]] = Log[10, z̲[[6]]];
  PlotCI6[[2, 1]] = Resultparameter[[6]];
  PlotCI6[[2, 2]] = Log[10, z̅[[6]]];
  PlotCI7[[1, 1]] = Resultparameter[[7]];
  PlotCI7[[1, 2]] = Log[10, z̲[[7]]];
  PlotCI7[[2, 1]] = Resultparameter[[7]];
  PlotCI7[[2, 2]] = Log[10, z̅[[7]]];
 ];
gCI1 = ListPlot[PlotCI1, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2 = ListPlot[PlotCI2, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3 = ListPlot[PlotCI3, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4 = ListPlot[PlotCI4, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5 = ListPlot[PlotCI5, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI6 = ListPlot[PlotCI6, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI7 = ListPlot[PlotCI7, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper]
```



Analytical result (Branching process approximation)

```
Clear[y]

fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 0;
SELβMM = 4 * y;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SEL}\beta\text{M} + \text{SELdM}\right) + \frac{1}{2} * (\text{SEL}\beta\text{MM} * \text{HETERO}\beta + \text{SELdM} * \text{HETEROd})$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\beta\text{RR} * \text{dRR}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

$$\text{FaiR} = \frac{\beta\text{R}}{\text{dR}};$$

$$\text{FaiM} = \frac{\beta\text{M}}{\text{dM}};$$

$$\text{FaiRR} = \frac{\beta\text{RR}}{\text{dRR}};$$

$$\text{FaiRM} = \frac{\beta\text{RM}}{\text{dRM}};$$

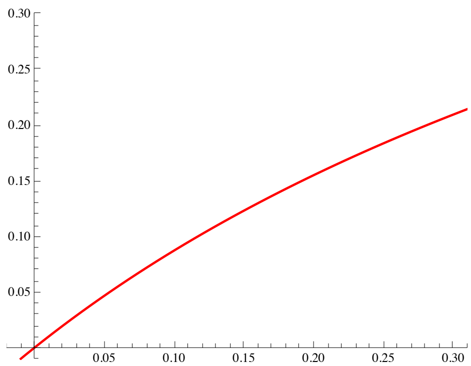
$$\text{avef} = \frac{\text{fM} + \text{fR}}{2};$$

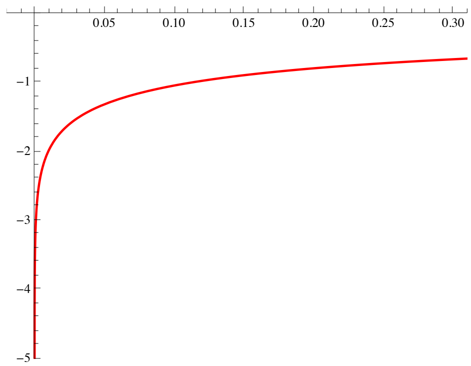
$$\text{FaiR2} = \frac{\text{fR}}{2} * \text{FaiR};$$

$$\text{FaiM2} = \frac{\text{avef}}{2} * \text{FaiM};$$

```
BPPaiH = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + FaiRM/2 * Sqrt[FaiR2 * FaiRR]);
BPPaiD = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + 2 * FaiM2 * Sqrt[FaiR2 * FaiRR]);
BPfix = EqρH / (EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD) / (EqρH + 2 * EqρD) * BPPaiD;
gBPfix = Plot[BPfix, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.3}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gLogBPfix = Plot[Log[10, BPfix], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
```

1. y





Analytical result (Diffusion approximation)

$$\text{Parameters} = \left\{\text{SUM} \to \text{Site},\ d_R \to \text{dR},\ d_{RR} \to \text{dRR},\ \hat{\rho}_H \to \frac{\sqrt{\beta\text{RR} * \text{dRR}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}},\right.$$

$$\hat{\rho}_D \to \frac{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}},\ \text{sfM} \to \text{SELfM},\ \text{s}\beta\text{M} \to \text{SEL}\beta\text{M},\ \text{s}\beta\text{RM} \to \text{SEL}\beta\text{RM},\ \text{s}\beta\text{MM} \to \text{SEL}\beta\text{MM},$$

$$\left.\text{sdM} \to \text{SELdM},\ \text{sdRM} \to \text{SELdRM},\ \text{sdMM} \to \text{SELdMM}\right\}$$

$$\text{Additive} = \text{s}\beta\text{MM} - 2 * \text{s}\beta\text{RM} + \text{sdMM} - 2 * \text{sdRM};$$

$$d_{\text{harmonic}} = \frac{2}{\frac{1}{d_R} + \frac{1}{d_{RR}}};$$

$$\hat{d} = d_R * \hat{\rho}_H + d_{RR} * \left(1 - \hat{\rho}_H\right);$$

$$\bar{d} = \frac{d_R + d_{RR}}{2};$$

$$m = \text{SUM} * p * (1 - p) * \frac{2 * s_{\text{average}} + p * \text{Additive}}{2};$$

$$v = \frac{p * (1 - p) * \left(\frac{d_R}{\hat{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\hat{d}} * \hat{\rho}_D\right)}{4 * \hat{\rho}_H * \hat{\rho}_D};$$

$$Q = \text{Integrate}\left[\frac{m}{v} \text{ /. Parameters},\ p\right];$$

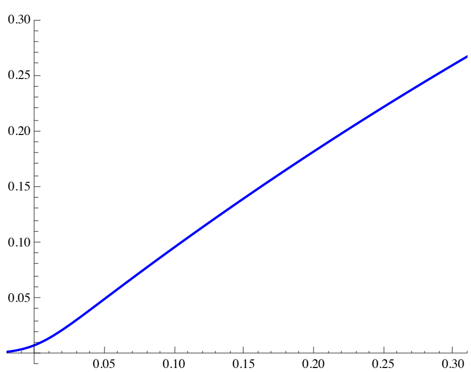
$$\text{INI} = \frac{1}{\text{SUM} * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)} \text{ /. Parameters};$$

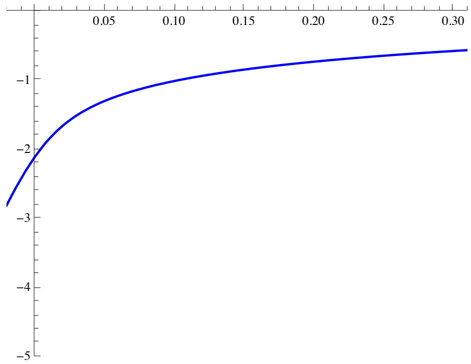
$$\text{DFfixWS} = \frac{\text{Integrate}[\text{Exp}[-2 * Q], \{p, 0, \text{INI}\}]}{\text{Integrate}[\text{Exp}[-2 * Q], \{p, 0, 1\}]};$$

```
gDFfixWS = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.3}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS = Plot[Log[10, DFfixWS], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

$$\left\{\text{SUM} \to 100,\ d_R \to 0.005,\ d_{RR} \to 0.005,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,\ \text{sfM} \to 0,\right.$$

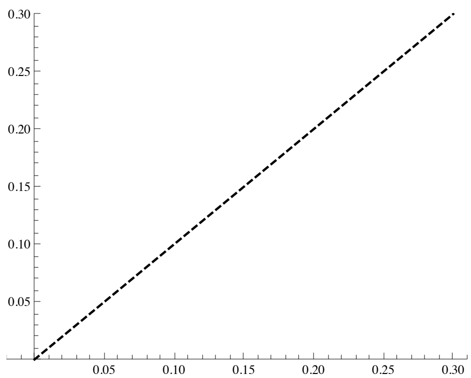
$$\left.\text{s}\beta\text{M} \to 0,\ \text{s}\beta\text{RM} \to \frac{-1000 + 1000\ (1 + 2.\ y)}{1000},\ \text{s}\beta\text{MM} \to 4\ y,\ \text{sdM} \to 0,\ \text{sdRM} \to 0.,\ \text{sdMM} \to 0\right\}$$

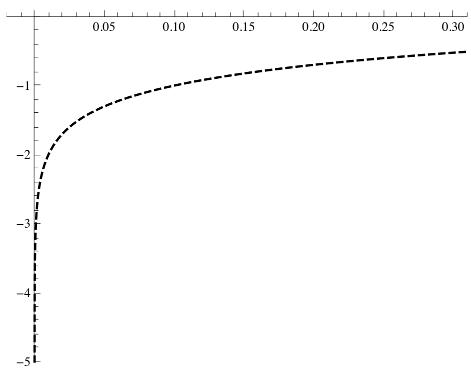




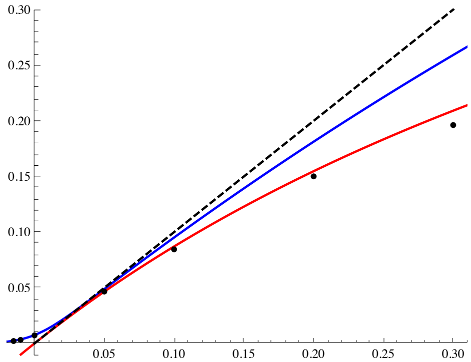
Analytical result (Lineary approximation)

```
FixLinear = (8 * ρ̂H * ρ̂D) / ((ρ̂H + 2 * ρ̂D) * (dR/d̄ * ρ̂H + 2 * dRR/d̄ * ρ̂D)) * saverage;
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.3}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
gLogFixLinear = Plot[Log[10, FixLinear] /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-5, 0}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```





```
Show[gDFfixWS, gBPfix, gFixLinear, gNumerical]
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical]
```



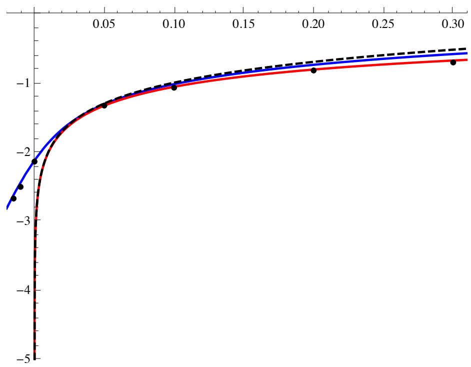
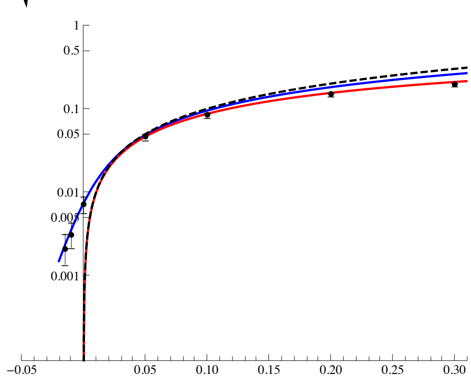


Figure for output

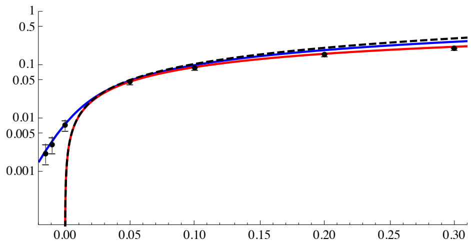
```
axeslabels = {{-3, "0.001"}, {Log[10, 0.005], "0.005"}, {-2, "0.01"},
   {Log[10, 0.05], "0.05"}, {-1, "0.1"}, {Log[10, 0.5], "0.5"}, {0, "1"}};
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{-0.05, MaxV}, {-4, 0}},
 AxesOrigin → {0, -4}, Ticks → {Automatic, axeslabels}]
```

$$\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR * \beta R * dR}{2}} + \sqrt{\beta RR * dRR}}$$

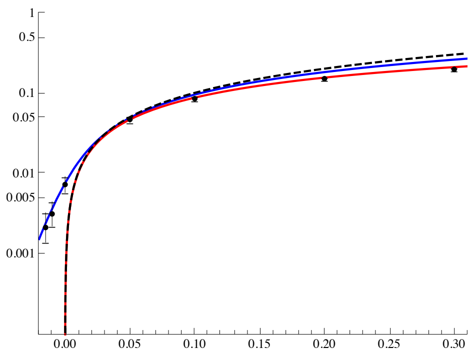


0.666667

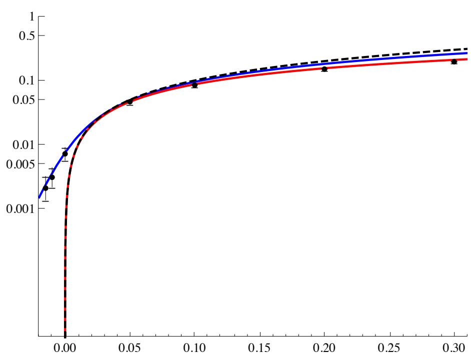
```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
 AxesOrigin → {-0.02, -4}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.5]
```



```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
 AxesOrigin → {-0.02, -4}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
 AxesOrigin → {-0.02, -5}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



- **Fig.1(c)**

Parameter values

```
Clear[x]

Site = 100;
fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 0;
SELβMM = 2 * x;
HETEROβ = 1.0;
SELdM = 0;
SELdMM = 0;
HETEROd = 0.5;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

```
MinV = -0.02;
MaxV = 0.31;
```

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SEL}\beta\text{M} + \text{SELdM}\right) + \frac{1}{2} * (\text{SEL}\beta\text{MM} * \text{HETERO}\beta + \text{SELdM} * \text{HETEROd})$$

$$\frac{1}{2}(0. + 2.\ x)$$

Out put

```
SampleMax = 10 000;
ChangeMax = 6;
OutputData = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulation

```
VectpM = Table[0, {SampleMax}];

Resultparameter = Table[0, {ChangeMax + 1}];
Result = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.015];
  If[change == 2, x = -0.01];
  If[change == 3, x = 0];
  If[change == 4, x = 0.05];
  If[change == 5, x = 0.1];
  If[change == 6, x = 0.2];
```

```
If[change == 7, x = 0.3];

sample = 1;
For[sample = 1, sample ≤ SampleMax, sample = sample + 1,


 EqρH = Sqrt[βRR * dRR] / Sqrt[(fR*βR*dR)/2 + Sqrt[βRR * dRR]];

 xR = Round[Site * EqρH];
 xM = 0;
 xRR = Site - xR;
 xRM = 0;
 xMM = 0;
 sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];
 xR = xR - sampleMutation[[1]];
 xM = xM + sampleMutation[[1]];
 xRR = xRR - sampleMutation[[2]];
 xRM = xRM + sampleMutation[[2]];
 xMM = 0;


 If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
 If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
 If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];


 step = 2;
 While[0 < pM && pM < 1,
  pDeathR = (dR * xR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathM = (dM * xM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathRR = (dRR * xRR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathRM = (dRM * xRM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathMM = (dMM * xMM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  sampleDeath =
   RandomVariate[MultinomialDistribution[1,
     {pDeathR, pDeathM, pDeathRR, pDeathRM, pDeathMM}]];
  xR = xR - sampleDeath[[1]];
  xM = xM - sampleDeath[[2]];
  xRR = xRR - sampleDeath[[3]];
  xRM = xRM - sampleDeath[[4]];
  xMM = xMM - sampleDeath[[5]];


  cR = βRR * xRR + βRM/2 * xRM;
  cM = βMM * xMM + βRM/2 * xRM;
  If[xR == 0 && xM == 0, cRR = 0];
```

```
      If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (βR^2 * xR^2)/(βR * xR + βM * xM)];
      If[xR == 0 && xM == 0, cRM = 0];
      If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (βR * βM * xR * xM)/(βR * xR + βM * xM)];
      If[xR == 0 && xM == 0, cMM = 0];
      If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (βM^2 * xM^2)/(βR * xR + βM * xM)];
      pCellR = cR/(cR + cM + cRR + cRM + cMM);
      pCellM = cM/(cR + cM + cRR + cRM + cMM);
      pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
      pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
      pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
      sampleBirth =
       RandomVariate[MultinomialDistribution[1,
         {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

      xR = xR + sampleBirth[[1]];
      xM = xM + sampleBirth[[2]];
      xRR = xRR + sampleBirth[[3]];
      xRM = xRM + sampleBirth[[4]];
      xMM = xMM + sampleBirth[[5]];

      If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
      If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
      If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

      step = step + 1;
     ];

    VectpM[[sample]] = pM;

    OutputData[[countData, 1]] = x;
    OutputData[[countData, 2]] = pM;
    countData = countData + 1;

   ];

  Resultparameter[[change]] = x;
  Result[[change]] = Total[VectpM];

  OutputResult[[countResult, 1]] = x;
  OutputResult[[countResult, 2]] = Total[VectpM];
  countResult = countResult + 1;

  Print[change]

 ];
```

1

2

3

4

5

6

7

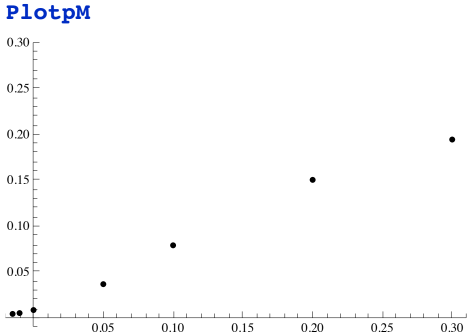
Output of the data

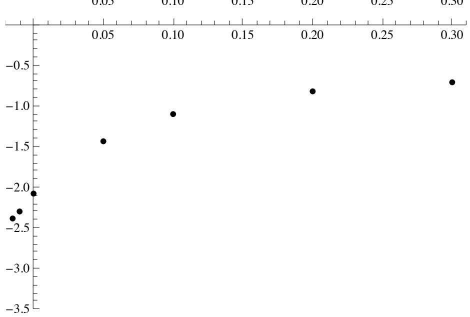
```
Resultparameter
Result
Export["/Users/Bessho/Desktop/workstation/OutputDataFig1c.txt", OutputData, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig1c.txt", OutputResult, "Table"];
```

```
{-0.015, -0.01, 0, 0.05, 0.1, 0.2, 0.3}
{41., 50., 83., 365., 788., 1501., 1940.}
```

Plot

```
PlotpM = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM[[change, 1]] = Resultparameter[[change]];
  PlotpM[[change, 2]] = Result[[change]] / SampleMax;
  PlotpMlog[[change, 1]] = Resultparameter[[change]];
  PlotpMlog[[change, 2]] = Log[10, Result[[change]] / SampleMax];
 ];
gNumerical = ListPlot[PlotpM, PlotRange → {{MinV, MaxV}, {-0.01, 0.3}},
  PlotStyle → {Thickness[0.005], Black}]
gLogNumerical = ListPlot[PlotpMlog, PlotRange → {{MinV, MaxV}, {-3.5, 0}},
  PlotStyle → {Thickness[0.005], Black}]
PlotpM
```





```
{{-0.015, 0.0041}, {-0.01, 0.005}, {0, 0.0083},
 {0.05, 0.0365}, {0.1, 0.0788}, {0.2, 0.1501}, {0.3, 0.194}}
```

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \overline{z}] \sim 0.95$.

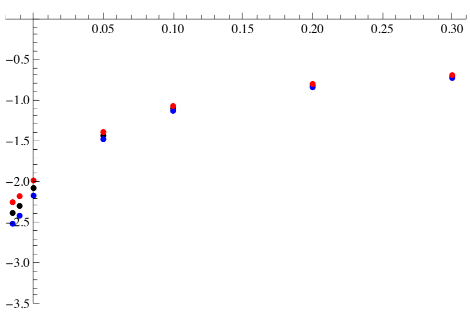
```
z = 1.96;
```

$$\underline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*\left(1-\hat{u}\right) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to \text{SampleMax} \text{ /. } \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

$$\overline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*\left(1-\hat{u}\right) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to \text{SampleMax} \text{ /. } \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

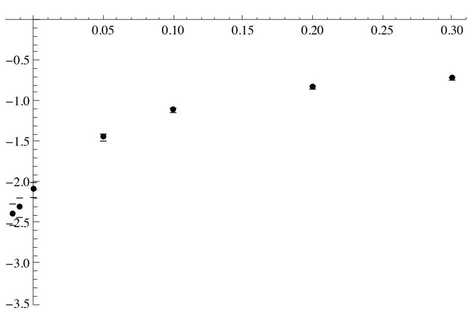
```
{0.00302384, 0.00379488, 0.00670094, 0.0329988, 0.0736795, 0.143234, 0.186368}
{0.00555703, 0.00658529, 0.0102767, 0.0403572, 0.084244, 0.157235, 0.201867}
```

```
PlotCIunder = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder[[change, 1]] = Resultparameter[[change]];
  PlotCIupper[[change, 1]] = Resultparameter[[change]];
  PlotCIunder[[change, 2]] = Log[10, z̲[[change]]];
  PlotCIupper[[change, 2]] = Log[10, z̄[[change]]];
 ];
gCIupperDot = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Red}];
gCIunderDot = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Blue}];
Show[gLogNumerical, gCIunderDot, gCIupperDot]
```

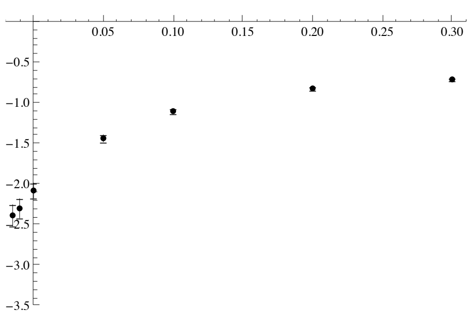



```
Log[10, z̄]
N[Log[10, PlotpM[[All, 2]]]]
Log[10, z̲]
gCIupper = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
Show[gLogNumerical, gCIunder, gCIupper]
```

```
{-2.25516, -2.18143, -1.98815, -1.39408, -1.07446, -0.803451, -0.694934}
{-2.38722, -2.30103, -2.08092, -1.43771, -1.10347, -0.823619, -0.712198}
{-2.51944, -2.4208, -2.17386, -1.4815, -1.13265, -0.843954, -0.729629}
```



```
PlotCI1 = Table[0, {i, 2}, {j, 2}];
PlotCI2 = Table[0, {i, 2}, {j, 2}];
PlotCI3 = Table[0, {i, 2}, {j, 2}];
PlotCI4 = Table[0, {i, 2}, {j, 2}];
PlotCI5 = Table[0, {i, 2}, {j, 2}];
PlotCI6 = Table[0, {i, 2}, {j, 2}];
PlotCI7 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1[[1, 1]] = Resultparameter[[1]];
  PlotCI1[[1, 2]] = Log[10, z̲[[1]]];
  PlotCI1[[2, 1]] = Resultparameter[[1]];
  PlotCI1[[2, 2]] = Log[10, z̄[[1]]];
  PlotCI2[[1, 1]] = Resultparameter[[2]];
  PlotCI2[[1, 2]] = Log[10, z̲[[2]]];
  PlotCI2[[2, 1]] = Resultparameter[[2]];
  PlotCI2[[2, 2]] = Log[10, z̄[[2]]];
  PlotCI3[[1, 1]] = Resultparameter[[3]];
  PlotCI3[[1, 2]] = Log[10, z̲[[3]]];
  PlotCI3[[2, 1]] = Resultparameter[[3]];
  PlotCI3[[2, 2]] = Log[10, z̄[[3]]];
  PlotCI4[[1, 1]] = Resultparameter[[4]];
  PlotCI4[[1, 2]] = Log[10, z̲[[4]]];
  PlotCI4[[2, 1]] = Resultparameter[[4]];
  PlotCI4[[2, 2]] = Log[10, z̄[[4]]];
  PlotCI5[[1, 1]] = Resultparameter[[5]];
  PlotCI5[[1, 2]] = Log[10, z̲[[5]]];
  PlotCI5[[2, 1]] = Resultparameter[[5]];
  PlotCI5[[2, 2]] = Log[10, z̄[[5]]];
  PlotCI6[[1, 1]] = Resultparameter[[6]];
  PlotCI6[[1, 2]] = Log[10, z̲[[6]]];
  PlotCI6[[2, 1]] = Resultparameter[[6]];
  PlotCI6[[2, 2]] = Log[10, z̄[[6]]];
  PlotCI7[[1, 1]] = Resultparameter[[7]];
  PlotCI7[[1, 2]] = Log[10, z̲[[7]]];
  PlotCI7[[2, 1]] = Resultparameter[[7]];
  PlotCI7[[2, 2]] = Log[10, z̄[[7]]];
 ];
gCI1 = ListPlot[PlotCI1, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2 = ListPlot[PlotCI2, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3 = ListPlot[PlotCI3, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4 = ListPlot[PlotCI4, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5 = ListPlot[PlotCI5, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI6 = ListPlot[PlotCI6, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI7 = ListPlot[PlotCI7, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper]
```



Analytical result (Branching process approximation)

```
Clear[y]

fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 0;
SELβMM = 2 * y;
HETEROβ = 1.0;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SEL}\beta\text{M} + \text{SELdM}\right) + \frac{1}{2} * (\text{SEL}\beta\text{MM} * \text{HETERO}\beta + \text{SELdM} * \text{HETEROd})$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\beta\text{RR} * \text{dRR}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

$$\text{FaiR} = \frac{\beta\text{R}}{\text{dR}};$$

$$\text{FaiM} = \frac{\beta\text{M}}{\text{dM}};$$

$$\text{FaiRR} = \frac{\beta\text{RR}}{\text{dRR}};$$

$$\text{FaiRM} = \frac{\beta\text{RM}}{\text{dRM}};$$

$$\text{avef} = \frac{\text{fM} + \text{fR}}{2};$$

$$\text{FaiR2} = \frac{\text{fR}}{2} * \text{FaiR};$$

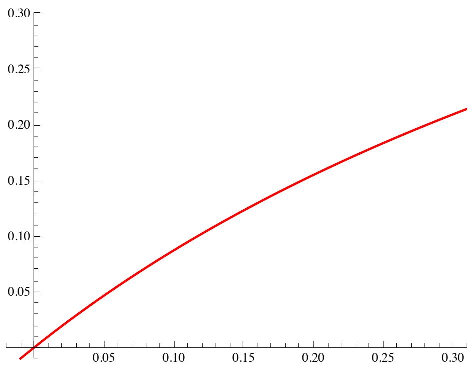
$$\text{FaiM2} = \frac{\text{avef}}{2} * \text{FaiM};$$

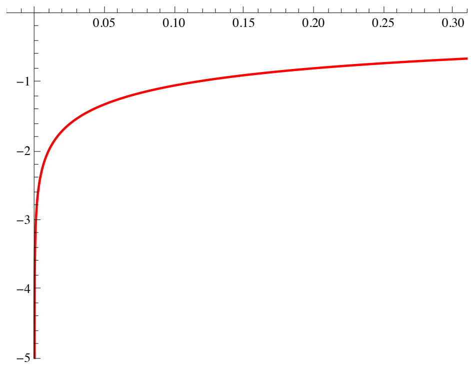
$$\text{BPPaiH} = \frac{\text{FaiM2} * \text{FaiRM} - \text{FaiR2} * \text{FaiRR}}{\text{FaiM2} * \text{FaiRM} + \frac{\text{FaiRM}}{2} * \sqrt{\text{FaiR2} * \text{FaiRR}}};$$

$$\text{BPPaiD} = \frac{\text{FaiM2} * \text{FaiRM} - \text{FaiR2} * \text{FaiRR}}{\text{FaiM2} * \text{FaiRM} + 2 * \text{FaiM2} * \sqrt{\text{FaiR2} * \text{FaiRR}}};$$

$$\text{BPfix} = \frac{\text{Eq}\rho\text{H}}{\text{Eq}\rho\text{H} + 2 * \text{Eq}\rho\text{D}} * \text{BPPaiH} + \frac{2 * \text{Eq}\rho\text{D}}{\text{Eq}\rho\text{H} + 2 * \text{Eq}\rho\text{D}} * \text{BPPaiD};$$

```
gBPfix = Plot[BPfix, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.3}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gLogBPfix = Plot[Log[10, BPfix], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
1. y
```





Analytical result (Diffusion approximation)

$$\text{Parameters} = \left\{\text{SUM} \to \text{Site},\ d_R \to \text{dR},\ d_{RR} \to \text{dRR},\ \hat{\rho}_H \to \frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}},\right.$$

$$\hat{\rho}_D \to \frac{\sqrt{\frac{fR*\beta R*dR}{2}}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}},\ sfM \to SELfM,\ s\beta M \to SEL\beta M,\ s\beta RM \to SEL\beta RM,\ s\beta MM \to SEL\beta MM,$$

$$\left. sdM \to SELdM,\ sdRM \to SELdRM,\ sdMM \to SELdMM \right\}$$

$$\text{Additive} = s\beta MM - 2 * s\beta RM + sdMM - 2 * sdRM;$$

$$d_{harmonic} = \frac{2}{\frac{1}{d_R} + \frac{1}{d_{RR}}};$$

$$\hat{d} = d_R * \hat{\rho}_H + d_{RR} * \left(1 - \hat{\rho}_H\right);$$

$$\overline{d} = \frac{d_R + d_{RR}}{2};$$

$$m = \text{SUM} * p * (1 - p) * \frac{2 * s_{average} + p * \text{Additive}}{2};$$

$$v = \frac{p * (1 - p) * \left(\frac{d_R}{\hat{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\hat{d}} * \hat{\rho}_D\right)}{4 * \hat{\rho}_H * \hat{\rho}_D};$$

$$Q = \text{Integrate}\left[\frac{m}{v}\ /.\ \text{Parameters},\ p\right];$$

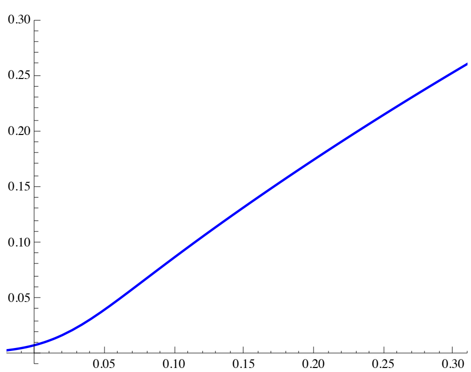
$$\text{INI} = \frac{1}{\text{SUM} * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)}\ /.\ \text{Parameters};$$

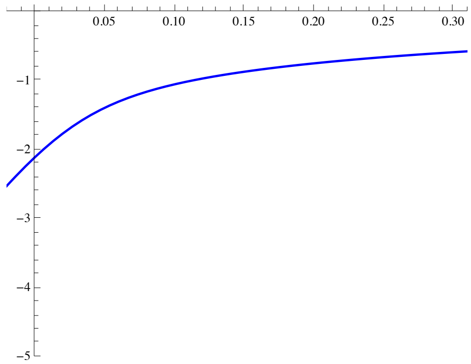
$$\text{DFfixWS} = \frac{\text{Integrate}[\text{Exp}[-2 * Q],\ \{p,\ 0,\ \text{INI}\}]}{\text{Integrate}[\text{Exp}[-2 * Q],\ \{p,\ 0,\ 1\}]};$$

```
gDFfixWS = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.3}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS = Plot[Log[10, DFfixWS], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

$$\left\{\text{SUM} \to 100,\ d_R \to 0.005,\ d_{RR} \to 0.005,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,\ sfM \to 0,\right.$$

$$\left. s\beta M \to 0,\ s\beta RM \to \frac{-1000 + 1000\ (1 + 2.\ y)}{1000},\ s\beta MM \to 2\ y,\ sdM \to 0,\ sdRM \to 0.,\ sdMM \to 0\right\}$$

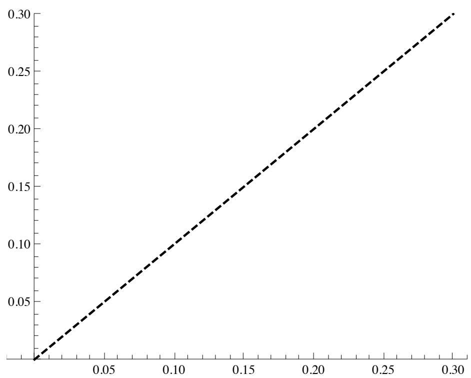


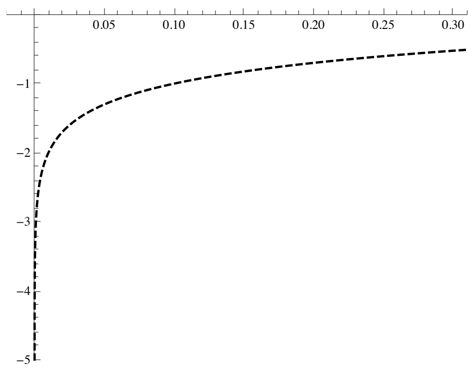


Analytical result (Lineary approximation)

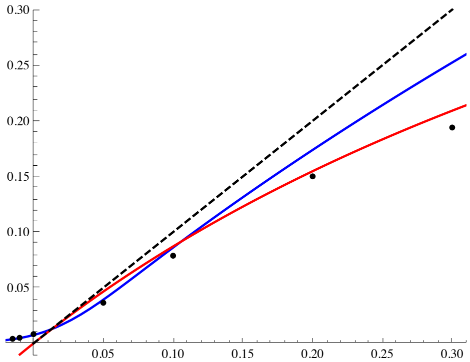
$$\texttt{FixLinear} = \frac{8 * \hat{\rho}_H * \hat{\rho}_D}{\left(\hat{\rho}_H + 2 * \hat{\rho}_D\right) * \left(\frac{d_R}{\bar{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\bar{d}} * \hat{\rho}_D\right)} * s_{average};$$

```
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.3}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
gLogFixLinear = Plot[Log[10, FixLinear] /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-5, 0}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```





```
Show[gDFfixWS, gBPfix, gFixLinear, gNumerical]
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical]
```



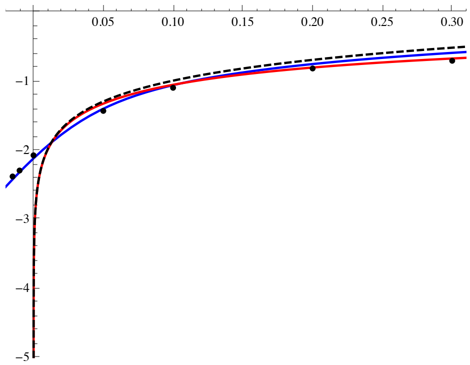
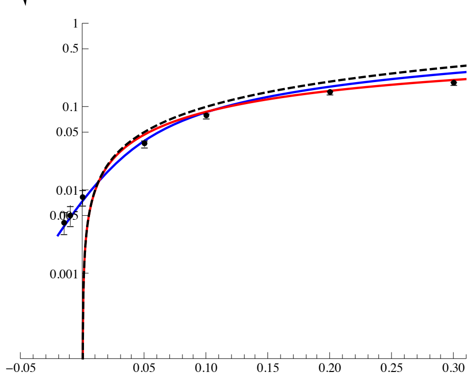


Figure for output

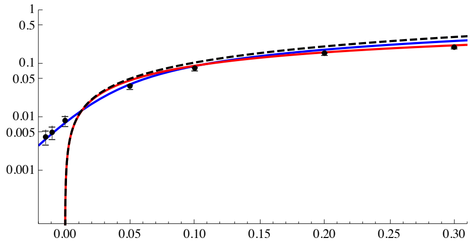
```
axeslabels = {{-3, "0.001"}, {Log[10, 0.005], "0.005"}, {-2, "0.01"},
   {Log[10, 0.05], "0.05"}, {-1, "0.1"}, {Log[10, 0.5], "0.5"}, {0, "1"}};
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{-0.05, MaxV}, {-4, 0}},
 AxesOrigin → {0, -4}, Ticks → {Automatic, axeslabels}]
```

$$\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR * \beta R * dR}{2}} + \sqrt{\beta RR * dRR}}$$

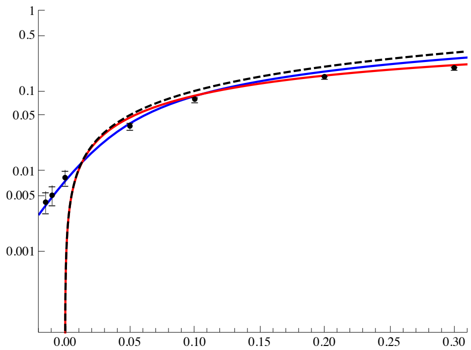



```
0.666667
```

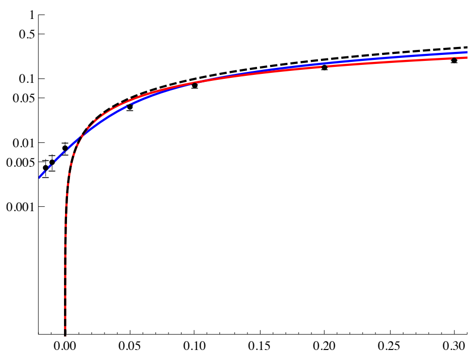
```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
 AxesOrigin → {-0.02, -4}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.5]
```




```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
 AxesOrigin → {-0.02, -4}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```




```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
 AxesOrigin → {-0.02, -5}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



- **Fig.1(d)**

Parameter values

```
Clear[x]

Site = 100;

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 2 * x;
SELwMM = 0;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\mathrm{SELwRM} = \frac{\mathrm{wRM} - \mathrm{wRR}}{\mathrm{wRR}};$$

```
MinV = -0.02;
MaxV = 0.31;
```

$$s_{\mathrm{average}} = \frac{1}{2} * \left(\frac{\mathrm{SELfM}}{2} + \mathrm{SELwM}\right) + \frac{1}{2} * (\mathrm{SELwMM} * \mathrm{HETEROw})$$

```
0. + x
```

Out put

```
SampleMax = 10000;
ChangeMax = 6;
OutputData = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulaton

```
VectpM = Table[0, {SampleMax}];

Resultparameter = Table[0, {ChangeMax + 1}];
Result = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.015];
  If[change == 2, x = -0.01];
  If[change == 3, x = 0];
  If[change == 4, x = 0.05];
  If[change == 5, x = 0.1];
  If[change == 6, x = 0.2];
  If[change == 7, x = 0.3];

  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,
```

$$\mathrm{Eq}\rho\mathrm{H} = \frac{\sqrt{\mathrm{wRR}}}{\sqrt{\frac{\mathrm{fR} * \mathrm{wR}}{2}} + \sqrt{\mathrm{wRR}}};$$

```
   xR = Round[Site * EqρH];
```

```
xM = 0;
xRR = Site - xR;
xRM = 0;
xMM = 0;

sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];
xR = xR - sampleMutation[[1]];
xM = xM + sampleMutation[[1]];
xRR = xRR - sampleMutation[[2]];
xRM = xRM + sampleMutation[[2]];
xMM = 0;

If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

step = 2;
While[0 < pM && pM < 1,
 cR = wRR * xRR + wRM/2 * xRM;
 cM = wMM * xMM + wRM/2 * xRM;
 If[xR == 0 && xM == 0, cRR = 0];
 If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (wR^2 * xR^2)/(wR * xR + wM * xM)];
 If[xR == 0 && xM == 0, cRM = 0];
 If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (wR * wM * xR * xM)/(wR * xR + wM * xM)];
 If[xR == 0 && xM == 0, cMM = 0];
 If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (wM^2 * xM^2)/(wR * xR + wM * xM)];
 pCellR = cR/(cR + cM + cRR + cRM + cMM);
 pCellM = cM/(cR + cM + cRR + cRM + cMM);
 pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
 pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
 pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
 sampleBirth = RandomVariate[MultinomialDistribution[Site,
    {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

 xR = sampleBirth[[1]];
 xM = sampleBirth[[2]];
 xRR = sampleBirth[[3]];
 xRM = sampleBirth[[4]];
 xMM = sampleBirth[[5]];
 If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
```

```
      If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];

      If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

      step = step + 1;
     ];

    VectpM[[sample]] = pM;

    OutputData[[countData, 1]] = x;
    OutputData[[countData, 2]] = pM;
    countData = countData + 1;

   ];

   Resultparameter[[change]] = x;
   Result[[change]] = Total[VectpM];

   OutputResult[[countResult, 1]] = x;
   OutputResult[[countResult, 2]] = Total[VectpM];
   countResult = countResult + 1;

   Print[change]

  ];
```

1

2

3

4

5

6

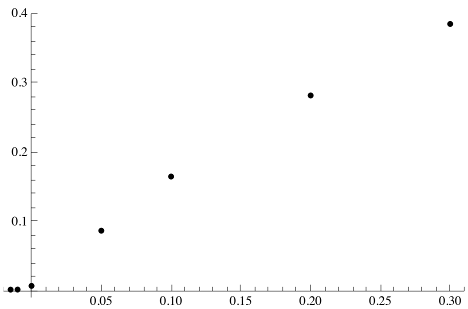
7

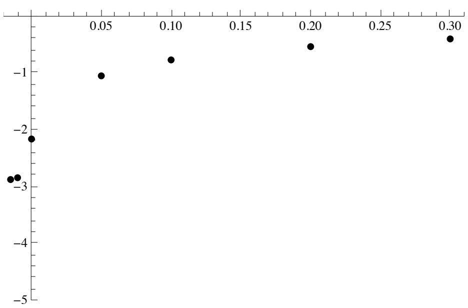
Output of the data

```
Resultparameter
Result
Export["/Users/Bessho/Desktop/workstation/OutputDataFig1d.txt", OutputData, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig1d.txt", OutputResult, "Table"];
{-0.015, -0.01, 0, 0.05, 0.1, 0.2, 0.3}
{13, 14, 67, 862, 1642, 2810, 3840}
```

Plot

```
PlotpM = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM[[change, 1]] = Resultparameter[[change]];
  PlotpM[[change, 2]] = Result[[change]] / SampleMax;
  PlotpMlog[[change, 1]] = Resultparameter[[change]];
  PlotpMlog[[change, 2]] = Log[10, Result[[change]] / SampleMax];
 ];
gNumerical = ListPlot[PlotpM, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Black}]
gLogNumerical = ListPlot[PlotpMlog, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {PointSize[0.015], Black}]
PlotpM
```





$$\left\{\left\{-0.015, \frac{13}{10\,000}\right\}, \left\{-0.01, \frac{7}{5000}\right\}, \left\{0, \frac{67}{10\,000}\right\}, \left\{0.05, \frac{431}{5000}\right\}, \left\{0.1, \frac{821}{5000}\right\}, \left\{0.2, \frac{281}{1000}\right\}, \left\{0.3, \frac{48}{125}\right\}\right\}$$

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, Pr[$\underline{z}$ < u < $\overline{z}$] ~ 0.95.

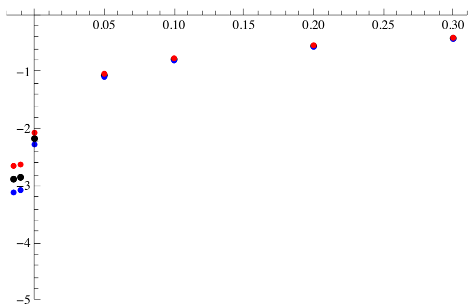
```
z = 1.96;
```

$$\underline{z} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) \;/.\; n \to \text{SampleMax} \;/.\; \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

$$\overline{z} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) \;/.\; n \to \text{SampleMax} \;/.\; \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

```
{0.000759905, 0.000834154, 0.00527961, 0.0808567, 0.157068, 0.272275, 0.374514}
{0.00222311, 0.00234878, 0.00849926, 0.0918611, 0.17159, 0.289893, 0.393575}
```

```
PlotCIunder = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder[[change, 1]] = Resultparameter[[change]];
  PlotCIupper[[change, 1]] = Resultparameter[[change]];
  PlotCIunder[[change, 2]] = Log[10, z[[change]]];
  PlotCIupper[[change, 2]] = Log[10, z̄[[change]]];
 ];
gCIupperDot = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Red}];
gCIunderDot = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Blue}];
Show[gLogNumerical, gCIunderDot, gCIupperDot]
```

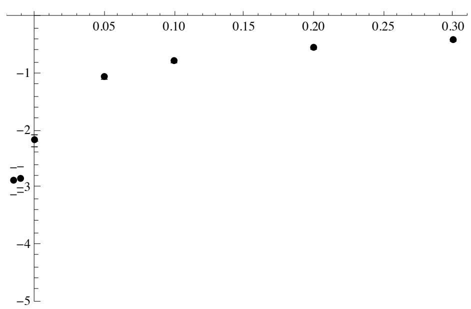


```
Log[10, z̄]
N[Log[10, PlotpM[[All, 2]]]]
Log[10, z]
gCIupper = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
Show[gLogNumerical, gCIunder, gCIupper]
```

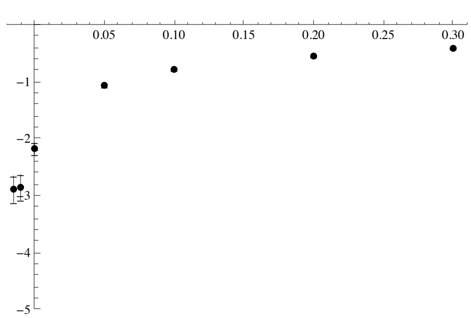
{-2.65304, -2.62916, -2.07062, -1.03687, -0.765509, -0.537763, -0.404972}

{-2.88606, -2.85387, -2.17393, -1.06449, -0.784627, -0.551294, -0.415669}

{-3.11924, -3.07875, -2.2774, -1.09228, -0.803912, -0.564992, -0.426532}



```
PlotCI1 = Table[0, {i, 2}, {j, 2}];
PlotCI2 = Table[0, {i, 2}, {j, 2}];
PlotCI3 = Table[0, {i, 2}, {j, 2}];
PlotCI4 = Table[0, {i, 2}, {j, 2}];
PlotCI5 = Table[0, {i, 2}, {j, 2}];
PlotCI6 = Table[0, {i, 2}, {j, 2}];
PlotCI7 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1[[1, 1]] = Resultparameter[[1]];
  PlotCI1[[1, 2]] = Log[10, z̲[[1]]];
  PlotCI1[[2, 1]] = Resultparameter[[1]];
  PlotCI1[[2, 2]] = Log[10, z̄[[1]]];
  PlotCI2[[1, 1]] = Resultparameter[[2]];
  PlotCI2[[1, 2]] = Log[10, z̲[[2]]];
  PlotCI2[[2, 1]] = Resultparameter[[2]];
  PlotCI2[[2, 2]] = Log[10, z̄[[2]]];
  PlotCI3[[1, 1]] = Resultparameter[[3]];
  PlotCI3[[1, 2]] = Log[10, z̲[[3]]];
  PlotCI3[[2, 1]] = Resultparameter[[3]];
  PlotCI3[[2, 2]] = Log[10, z̄[[3]]];
  PlotCI4[[1, 1]] = Resultparameter[[4]];
  PlotCI4[[1, 2]] = Log[10, z̲[[4]]];
  PlotCI4[[2, 1]] = Resultparameter[[4]];
  PlotCI4[[2, 2]] = Log[10, z̄[[4]]];
  PlotCI5[[1, 1]] = Resultparameter[[5]];
  PlotCI5[[1, 2]] = Log[10, z̲[[5]]];
  PlotCI5[[2, 1]] = Resultparameter[[5]];
  PlotCI5[[2, 2]] = Log[10, z̄[[5]]];
  PlotCI6[[1, 1]] = Resultparameter[[6]];
  PlotCI6[[1, 2]] = Log[10, z̲[[6]]];
  PlotCI6[[2, 1]] = Resultparameter[[6]];
  PlotCI6[[2, 2]] = Log[10, z̄[[6]]];
  PlotCI7[[1, 1]] = Resultparameter[[7]];
  PlotCI7[[1, 2]] = Log[10, z̲[[7]]];
  PlotCI7[[2, 1]] = Resultparameter[[7]];
  PlotCI7[[2, 2]] = Log[10, z̄[[7]]];
 ];
gCI1 = ListPlot[PlotCI1, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2 = ListPlot[PlotCI2, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3 = ListPlot[PlotCI3, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4 = ListPlot[PlotCI4, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5 = ListPlot[PlotCI5, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI6 = ListPlot[PlotCI6, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI7 = ListPlot[PlotCI7, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper]
```



Fixation probability from branching process approximation

```
Clear[y]

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 2 * y;
SELwMM = 0.0;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\text{SELwRM} = \frac{\text{wRM} - \text{wRR}}{\text{wRR}};$$

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SELwM}\right) + \frac{1}{2} * (\text{SELwMM} * \text{HETEROw})$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR}*\text{wR}}{2}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{avef} = \frac{\text{fM} + \text{fR}}{2};$$

$$\text{W} = \frac{\text{fR} * \text{wR}}{2} * \text{Eq}\rho\text{H} + \text{wRR} * \text{Eq}\rho\text{D};$$

$$\text{EqD4a} = 1 - \text{PH} - \text{Exp}\left[-\text{PD} * \frac{\text{avef} * \text{wM}}{\text{W}}\right];$$

$$\text{EqD4b} = 1 - \text{PD} - \text{Exp}\left[-\text{PH} * \frac{\text{wRM}}{2\,\text{W}}\right];$$

```
RepeatMax = 200;
ResultH = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
ResultD = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
ResultBPfix = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
LogResultBPfix = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
repeat = 0;
For[repeat1 = 1, repeat1 ≤ RepeatMax + 1, repeat1 = repeat1 + 1,
  repeat = repeat + 1;
  x1 = 0 + (MaxV - 0)/RepeatMax * (repeat1 - 1);
  CondH = EqD4a /. y → x1;
  CondD = EqD4b /. y → x1;
  sol = Solve[CondH == 0 && CondD == 0 && 0 < PH && PH < 1 && 0 < PD && PD < 1, {PH, PD}, Reals];
  BPPaiH = Part[(PH /. sol), 1];
  BPPaiD = Part[(PD /. sol), 1];
  ResultH[[repeat, 1]] = x1;
  ResultD[[repeat, 1]] = x1;
```

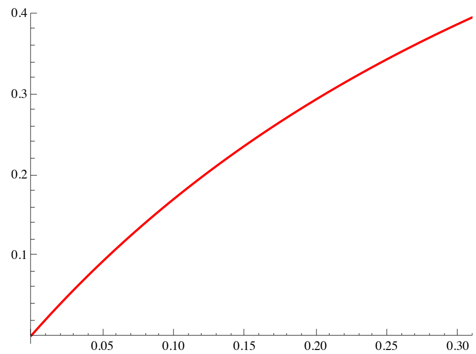
```
    ResultBPfix[[repeat, 1]] = x1;
    LogResultBPfix[[repeat, 1]] = x1;
    ResultH[[repeat, 2]] = BPPaiH;
    ResultD[[repeat, 2]] = BPPaiD;
    ResultBPfix[[repeat, 2]] = EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD;
    LogResultBPfix[[repeat, 2]] = Log[10, EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD];
  ];
gBPfix = ListPlot[ResultBPfix, Joined → True, PlotRange → {{0, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gLogBPfix = ListPlot[LogResultBPfix, Joined → True, PlotRange → {{0, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
0. + y
```

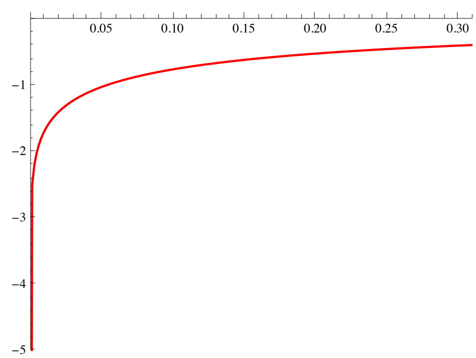
Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

General::stop : この計算中に，Solve::ratnzのこれ以上の出力は表示されません． ≫

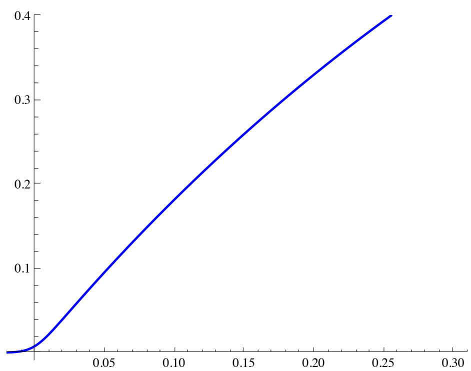


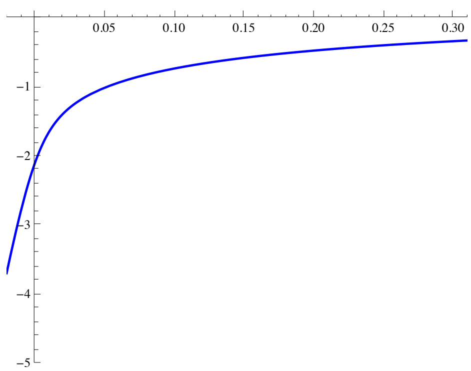


Fixation probability from diffusion approximation

```
Parameters = {SUM → Site, ρ̂_H → √wRR / (√(fR*wR/2) + √wRR), ρ̂_D → √(fR*wR/2) / (√(fR*wR/2) + √wRR), sfM → SELfM,
   swM → SELwM, swRM → SELwRM, swMM → SELwMM }
Additive = swMM - 2 * swRM;
m = SUM * p * (1 - p) * (2 * s_average + p * Additive) / 2;
v = p * (1 - p) * (ρ̂_H + 2 * ρ̂_D) / (8 * ρ̂_H * ρ̂_D);
Q = Integrate[m/v /. Parameters, p];
INI = 1 / (SUM * (ρ̂_H + 2 * ρ̂_D)) /. Parameters;
DFfixWS = Integrate[Exp[-2 * Q], {p, 0, INI}] / Integrate[Exp[-2 * Q], {p, 0, 1}];
gDFfixWS = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
   PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS = Plot[Log[10, DFfixWS], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

{SUM → 100, ρ̂_H → 0.666667, ρ̂_D → 0.333333, sfM → 0, swM → 2 y, swRM → 0., swMM → 0.}

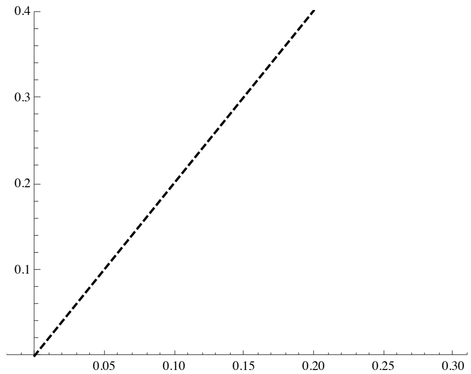


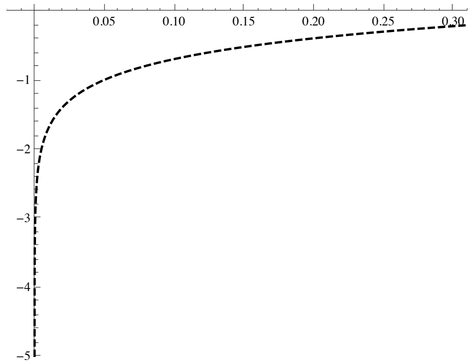


Analytical result (Lineary approximation)

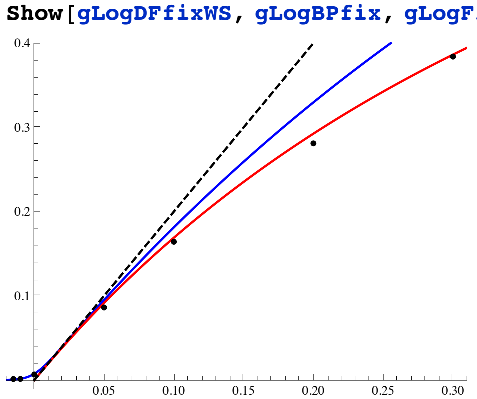
$$\texttt{FixLinear} = \frac{8 * \hat{\rho}_H * \hat{\rho}_D}{\left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)^2} * 2 * s_{average};$$

```
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.4}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
gLogFixLinear = Plot[Log[10, FixLinear /. Parameters], {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-5, 0}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```






```
Show[gDFfixWS, gBPfix, gFixLinear, gNumerical]
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical]
```



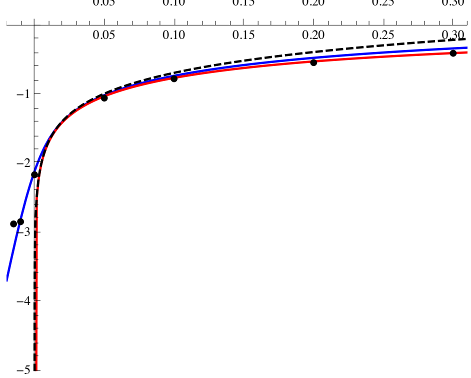
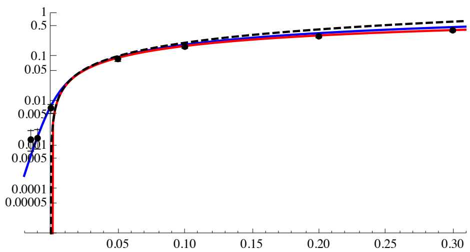


Figure for output

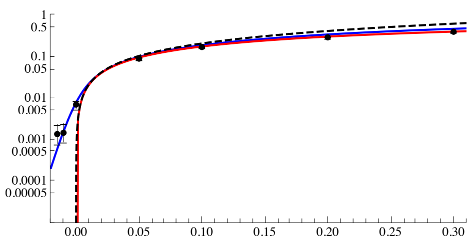
```
axeslabels = {{-5, "0.00001"}, {Log[10, 0.00005], "0.00005"}, {-4, "0.0001"},
   {Log[10, 0.0005], "0.0005"}, {-3, "0.001"}, {Log[10, 0.005], "0.005"},
   {-2, "0.01"}, {Log[10, 0.05], "0.05"}, {-1, "0.1"}, {Log[10, 0.5], "0.5"}, {0, "1"}};
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
 AxesOrigin → {0, -5}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.5]
```

$$\frac{\sqrt{\mathrm{wRR}}}{\sqrt{\frac{\mathrm{fR}*\mathrm{wR}}{2}} + \sqrt{\mathrm{wRR}}}$$

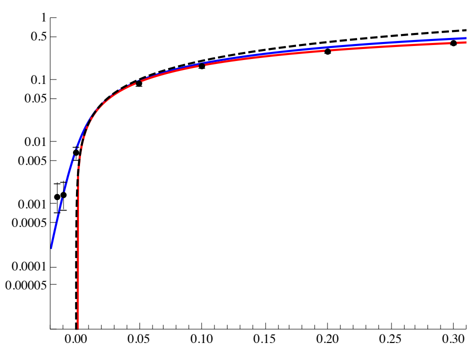


0.666667

```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
 AxesOrigin → {-0.02, -5}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.5]
```




```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-5.0, 0}},
 AxesOrigin → {-0.02, -5.0}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



- **Fig.1(e)**

Parameter values

```
Clear[x]

Site = 100;

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 0;
SELwMM = 4 * x;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);

SELwRM = (wRM - wRR)/wRR;

MinV = -0.02;
MaxV = 0.31;

saverage = 1/2 * (SELfM/2 + SELwM) + 1/2 * (SELwMM * HETEROw)
1. x
```

Out put

```
SampleMax = 10 000;
ChangeMax = 6;
OutputData = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulaton

```
VectpM = Table[0, {SampleMax}];

Resultparameter = Table[0, {ChangeMax + 1}];
Result = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.015];
  If[change == 2, x = -0.01];
  If[change == 3, x = 0];
  If[change == 4, x = 0.05];
  If[change == 5, x = 0.1];
  If[change == 6, x = 0.2];
  If[change == 7, x = 0.3];

  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,
```

```
EqρH = Sqrt[wRR] / Sqrt[Sqrt[fR*wR/2] + Sqrt[wRR]];

xR = Round[Site * EqρH];
xM = 0;
xRR = Site - xR;
xRM = 0;
xMM = 0;

sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];
xR = xR - sampleMutation[[1]];
xM = xM + sampleMutation[[1]];
xRR = xRR - sampleMutation[[2]];
xRM = xRM + sampleMutation[[2]];
xMM = 0;

If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

step = 2;
While[0 < pM && pM < 1,
 cR = wRR * xRR + wRM/2 * xRM;
 cM = wMM * xMM + wRM/2 * xRM;
 If[xR == 0 && xM == 0, cRR = 0];
 If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (wR^2 * xR^2)/(wR * xR + wM * xM)];
 If[xR == 0 && xM == 0, cRM = 0];
 If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (wR * wM * xR * xM)/(wR * xR + wM * xM)];
 If[xR == 0 && xM == 0, cMM = 0];
 If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (wM^2 * xM^2)/(wR * xR + wM * xM)];
 pCellR = cR/(cR + cM + cRR + cRM + cMM);
 pCellM = cM/(cR + cM + cRR + cRM + cMM);
 pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
 pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
 pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
 sampleBirth = RandomVariate[MultinomialDistribution[Site,
    {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

 xR = sampleBirth[[1]];
 xM = sampleBirth[[2]];
 xRR = sampleBirth[[3]];
 xRM = sampleBirth[[4]];
```

```
        xMM = sampleBirth[[5]];
        If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
        If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
        If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

        step = step + 1;
       ];

      VectpM[[sample]] = pM;

      OutputData[[countData, 1]] = x;
      OutputData[[countData, 2]] = pM;
      countData = countData + 1;

     ];

    Resultparameter[[change]] = x;
    Result[[change]] = Total[VectpM];

    OutputResult[[countResult, 1]] = x;
    OutputResult[[countResult, 2]] = Total[VectpM];
    countResult = countResult + 1;

    Print[change]

   ];
```

1

2

3

4

5

6

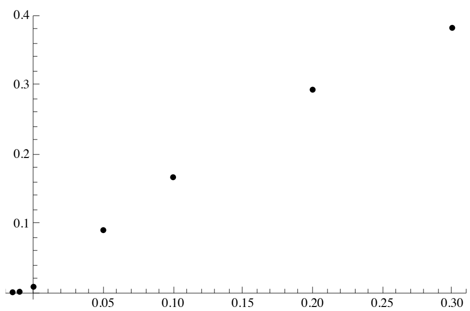
7

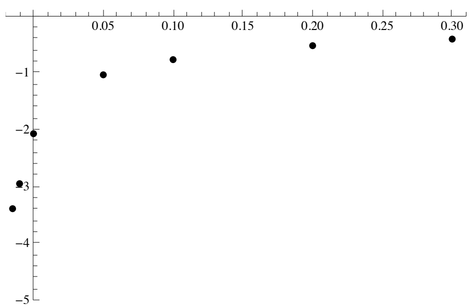
Output of the data

```
Resultparameter
Result
Export["/Users/Bessho/Desktop/workstation/OutputDataFig1e.txt", OutputData, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig1e.txt", OutputResult, "Table"];
{-0.015, -0.01, 0, 0.05, 0.1, 0.2, 0.3}
{4, 11, 83, 899, 1661, 2922, 3813}
```

Plot

```
PlotpM = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM[[change, 1]] = Resultparameter[[change]];
  PlotpM[[change, 2]] = Result[[change]] / SampleMax;
  PlotpMlog[[change, 1]] = Resultparameter[[change]];
  PlotpMlog[[change, 2]] = Log[10, Result[[change]] / SampleMax];
 ];
gNumerical = ListPlot[PlotpM, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Black}]
gLogNumerical = ListPlot[PlotpMlog, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {PointSize[0.015], Black}]
PlotpM
```





$$\left\{\left\{-0.015, \frac{1}{2500}\right\}, \left\{-0.01, \frac{11}{10\,000}\right\}, \left\{0, \frac{83}{10\,000}\right\}, \left\{0.05, \frac{899}{10\,000}\right\}, \left\{0.1, \frac{1661}{10\,000}\right\}, \left\{0.2, \frac{1461}{5000}\right\}, \left\{0.3, \frac{3813}{10\,000}\right\}\right\}$$

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \overline{z}] \sim 0.95$.

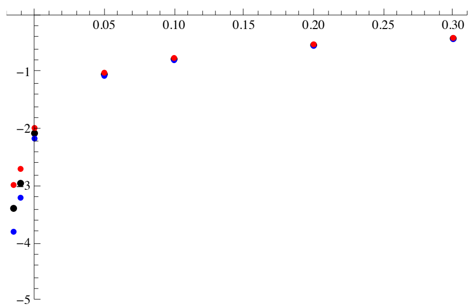
```
z = 1.96;
```

$$\underline{z} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2*n} * z^2 - z * \sqrt{\frac{1}{n} * \hat{u} * (1 - \hat{u}) + \frac{1}{4*n^2} * z^2}\right) \text{ /. } n \to \text{SampleMax} \text{ /. } \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

$$\overline{z} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2*n} * z^2 + z * \sqrt{\frac{1}{n} * \hat{u} * (1 - \hat{u}) + \frac{1}{4*n^2} * z^2}\right) \text{ /. } n \to \text{SampleMax} \text{ /. } \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

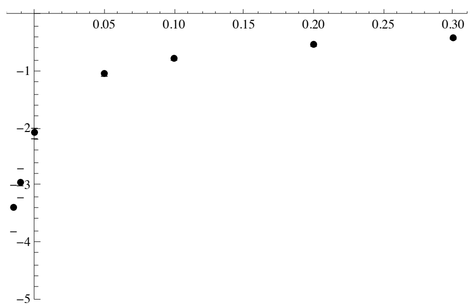
```
{0.00015556, 0.000614344, 0.00670094, 0.08445, 0.158934, 0.283368, 0.371827}
{0.00102814, 0.00196882, 0.0102767, 0.095665, 0.173522, 0.301192, 0.390864}
```

```
PlotCIunder = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder[[change, 1]] = Resultparameter[[change]];
  PlotCIupper[[change, 1]] = Resultparameter[[change]];
  PlotCIunder[[change, 2]] = Log[10, z[[change]]];
  PlotCIupper[[change, 2]] = Log[10, z̄[[change]]];
 ];
gCIupperDot = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Red}];
gCIunderDot = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Blue}];
Show[gLogNumerical, gCIunderDot, gCIupperDot]
```

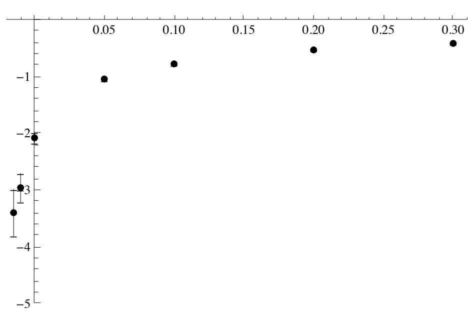



```
Log[10, z̄]
N[Log[10, PlotpM[[All, 2]]]]
Log[10, z]
gCIupper = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
Show[gLogNumerical, gCIunder, gCIupper]
```

```
{-2.98795, -2.70579, -1.98815, -1.01925, -0.760644, -0.521157, -0.407975}
{-3.39794, -2.95861, -2.08092, -1.04624, -0.77963, -0.53432, -0.418733}
{-3.8081, -3.21159, -2.17386, -1.0734, -0.798783, -0.54765, -0.429659}
```



```
PlotCI1 = Table[0, {i, 2}, {j, 2}];
PlotCI2 = Table[0, {i, 2}, {j, 2}];
PlotCI3 = Table[0, {i, 2}, {j, 2}];
PlotCI4 = Table[0, {i, 2}, {j, 2}];
PlotCI5 = Table[0, {i, 2}, {j, 2}];
PlotCI6 = Table[0, {i, 2}, {j, 2}];
PlotCI7 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1[[1, 1]] = Resultparameter[[1]];
  PlotCI1[[1, 2]] = Log[10, z̲[[1]]];
  PlotCI1[[2, 1]] = Resultparameter[[1]];
  PlotCI1[[2, 2]] = Log[10, z̅[[1]]];
  PlotCI2[[1, 1]] = Resultparameter[[2]];
  PlotCI2[[1, 2]] = Log[10, z̲[[2]]];
  PlotCI2[[2, 1]] = Resultparameter[[2]];
  PlotCI2[[2, 2]] = Log[10, z̅[[2]]];
  PlotCI3[[1, 1]] = Resultparameter[[3]];
  PlotCI3[[1, 2]] = Log[10, z̲[[3]]];
  PlotCI3[[2, 1]] = Resultparameter[[3]];
  PlotCI3[[2, 2]] = Log[10, z̅[[3]]];
  PlotCI4[[1, 1]] = Resultparameter[[4]];
  PlotCI4[[1, 2]] = Log[10, z̲[[4]]];
  PlotCI4[[2, 1]] = Resultparameter[[4]];
  PlotCI4[[2, 2]] = Log[10, z̅[[4]]];
  PlotCI5[[1, 1]] = Resultparameter[[5]];
  PlotCI5[[1, 2]] = Log[10, z̲[[5]]];
  PlotCI5[[2, 1]] = Resultparameter[[5]];
  PlotCI5[[2, 2]] = Log[10, z̅[[5]]];
  PlotCI6[[1, 1]] = Resultparameter[[6]];
  PlotCI6[[1, 2]] = Log[10, z̲[[6]]];
  PlotCI6[[2, 1]] = Resultparameter[[6]];
  PlotCI6[[2, 2]] = Log[10, z̅[[6]]];
  PlotCI7[[1, 1]] = Resultparameter[[7]];
  PlotCI7[[1, 2]] = Log[10, z̲[[7]]];
  PlotCI7[[2, 1]] = Resultparameter[[7]];
  PlotCI7[[2, 2]] = Log[10, z̅[[7]]];
 ];
gCI1 = ListPlot[PlotCI1, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2 = ListPlot[PlotCI2, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3 = ListPlot[PlotCI3, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4 = ListPlot[PlotCI4, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5 = ListPlot[PlotCI5, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI6 = ListPlot[PlotCI6, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI7 = ListPlot[PlotCI7, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper]
```



Fixation probability from branching process approximation

```
Clear[y]

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 0;
SELwMM = 4 * y;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);

SELwRM = (wRM - wRR)/wRR;

saverage = 1/2 * (SELfM/2 + SELwM) + 1/2 * (SELwMM * HETEROw)

EqρH = Sqrt[wRR]/Sqrt[(fR*wR)/2 + Sqrt[wRR]];

EqρD = Sqrt[(fR*wR)/2]/Sqrt[(fR*wR)/2 + Sqrt[wRR]];

avef = (fM + fR)/2;
W = (fR * wR)/2 * EqρH + wRR * EqρD;
EqD4a = 1 - PH - Exp[-PD * (avef * wM)/W];
EqD4b = 1 - PD - Exp[-PH * wRM/(2 W)];
RepeatMax = 200;
ResultH = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
ResultD = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
ResultBPfix = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
LogResultBPfix = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
repeat = 0;
For[repeat1 = 1, repeat1 ≤ RepeatMax + 1, repeat1 = repeat1 + 1,
  repeat = repeat + 1;
  x1 = 0 + (MaxV - 0)/RepeatMax * (repeat1 - 1);
  CondH = EqD4a /. y → x1;
  CondD = EqD4b /. y → x1;
  sol = Solve[CondH == 0 && CondD == 0 && 0 < PH && PH < 1 && 0 < PD && PD < 1, {PH, PD}, Reals];
  BPPaiH = Part[(PH /. sol), 1];
  BPPaiD = Part[(PD /. sol), 1];
  ResultH[[repeat, 1]] = x1;
  ResultD[[repeat, 1]] = x1;
```

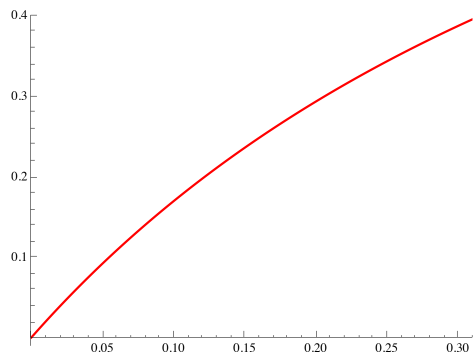
```
    ResultBPfix[[repeat, 1]] = x1;
    LogResultBPfix[[repeat, 1]] = x1;
    ResultH[[repeat, 2]] = BPPaiH;
    ResultD[[repeat, 2]] = BPPaiD;
    ResultBPfix[[repeat, 2]] = EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD;
    LogResultBPfix[[repeat, 2]] = Log[10, EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD];
  ];
gBPfix = ListPlot[ResultBPfix, Joined → True, PlotRange → {{0, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gLogBPfix = ListPlot[LogResultBPfix, Joined → True, PlotRange → {{0, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
1. y
```

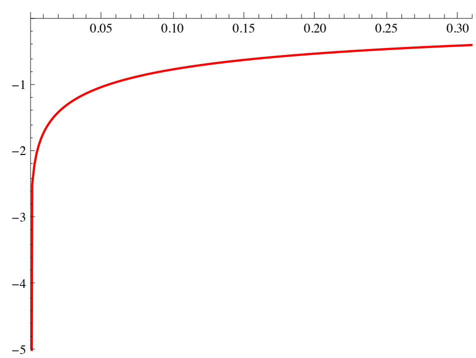
Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

General::stop : この計算中に，Solve::ratnzのこれ以上の出力は表示されません． ≫

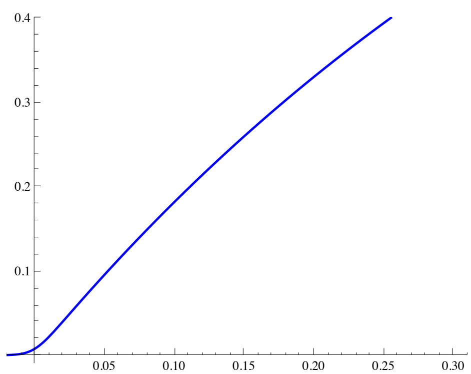


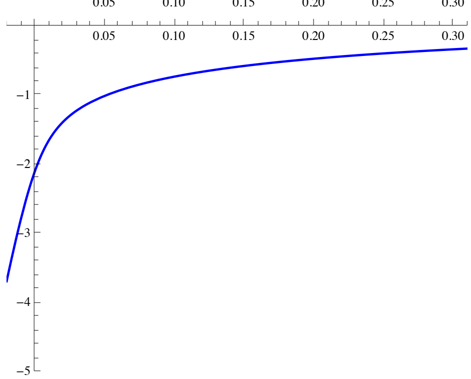


Fixation probability from diffusion approximation

```
Parameters = {SUM → Site, ρ̂H → Sqrt[wRR]/(Sqrt[fR*wR/2] + Sqrt[wRR]), ρ̂D → Sqrt[fR*wR/2]/(Sqrt[fR*wR/2] + Sqrt[wRR]), sfM → SELfM,
   swM → SELwM, swRM → SELwRM, swMM → SELwMM}
Additive = swMM - 2 * swRM;
m = SUM * p * (1 - p) * (2 * s_average + p * Additive)/2;
v = p * (1 - p) * (ρ̂H + 2 * ρ̂D)/(8 * ρ̂H * ρ̂D);
Q = Integrate[m/v /. Parameters, p];
INI = 1/(SUM * (ρ̂H + 2 * ρ̂D)) /. Parameters;
DFfixWS = Integrate[Exp[-2 * Q], {p, 0, INI}]/Integrate[Exp[-2 * Q], {p, 0, 1}];
gDFfixWS = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
   PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS = Plot[Log[10, DFfixWS], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

$$\left\{SUM \to 100,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,\ sfM \to 0,\ swM \to 0,\ swRM \to \frac{-1000 + 1000\ (1 + 2.\ y)}{1000},\ swMM \to 4\ y\right\}$$

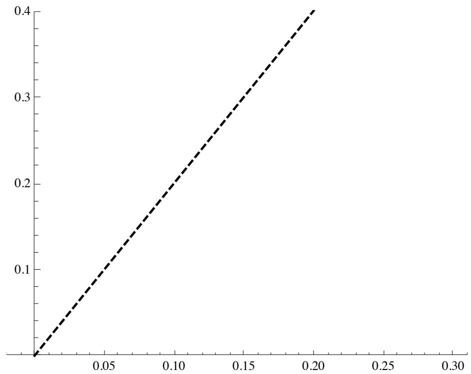


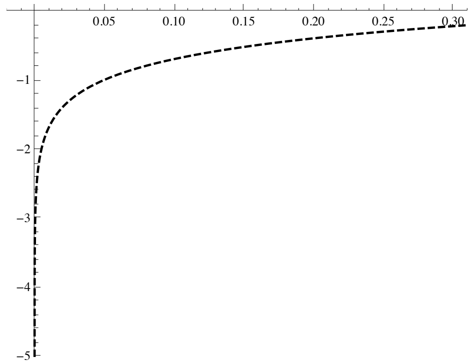


Analytical result (Lineary approximation)

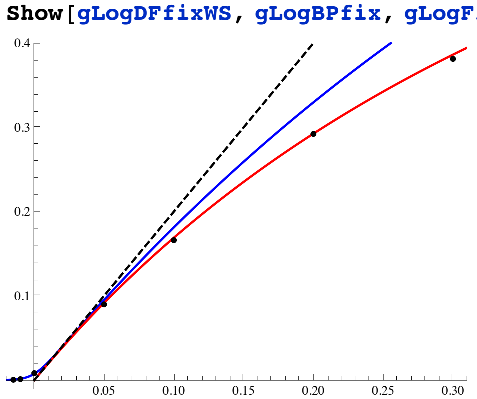
$$\text{FixLinear} = \frac{8 * \hat{\rho}_H * \hat{\rho}_D}{\left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)^2} * 2 * s_{\text{average}};$$

```
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.4}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
gLogFixLinear = Plot[Log[10, FixLinear /. Parameters], {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-5, 0}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```






```
Show[gDFfixWS, gBPfix, gFixLinear, gNumerical]
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical]
```



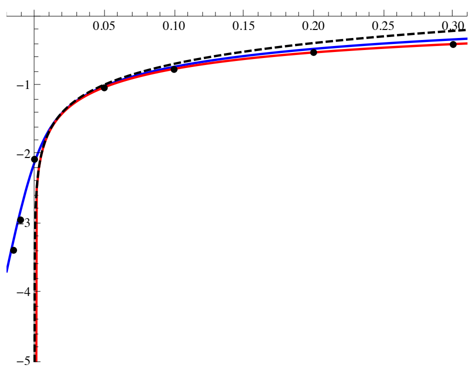
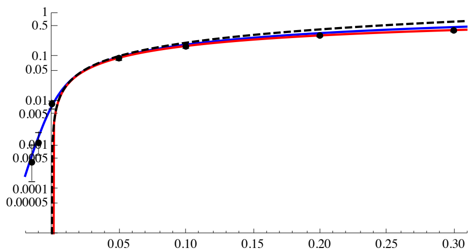


Figure for output

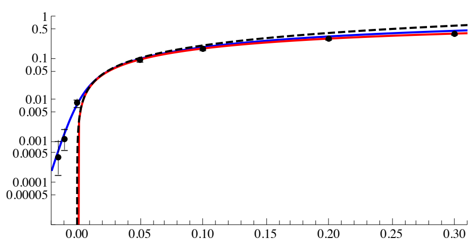
```
axeslabels = {{-5, "0.00001"}, {Log[10, 0.00005], "0.00005"}, {-4, "0.0001"},
   {Log[10, 0.0005], "0.0005"}, {-3, "0.001"}, {Log[10, 0.005], "0.005"},
   {-2, "0.01"}, {Log[10, 0.05], "0.05"}, {-1, "0.1"}, {Log[10, 0.5], "0.5"}, {0, "1"}};
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
 AxesOrigin → {0, -5}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.5]
```

$$\frac{\sqrt{wRR}}{\sqrt{\frac{fR*wR}{2}} + \sqrt{wRR}}$$

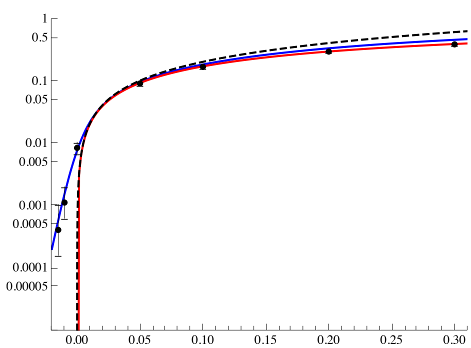


```
0.666667
```

```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
 AxesOrigin → {-0.02, -5}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.5]
```



```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
 AxesOrigin → {-0.02, -5}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



- **Fig.1(f)**

Parameter values

```
Clear[x]

Site = 100;

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 0;
SELwMM = 2 * x;
HETEROw = 1.0;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\text{SELwRM} = \frac{\text{wRM} - \text{wRR}}{\text{wRR}};$$

```
MinV = -0.02;
MaxV = 0.31;
```

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SELwM}\right) + \frac{1}{2} * (\text{SELwMM} * \text{HETEROw})$$

```
1. x
```

Out put

```
SampleMax = 10 000;
ChangeMax = 6;
OutputData = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulaton

```
VectpM = Table[0, {SampleMax}];

Resultparameter = Table[0, {ChangeMax + 1}];
Result = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.015];
  If[change == 2, x = -0.01];
  If[change == 3, x = 0];
  If[change == 4, x = 0.05];
  If[change == 5, x = 0.1];
  If[change == 6, x = 0.2];
  If[change == 7, x = 0.3];

  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,
```

```
EqρH = Sqrt[wRR] / Sqrt[Sqrt[(fR*wR)/2] + Sqrt[wRR]];

xR = Round[Site * EqρH];
xM = 0;
xRR = Site - xR;
xRM = 0;
xMM = 0;

sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];

xR = xR - sampleMutation[[1]];
xM = xM + sampleMutation[[1]];
xRR = xRR - sampleMutation[[2]];
xRM = xRM + sampleMutation[[2]];
xMM = 0;

If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

step = 2;
While[0 < pM && pM < 1,
 cR = wRR * xRR + wRM/2 * xRM;
 cM = wMM * xMM + wRM/2 * xRM;
 If[xR == 0 && xM == 0, cRR = 0];
 If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (wR^2 * xR^2)/(wR * xR + wM * xM)];
 If[xR == 0 && xM == 0, cRM = 0];
 If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (wR * wM * xR * xM)/(wR * xR + wM * xM)];
 If[xR == 0 && xM == 0, cMM = 0];
 If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (wM^2 * xM^2)/(wR * xR + wM * xM)];
 pCellR = cR/(cR + cM + cRR + cRM + cMM);
 pCellM = cM/(cR + cM + cRR + cRM + cMM);
 pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
 pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
 pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
 sampleBirth = RandomVariate[MultinomialDistribution[Site,
    {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

 xR = sampleBirth[[1]];
 xM = sampleBirth[[2]];
 xRR = sampleBirth[[3]];
 xRM = sampleBirth[[4]];
```

```
        xMM = sampleBirth[[5]];
        If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
        If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
        If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

        step = step + 1;
       ];

      VectpM[[sample]] = pM;

      OutputData[[countData, 1]] = x;
      OutputData[[countData, 2]] = pM;
      countData = countData + 1;

     ];

    Resultparameter[[change]] = x;
    Result[[change]] = Total[VectpM];

    OutputResult[[countResult, 1]] = x;
    OutputResult[[countResult, 2]] = Total[VectpM];
    countResult = countResult + 1;

    Print[change]

   ];
```

1

2

3

4

5

6

7

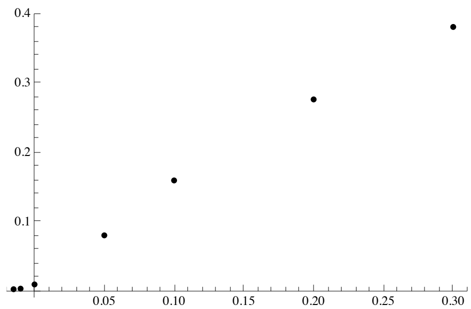
Output of the data

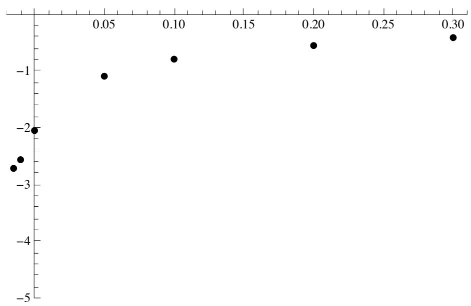
```
Resultparameter
Result
Export["/Users/Bessho/Desktop/workstation/OutputDataFig1f.txt", OutputData, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig1f.txt", OutputResult, "Table"];
```

```
{-0.015, -0.01, 0, 0.05, 0.1, 0.2, 0.3}
{19, 27, 88, 795, 1586, 2753, 3797}
```

Plot

```
PlotpM = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM[[change, 1]] = Resultparameter[[change]];
  PlotpM[[change, 2]] = Result[[change]] / SampleMax;
  PlotpMlog[[change, 1]] = Resultparameter[[change]];
  PlotpMlog[[change, 2]] = Log[10, Result[[change]] / SampleMax];
 ];
gNumerical = ListPlot[PlotpM, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Black}]
gLogNumerical = ListPlot[PlotpMlog, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {PointSize[0.015], Black}]
PlotpM
```





$$\left\{\left\{-0.015, \frac{19}{10\,000}\right\}, \left\{-0.01, \frac{27}{10\,000}\right\}, \left\{0, \frac{11}{1250}\right\}, \left\{0.05, \frac{159}{2000}\right\}, \left\{0.1, \frac{793}{5000}\right\}, \left\{0.2, \frac{2753}{10\,000}\right\}, \left\{0.3, \frac{3797}{10\,000}\right\}\right\}$$

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \overline{z}] \sim 0.95$.

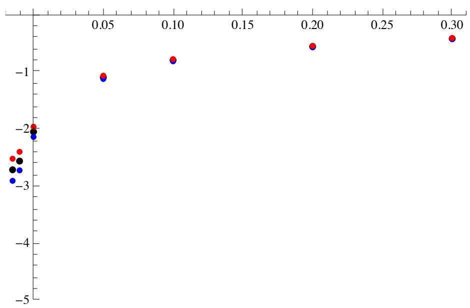
```
z = 1.96;
```

$$\underline{z} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /.\ n \to \text{SampleMax} /.\ \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

$$\overline{z} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /.\ n \to \text{SampleMax} /.\ \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

```
{0.00121673, 0.00185632, 0.00714875, 0.0743579, 0.151571, 0.266633, 0.370236}
{0.00296582, 0.00392562, 0.0108285, 0.0849651, 0.165891, 0.28414, 0.389257}
```

```
PlotCIunder = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder[[change, 1]] = Resultparameter[[change]];
  PlotCIupper[[change, 1]] = Resultparameter[[change]];
  PlotCIunder[[change, 2]] = Log[10, z̲[[change]]];
  PlotCIupper[[change, 2]] = Log[10, z̅[[change]]];
 ];
gCIupperDot = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Red}];
gCIunderDot = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Blue}];
Show[gLogNumerical, gCIunderDot, gCIupperDot]
```

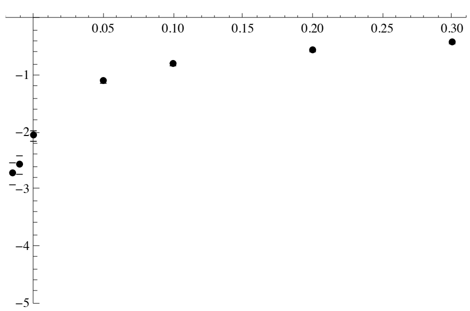



```
Log[10, z̅]
N[Log[10, PlotpM[[All, 2]]]]
Log[10, z̲]
gCIupper = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
Show[gLogNumerical, gCIunder, gCIupper]
```

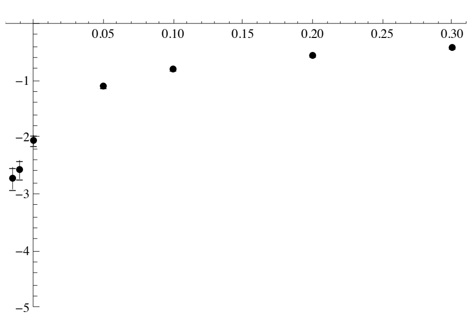
{-2.52786, -2.40609, -1.96543, -1.07076, -0.780178, -0.546468, -0.409764}

{-2.72125, -2.56864, -2.05552, -1.09963, -0.799697, -0.560194, -0.420559}

{-2.9148, -2.73135, -2.14577, -1.12867, -0.819383, -0.574086, -0.431522}



```
PlotCI1 = Table[0, {i, 2}, {j, 2}];
PlotCI2 = Table[0, {i, 2}, {j, 2}];
PlotCI3 = Table[0, {i, 2}, {j, 2}];
PlotCI4 = Table[0, {i, 2}, {j, 2}];
PlotCI5 = Table[0, {i, 2}, {j, 2}];
PlotCI6 = Table[0, {i, 2}, {j, 2}];
PlotCI7 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1[[1, 1]] = Resultparameter[[1]];
  PlotCI1[[1, 2]] = Log[10, z̲[[1]]];
  PlotCI1[[2, 1]] = Resultparameter[[1]];
  PlotCI1[[2, 2]] = Log[10, z̅[[1]]];
  PlotCI2[[1, 1]] = Resultparameter[[2]];
  PlotCI2[[1, 2]] = Log[10, z̲[[2]]];
  PlotCI2[[2, 1]] = Resultparameter[[2]];
  PlotCI2[[2, 2]] = Log[10, z̅[[2]]];
  PlotCI3[[1, 1]] = Resultparameter[[3]];
  PlotCI3[[1, 2]] = Log[10, z̲[[3]]];
  PlotCI3[[2, 1]] = Resultparameter[[3]];
  PlotCI3[[2, 2]] = Log[10, z̅[[3]]];
  PlotCI4[[1, 1]] = Resultparameter[[4]];
  PlotCI4[[1, 2]] = Log[10, z̲[[4]]];
  PlotCI4[[2, 1]] = Resultparameter[[4]];
  PlotCI4[[2, 2]] = Log[10, z̅[[4]]];
  PlotCI5[[1, 1]] = Resultparameter[[5]];
  PlotCI5[[1, 2]] = Log[10, z̲[[5]]];
  PlotCI5[[2, 1]] = Resultparameter[[5]];
  PlotCI5[[2, 2]] = Log[10, z̅[[5]]];
  PlotCI6[[1, 1]] = Resultparameter[[6]];
  PlotCI6[[1, 2]] = Log[10, z̲[[6]]];
  PlotCI6[[2, 1]] = Resultparameter[[6]];
  PlotCI6[[2, 2]] = Log[10, z̅[[6]]];
  PlotCI7[[1, 1]] = Resultparameter[[7]];
  PlotCI7[[1, 2]] = Log[10, z̲[[7]]];
  PlotCI7[[2, 1]] = Resultparameter[[7]];
  PlotCI7[[2, 2]] = Log[10, z̅[[7]]];
 ];
gCI1 = ListPlot[PlotCI1, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2 = ListPlot[PlotCI2, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3 = ListPlot[PlotCI3, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4 = ListPlot[PlotCI4, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5 = ListPlot[PlotCI5, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI6 = ListPlot[PlotCI6, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI7 = ListPlot[PlotCI7, PlotRange → {{MinV, MaxV}, {-5, -0.5}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper]
```



Fixation probability from branching process approximation

```
Clear[y]

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 0;
SELwMM = 2 * y;
HETEROw = 1.0;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);

SELwRM = (wRM - wRR)/wRR;

saverage = 1/2 * (SELfM/2 + SELwM) + 1/2 * (SELwMM * HETEROw)

EqρH = Sqrt[wRR] / (Sqrt[(fR*wR)/2] + Sqrt[wRR]);

EqρD = Sqrt[(fR*wR)/2] / (Sqrt[(fR*wR)/2] + Sqrt[wRR]);

avef = (fM + fR)/2;
W = (fR * wR)/2 * EqρH + wRR * EqρD;
EqD4a = 1 - PH - Exp[-PD * (avef * wM)/W];
EqD4b = 1 - PD - Exp[-PH * wRM/(2 W)];
RepeatMax = 200;
ResultH = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
ResultD = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
ResultBPfix = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
LogResultBPfix = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
repeat = 0;
For[repeat1 = 1, repeat1 ≤ RepeatMax + 1, repeat1 = repeat1 + 1,
  repeat = repeat + 1;
  x1 = 0 + (MaxV - 0)/RepeatMax * (repeat1 - 1);
  CondH = EqD4a /. y → x1;
  CondD = EqD4b /. y → x1;
  sol = Solve[CondH == 0 && CondD == 0 && 0 < PH && PH < 1 && 0 < PD && PD < 1, {PH, PD}, Reals];
  BPPaiH = Part[(PH /. sol), 1];
  BPPaiD = Part[(PD /. sol), 1];
  ResultH[[repeat, 1]] = x1;
  ResultD[[repeat, 1]] = x1;
```

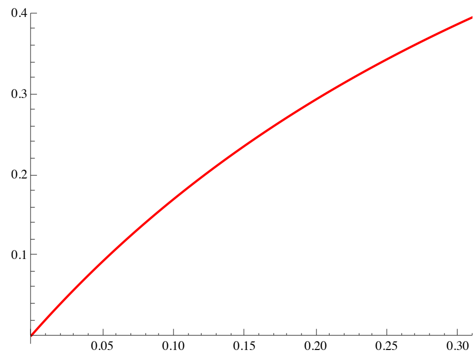
```
    ResultBPfix[[repeat, 1]] = x1;
    LogResultBPfix[[repeat, 1]] = x1;
    ResultH[[repeat, 2]] = BPPaiH;
    ResultD[[repeat, 2]] = BPPaiD;
    ResultBPfix[[repeat, 2]] = EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD;
    LogResultBPfix[[repeat, 2]] = Log[10, EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD];
  ];
gBPfix = ListPlot[ResultBPfix, Joined → True, PlotRange → {{0, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gLogBPfix = ListPlot[LogResultBPfix, Joined → True, PlotRange → {{0, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
1. y
```

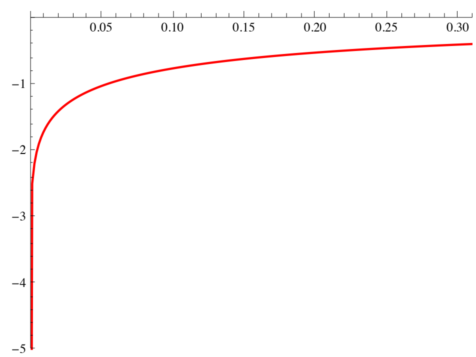
Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

General::stop : この計算中に，Solve::ratnzのこれ以上の出力は表示されません． ≫

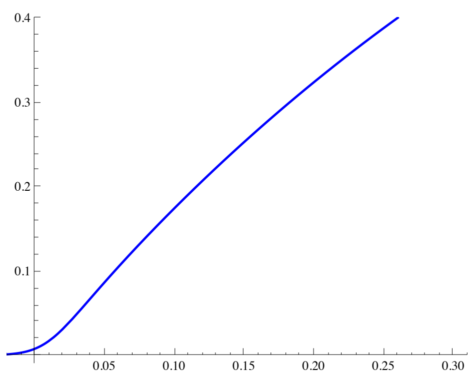


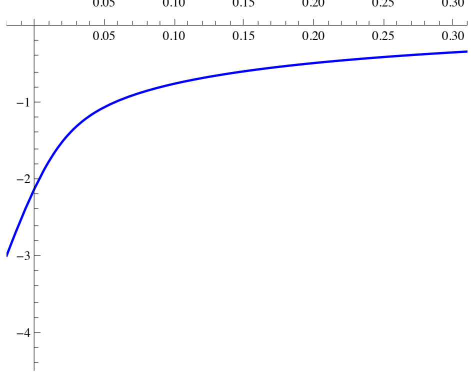


Fixation probability from diffusion approximation

```
Parameters = {SUM → Site, ρ̂_H → Sqrt[wRR]/(Sqrt[fR*wR/2] + Sqrt[wRR]), ρ̂_D → Sqrt[fR*wR/2]/(Sqrt[fR*wR/2] + Sqrt[wRR]), sfM → SELfM,
   swM → SELwM, swRM → SELwRM, swMM → SELwMM }
Additive = swMM - 2 * swRM;
m = SUM * p * (1 - p) * (2 * s_average + p * Additive)/2;
v = (p * (1 - p) * (ρ̂_H + 2 * ρ̂_D))/(8 * ρ̂_H * ρ̂_D);
Q = Integrate[m/v /. Parameters, p];
INI = 1/(SUM * (ρ̂_H + 2 * ρ̂_D)) /. Parameters;
DFfixWS = Integrate[Exp[-2 * Q], {p, 0, INI}]/Integrate[Exp[-2 * Q], {p, 0, 1}];
gDFfixWS = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
   PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS = Plot[Log[10, DFfixWS], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-4.5, 0}},
   PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

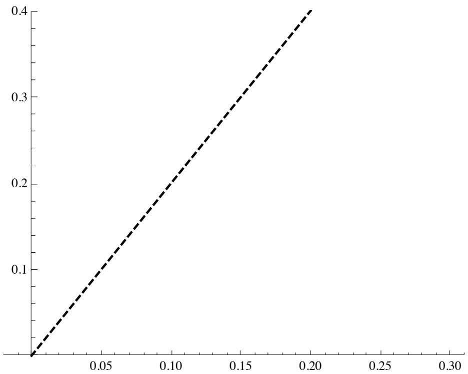
$$\left\{SUM \to 100,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,\ sfM \to 0,\ swM \to 0,\ swRM \to \frac{-1000 + 1000\,(1 + 2.\,y)}{1000},\ swMM \to 2\,y\right\}$$

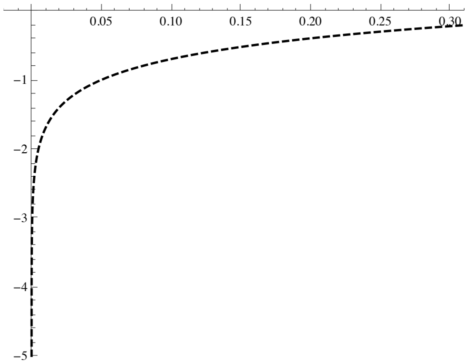




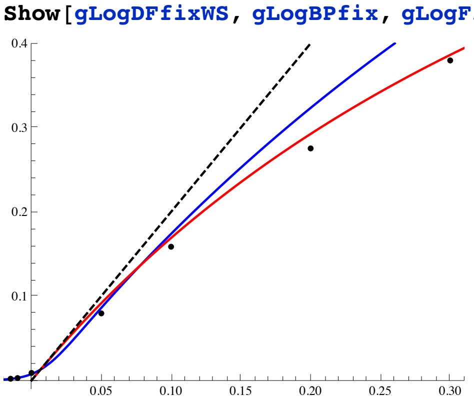
Analytical result (Lineary approximation)

```
FixLinear = (8 * ρ̂H * ρ̂D) / (ρ̂H + 2 * ρ̂D)^2 * 2 * saverage;
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.4}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
gLogFixLinear = Plot[Log[10, FixLinear /. Parameters], {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-5, 0}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```






```
Show[gDFfixWS, gBPfix, gFixLinear, gNumerical]
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical]
```



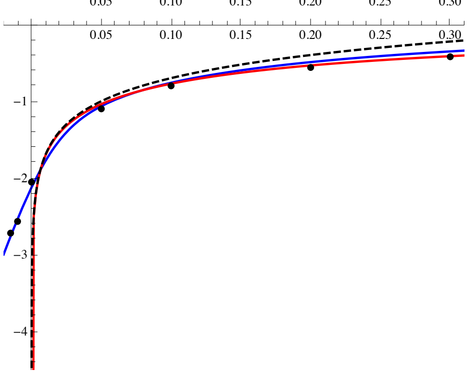
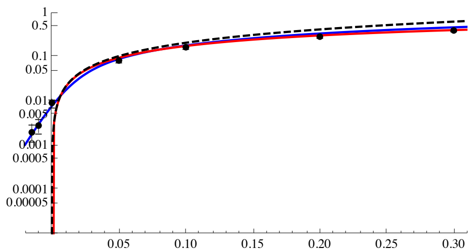


Figure for output

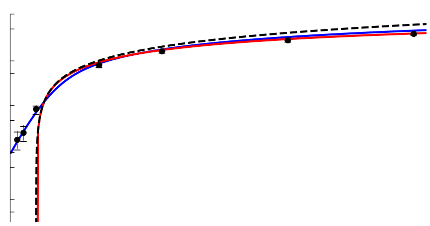
```
axeslabels = {{-5, "0.00001"}, {Log[10, 0.00005], "0.00005"}, {-4, "0.0001"},
   {Log[10, 0.0005], "0.0005"}, {-3, "0.001"}, {Log[10, 0.005], "0.005"},
   {-2, "0.01"}, {Log[10, 0.05], "0.05"}, {-1, "0.1"}, {Log[10, 0.5], "0.5"}, {0, "1"}};
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
 AxesOrigin → {0, -5}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.5]
```

$$\frac{\sqrt{wRR}}{\sqrt{\frac{fR*wR}{2}}+\sqrt{wRR}}$$

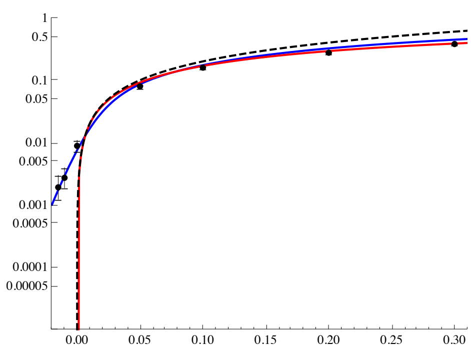


0.666667

```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-4.5, 0}},
 AxesOrigin → {-0.02, -5}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.5]
```



```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
 AxesOrigin → {-0.02, -5}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



- **Fig.2(a)**

Parameter values

```
Clear[x]

Site = 100;
fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 0;
SELβMM = x;
HETEROβ = 0;
SELdM = 0;
SELdMM = 0;
HETEROd = 0.5;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

```
MinV = -0.02;
MaxV = 0.31;
```

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SEL}\beta\text{M} + \text{SELdM}\right) + \frac{1}{2} * (\text{SEL}\beta\text{MM} * \text{HETERO}\beta + \text{SELdM} * \text{HETEROd})$$

```
0.
```

Out put

```
SampleMax = 10 000;
ChangeMax = 6;
OutputData = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulation

```
VectpM = Table[0, {SampleMax}];

Resultparameter = Table[0, {ChangeMax + 1}];
Result = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.015];
  If[change == 2, x = -0.01];
  If[change == 3, x = 0];
  If[change == 4, x = 0.05];
  If[change == 5, x = 0.1];
  If[change == 6, x = 0.2];
  If[change == 7, x = 0.3];
```

```
sample = 1;
For[sample = 1, sample ≤ SampleMax, sample = sample + 1,

 EqρH = Sqrt[βRR * dRR] / Sqrt[(fR*βR*dR)/2 + Sqrt[βRR * dRR]];
 xR = Round[Site * EqρH];
 xM = 0;
 xRR = Site - xR;
 xRM = 0;
 xMM = 0;
 sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];
 xR = xR - sampleMutation[[1]];
 xM = xM + sampleMutation[[1]];
 xRR = xRR - sampleMutation[[2]];
 xRM = xRM + sampleMutation[[2]];
 xMM = 0;

 If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
 If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
 If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

 step = 2;
 While[0 < pM && pM < 1,
  pDeathR = (dR * xR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathM = (dM * xM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathRR = (dRR * xRR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathRM = (dRM * xRM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathMM = (dMM * xMM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  sampleDeath =
   RandomVariate[MultinomialDistribution[1,
     {pDeathR, pDeathM, pDeathRR, pDeathRM, pDeathMM}]];
  xR = xR - sampleDeath[[1]];
  xM = xM - sampleDeath[[2]];
  xRR = xRR - sampleDeath[[3]];
  xRM = xRM - sampleDeath[[4]];
  xMM = xMM - sampleDeath[[5]];

  cR = βRR * xRR + βRM/2 * xRM;
  cM = βMM * xMM + βRM/2 * xRM;
  If[xR == 0 && xM == 0, cRR = 0];
```

```
    If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (βR^2 * xR^2)/(βR * xR + βM * xM)];
    If[xR == 0 && xM == 0, cRM = 0];
    If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (βR * βM * xR * xM)/(βR * xR + βM * xM)];
    If[xR == 0 && xM == 0, cMM = 0];
    If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (βM^2 * xM^2)/(βR * xR + βM * xM)];
    pCellR = cR/(cR + cM + cRR + cRM + cMM);
    pCellM = cM/(cR + cM + cRR + cRM + cMM);
    pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
    pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
    pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
    sampleBirth =
     RandomVariate[MultinomialDistribution[1,
       {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

    xR = xR + sampleBirth[[1]];
    xM = xM + sampleBirth[[2]];
    xRR = xRR + sampleBirth[[3]];
    xRM = xRM + sampleBirth[[4]];
    xMM = xMM + sampleBirth[[5]];


    If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
    If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
    If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

    step = step + 1;
   ];

  VectpM[[sample]] = pM;

  OutputData[[countData, 1]] = x;
  OutputData[[countData, 2]] = pM;
  countData = countData + 1;

 ];

 Resultparameter[[change]] = x;
 Result[[change]] = Total[VectpM];

 OutputResult[[countResult, 1]] = x;
 OutputResult[[countResult, 2]] = Total[VectpM];
 countResult = countResult + 1;

 Print[change]

];
```

1

2

3

4

5

6

7

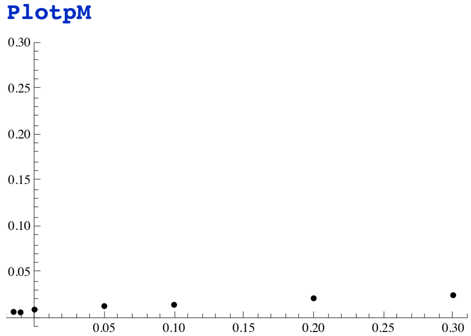
Output of the data

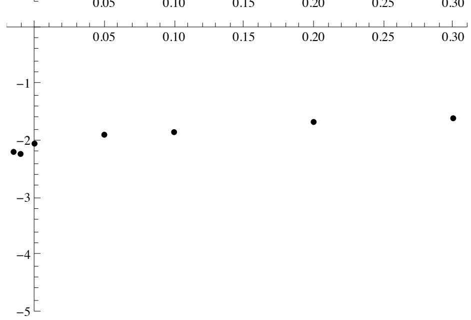
```
Resultparameter
Result
Export["/Users/Bessho/Desktop/workstation/OutputDataFig2a.txt", OutputData, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig2a.txt", OutputResult, "Table"];
```

```
{-0.015, -0.01, 0, 0.05, 0.1, 0.2, 0.3}
{63., 58., 88., 126., 140., 211., 245.}
```

Plot

```
PlotpM = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM[[change, 1]] = Resultparameter[[change]];
  PlotpM[[change, 2]] = Result[[change]] / SampleMax;
  PlotpMlog[[change, 1]] = Resultparameter[[change]];
  PlotpMlog[[change, 2]] = Log[10, Result[[change]] / SampleMax];
 ];
gNumerical = ListPlot[PlotpM, PlotRange → {{MinV, MaxV}, {-0.01, 0.3}},
  PlotStyle → {Thickness[0.005], Black}]
gLogNumerical = ListPlot[PlotpMlog, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Black}]
PlotpM
```






```
{{-0.015, 0.0063}, {-0.01, 0.0058}, {0, 0.0088},
 {0.05, 0.0126}, {0.1, 0.014}, {0.2, 0.0211}, {0.3, 0.0245}}
```

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \overline{z}] \sim 0.95$.

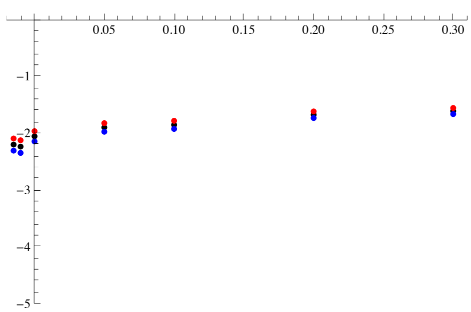
```
z = 1.96;
```

$$\underline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /.\ n \to \text{SampleMax} /.\ \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

$$\overline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /.\ n \to \text{SampleMax} /.\ \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

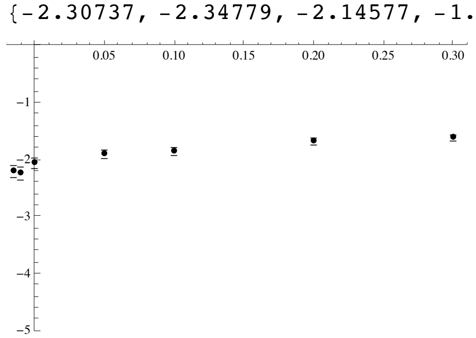
```
{0.00492754, 0.00448966, 0.00714875, 0.0105934, 0.0118767, 0.0184616, 0.0216476}
{0.00805163, 0.0074899, 0.0108285, 0.0149809, 0.0164966, 0.0241062, 0.0277176}
```

```
PlotCIunder = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder[[change, 1]] = Resultparameter[[change]];
  PlotCIupper[[change, 1]] = Resultparameter[[change]];
  PlotCIunder[[change, 2]] = Log[10, z̲[[change]]];
  PlotCIupper[[change, 2]] = Log[10, z̄[[change]]];
 ];
gCIupperDot = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Red}];
gCIunderDot = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Blue}];
Show[gLogNumerical, gCIunderDot, gCIupperDot]
```

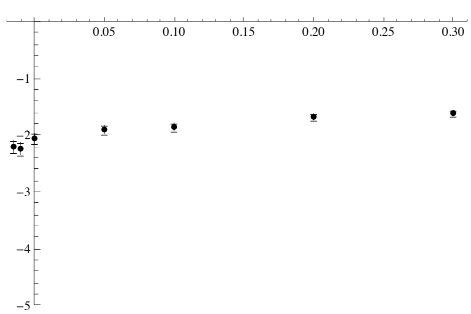


```
Log[10, z̄]
N[Log[10, PlotpM[[All, 2]]]]
Log[10, z̲]
gCIupper = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
Show[gLogNumerical, gCIunder, gCIupper]
```

```
{-2.09412, -2.12552, -1.96543, -1.82446, -1.78261, -1.61787, -1.55724}
{-2.20066, -2.23657, -2.05552, -1.89963, -1.85387, -1.67572, -1.61083}
{-2.30737, -2.34779, -2.14577, -1.97496, -1.9253, -1.73373, -1.66459}
```



```
PlotCI1 = Table[0, {i, 2}, {j, 2}];
PlotCI2 = Table[0, {i, 2}, {j, 2}];
PlotCI3 = Table[0, {i, 2}, {j, 2}];
PlotCI4 = Table[0, {i, 2}, {j, 2}];
PlotCI5 = Table[0, {i, 2}, {j, 2}];
PlotCI6 = Table[0, {i, 2}, {j, 2}];
PlotCI7 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1[[1, 1]] = Resultparameter[[1]];
  PlotCI1[[1, 2]] = Log[10, z̲[[1]]];
  PlotCI1[[2, 1]] = Resultparameter[[1]];
  PlotCI1[[2, 2]] = Log[10, z̄[[1]]];
  PlotCI2[[1, 1]] = Resultparameter[[2]];
  PlotCI2[[1, 2]] = Log[10, z̲[[2]]];
  PlotCI2[[2, 1]] = Resultparameter[[2]];
  PlotCI2[[2, 2]] = Log[10, z̄[[2]]];
  PlotCI3[[1, 1]] = Resultparameter[[3]];
  PlotCI3[[1, 2]] = Log[10, z̲[[3]]];
  PlotCI3[[2, 1]] = Resultparameter[[3]];
  PlotCI3[[2, 2]] = Log[10, z̄[[3]]];
  PlotCI4[[1, 1]] = Resultparameter[[4]];
  PlotCI4[[1, 2]] = Log[10, z̲[[4]]];
  PlotCI4[[2, 1]] = Resultparameter[[4]];
  PlotCI4[[2, 2]] = Log[10, z̄[[4]]];
  PlotCI5[[1, 1]] = Resultparameter[[5]];
  PlotCI5[[1, 2]] = Log[10, z̲[[5]]];
  PlotCI5[[2, 1]] = Resultparameter[[5]];
  PlotCI5[[2, 2]] = Log[10, z̄[[5]]];
  PlotCI6[[1, 1]] = Resultparameter[[6]];
  PlotCI6[[1, 2]] = Log[10, z̲[[6]]];
  PlotCI6[[2, 1]] = Resultparameter[[6]];
  PlotCI6[[2, 2]] = Log[10, z̄[[6]]];
  PlotCI7[[1, 1]] = Resultparameter[[7]];
  PlotCI7[[1, 2]] = Log[10, z̲[[7]]];
  PlotCI7[[2, 1]] = Resultparameter[[7]];
  PlotCI7[[2, 2]] = Log[10, z̄[[7]]];
 ];
gCI1 = ListPlot[PlotCI1, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2 = ListPlot[PlotCI2, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3 = ListPlot[PlotCI3, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4 = ListPlot[PlotCI4, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5 = ListPlot[PlotCI5, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI6 = ListPlot[PlotCI6, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI7 = ListPlot[PlotCI7, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper]
```



Analytical result (Branching process approximation)

```
Clear[y]

fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 0;
SELβMM = y;
HETEROβ = 0;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SEL}\beta\text{M} + \text{SELdM}\right) + \frac{1}{2} * (\text{SEL}\beta\text{MM} * \text{HETERO}\beta + \text{SELdM} * \text{HETEROd})$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\beta\text{RR} * \text{dRR}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

$$\text{FaiR} = \frac{\beta\text{R}}{\text{dR}};$$

$$\text{FaiM} = \frac{\beta\text{M}}{\text{dM}};$$

$$\text{FaiRR} = \frac{\beta\text{RR}}{\text{dRR}};$$

$$\text{FaiRM} = \frac{\beta\text{RM}}{\text{dRM}};$$

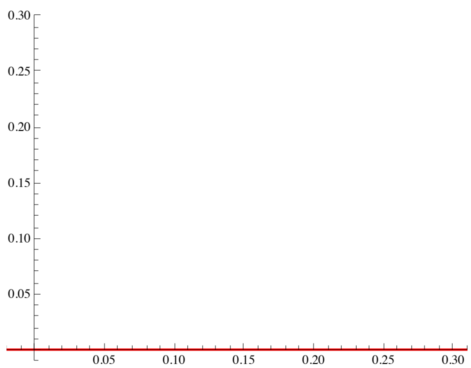
$$\text{avef} = \frac{\text{fM} + \text{fR}}{2};$$

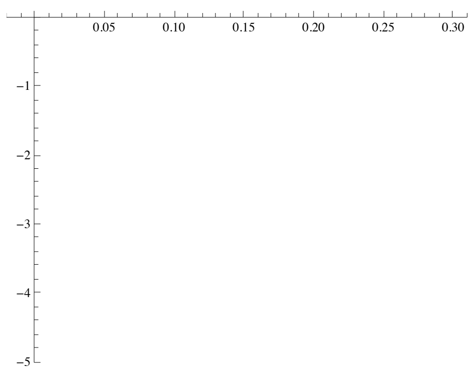
$$\text{FaiR2} = \frac{\text{fR}}{2} * \text{FaiR};$$

$$\text{FaiM2} = \frac{\text{avef}}{2} * \text{FaiM};$$

```
BPPaiH = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + FaiRM/2 * Sqrt[FaiR2 * FaiRR]);
BPPaiD = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + 2 * FaiM2 * Sqrt[FaiR2 * FaiRR]);
BPfix = EqρH / (EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD) / (EqρH + 2 * EqρD) * BPPaiD;
gBPfix = Plot[BPfix, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.3}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gLogBPfix = Plot[Log[10, BPfix], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
```

0





Analytical result (Diffusion approximation)

```
Parameters = {SUM → Site, d_R → dR, d_RR → dRR, ρ̂_H → Sqrt[βRR * dRR] / (Sqrt[(fR*βR*dR)/2] + Sqrt[βRR * dRR]),
  ρ̂_D → Sqrt[(fR*βR*dR)/2] / (Sqrt[(fR*βR*dR)/2] + Sqrt[βRR * dRR]), sfM → SELfM, sβM → SELβM, sβRM → SELβRM, sβMM → SELβMM,
  sdM → SELdM, sdRM → SELdRM, sdMM → SELdMM}
Additive = sβMM - 2 * sβRM + sdMM - 2 * sdRM;
```

$$d_{harmonic} = \frac{2}{\frac{1}{d_R} + \frac{1}{d_{RR}}};$$

$$\hat{d} = d_R * \hat{\rho}_H + d_{RR} * \left(1 - \hat{\rho}_H\right);$$

$$\overline{d} = \frac{d_R + d_{RR}}{2};$$

$$m = SUM * p * (1 - p) * \frac{2 * s_{average} + p * Additive}{2};$$

$$v = \frac{p * (1 - p) * \left(\frac{d_R}{\hat{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\hat{d}} * \hat{\rho}_D\right)}{4 * \hat{\rho}_H * \hat{\rho}_D};$$

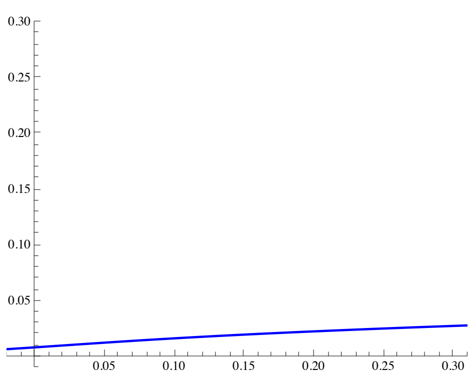
$$Q = Integrate\left[\frac{m}{v} \,/.\, Parameters,\ p\right];$$

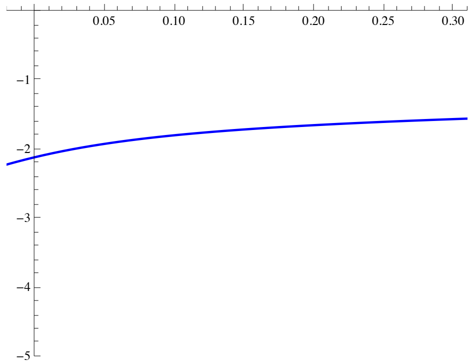
$$INI = \frac{1}{SUM * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)} \,/.\, Parameters;$$

$$DFfixWS = \frac{Integrate[Exp[-2 * Q], \{p, 0, INI\}]}{Integrate[Exp[-2 * Q], \{p, 0, 1\}]};$$

```
gDFfixWS = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.3}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS = Plot[Log[10, DFfixWS], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

$\{SUM \to 100,\ d_R \to 0.005,\ d_{RR} \to 0.005,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,$
$sfM \to 0,\ s\beta M \to 0,\ s\beta RM \to 0,\ s\beta MM \to y,\ sdM \to 0,\ sdRM \to 0.,\ sdMM \to 0\}$

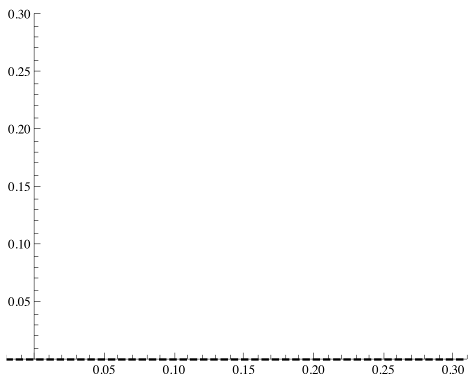


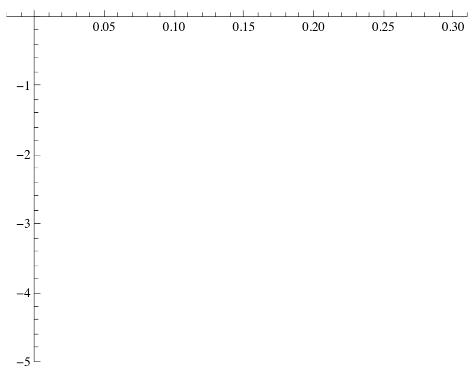


Analytical result (Lineary approximation)

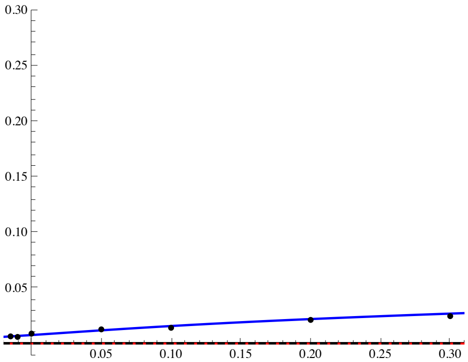
$$\text{FixLinear} = \frac{8 * \hat{\rho}_H * \hat{\rho}_D}{\left(\hat{\rho}_H + 2 * \hat{\rho}_D\right) * \left(\frac{d_R}{\bar{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\bar{d}} * \hat{\rho}_D\right)} * s_{average};$$

```
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.3}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
gLogFixLinear = Plot[Log[10, FixLinear] /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-5, 0}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```





```
Show[gDFfixWS, gBPfix, gFixLinear, gNumerical]
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical]
```



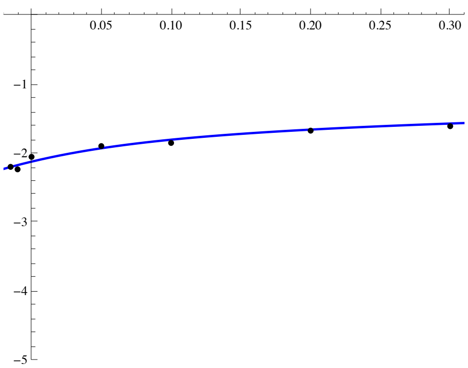
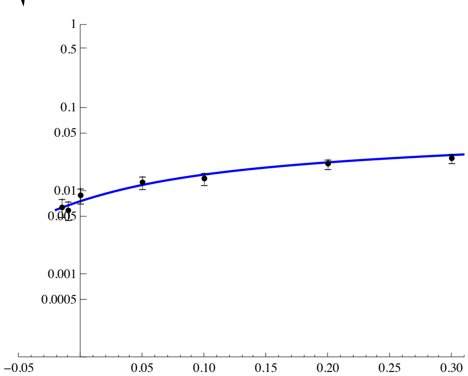


Figure for output

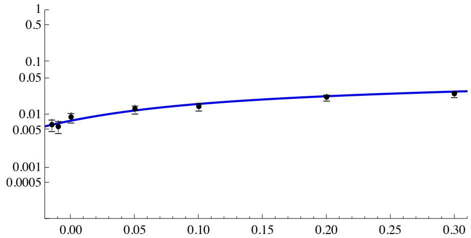
```
axeslabels = {{-4, "0.0001"}, {Log[10, 0.0005], "0.0005"}, {-3, "0.001"},
   {Log[10, 0.005], "0.005"}, {-2, "0.01"}, {Log[10, 0.05], "0.05"}, {-1, "0.1"},
   {Log[10, 0.5], "0.5"}, {0, "1"}};
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{-0.05, MaxV}, {-4, 0}},
 AxesOrigin → {0, -4}, Ticks → {Automatic, axeslabels}]
```

$$\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR * \beta R * dR}{2}} + \sqrt{\beta RR * dRR}}$$

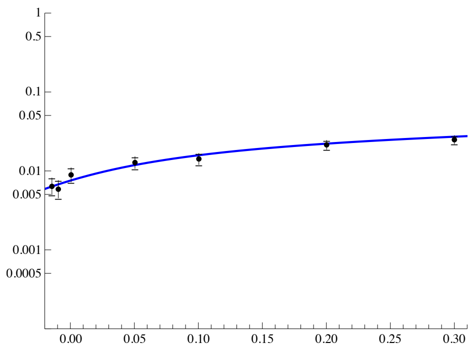


0.666667

```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
 AxesOrigin → {-0.02, -4}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.5]
```




```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
 AxesOrigin → {-0.02, -4}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



- **Fig.2(b)**

Parameter values

```
Clear[x]

Site = 100;

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 0;
SELwMM = x;
HETEROw = 0;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\mathrm{SELwRM} = \frac{\mathrm{wRM} - \mathrm{wRR}}{\mathrm{wRR}};$$

```
MinV = -0.02;
MaxV = 0.31;
```

$$s_{average} = \frac{1}{2} * \left(\frac{\mathrm{SELfM}}{2} + \mathrm{SELwM}\right) + \frac{1}{2} * (\mathrm{SELwMM} * \mathrm{HETEROw})$$

0

Out put

```
SampleMax = 10 000;
ChangeMax = 6;
OutputData = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulaton

```
VectpM = Table[0, {SampleMax}];

Resultparameter = Table[0, {ChangeMax + 1}];
Result = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.015];
  If[change == 2, x = -0.01];
  If[change == 3, x = 0];
  If[change == 4, x = 0.05];
  If[change == 5, x = 0.1];
  If[change == 6, x = 0.2];
  If[change == 7, x = 0.3];

  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,

   EqρH = Sqrt[wRR] / (Sqrt[fR*wR/2 + Sqrt[wRR]]);

   xR = Round[Site * EqρH];
   xM = 0;
   xRR = Site - xR;
   xRM = 0;
   xMM = 0;

   sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), 2 * xRR/(xR + 2 * xRR)}]];
   xR = xR - sampleMutation[[1]];
   xM = xM + sampleMutation[[1]];
   xRR = xRR - sampleMutation[[2]];
   xRM = xRM + sampleMutation[[2]];
   xMM = 0;


   If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
   If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
   If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

   step = 2;
   While[0 < pM && pM < 1,

    cR = wRR * xRR + wRM/2 * xRM;
    cM = wMM * xMM + wRM/2 * xRM;
```

```
If[xR == 0 && xM == 0, cRR = 0];
If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (wR^2 * xR^2)/(wR * xR + wM * xM)];
If[xR == 0 && xM == 0, cRM = 0];
If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (wR * wM * xR * xM)/(wR * xR + wM * xM)];
If[xR == 0 && xM == 0, cMM = 0];
If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (wM^2 * xM^2)/(wR * xR + wM * xM)];
pCellR = cR/(cR + cM + cRR + cRM + cMM);
pCellM = cM/(cR + cM + cRR + cRM + cMM);
pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
sampleBirth = RandomVariate[MultinomialDistribution[Site,
    {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

xR = sampleBirth[[1]];
xM = sampleBirth[[2]];
xRR = sampleBirth[[3]];
xRM = sampleBirth[[4]];
xMM = sampleBirth[[5]];
If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

step = step + 1;
];

VectpM[[sample]] = pM;

OutputData[[countData, 1]] = x;
OutputData[[countData, 2]] = pM;
countData = countData + 1;

];

Resultparameter[[change]] = x;
Result[[change]] = Total[VectpM];

OutputResult[[countResult, 1]] = x;
OutputResult[[countResult, 2]] = Total[VectpM];
countResult = countResult + 1;

Print[change]

];
```

1

2

3

4

5

6

7

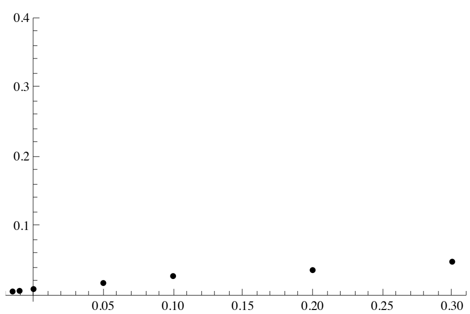
Output of the data

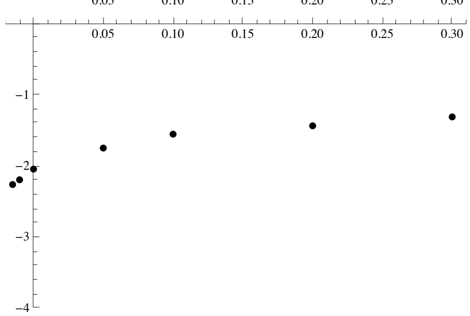
```
Resultparameter
Result
Export["/Users/Bessho/Desktop/workstation/OutputDataFig2b.txt", OutputData, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig2b.txt", OutputResult, "Table"];
```

```
{-0.015, -0.01, 0, 0.05, 0.1, 0.2, 0.3}
{54, 63, 89, 176, 276, 362, 483}
```

Plot

```
PlotpM = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM[[change, 1]] = Resultparameter[[change]];
  PlotpM[[change, 2]] = Result[[change]] / SampleMax;
  PlotpMlog[[change, 1]] = Resultparameter[[change]];
  PlotpMlog[[change, 2]] = Log[10, Result[[change]] / SampleMax];
 ];
gNumerical = ListPlot[PlotpM, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Black}]
gLogNumerical = ListPlot[PlotpMlog, PlotRange → {{MinV, MaxV}, {-4, 0}},
  PlotStyle → {PointSize[0.015], Black}]
PlotpM
```





$$\left\{\left\{-0.015, \frac{27}{5000}\right\}, \left\{-0.01, \frac{63}{10\,000}\right\}, \left\{0, \frac{89}{10\,000}\right\}, \left\{0.05, \frac{11}{625}\right\}, \left\{0.1, \frac{69}{2500}\right\}, \left\{0.2, \frac{181}{5000}\right\}, \left\{0.3, \frac{483}{10\,000}\right\}\right\}$$

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u}=\frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \overline{z}] \sim 0.95$.

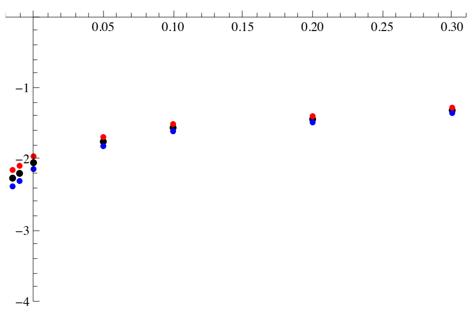
```
z = 1.96;
```

$$\underline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*\left(1-\hat{u}\right) + \frac{1}{4*n^2}*z^2}\right) \text{/. } n \to \text{SampleMax /. } \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

$$\overline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*\left(1-\hat{u}\right) + \frac{1}{4*n^2}*z^2}\right) \text{/. } n \to \text{SampleMax /. } \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

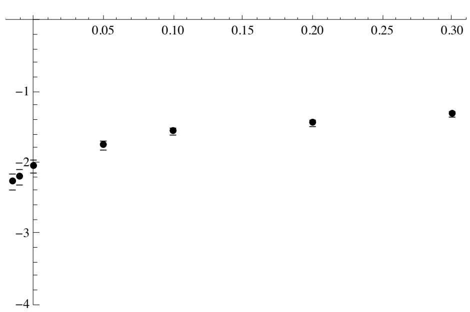
```
{0.0041413, 0.00492754, 0.00723849, 0.0152018, 0.024566, 0.0327134, 0.0442685}
{0.00703857, 0.00805163, 0.0109387, 0.0203687, 0.0309969, 0.0400428, 0.0526785}
```

```
PlotCIunder = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder[[change, 1]] = Resultparameter[[change]];
  PlotCIupper[[change, 1]] = Resultparameter[[change]];
  PlotCIunder[[change, 2]] = Log[10, z̲[[change]]];
  PlotCIupper[[change, 2]] = Log[10, z̄[[change]]];
 ];
gCIupperDot = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
   PlotStyle → {Thickness[0.005], Red}];
gCIunderDot = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-4, 0}},
   PlotStyle → {Thickness[0.005], Blue}];
Show[gLogNumerical, gCIunderDot, gCIupperDot]
```

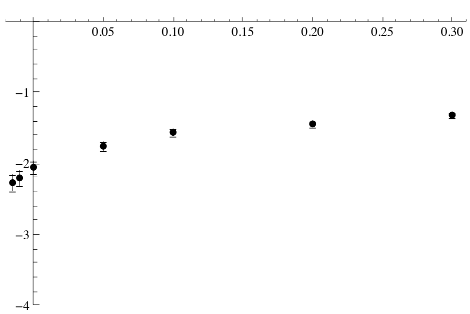


```
Log[10, z̄]
N[Log[10, PlotpM[[All, 2]]]]
Log[10, z̲]
gCIupper = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-4, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
Show[gLogNumerical, gCIunder, gCIupper]
```

```
{-2.15252, -2.09412, -1.96103, -1.69104, -1.50868, -1.39748, -1.27837}
{-2.26761, -2.20066, -2.05061, -1.75449, -1.55909, -1.44129, -1.31605}
{-2.38286, -2.30737, -2.14035, -1.8181, -1.60967, -1.48527, -1.35391}
```



```
PlotCI1 = Table[0, {i, 2}, {j, 2}];
PlotCI2 = Table[0, {i, 2}, {j, 2}];
PlotCI3 = Table[0, {i, 2}, {j, 2}];
PlotCI4 = Table[0, {i, 2}, {j, 2}];
PlotCI5 = Table[0, {i, 2}, {j, 2}];
PlotCI6 = Table[0, {i, 2}, {j, 2}];
PlotCI7 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1[[1, 1]] = Resultparameter[[1]];
  PlotCI1[[1, 2]] = Log[10, z̲[[1]]];
  PlotCI1[[2, 1]] = Resultparameter[[1]];
  PlotCI1[[2, 2]] = Log[10, z̅[[1]]];
  PlotCI2[[1, 1]] = Resultparameter[[2]];
  PlotCI2[[1, 2]] = Log[10, z̲[[2]]];
  PlotCI2[[2, 1]] = Resultparameter[[2]];
  PlotCI2[[2, 2]] = Log[10, z̅[[2]]];
  PlotCI3[[1, 1]] = Resultparameter[[3]];
  PlotCI3[[1, 2]] = Log[10, z̲[[3]]];
  PlotCI3[[2, 1]] = Resultparameter[[3]];
  PlotCI3[[2, 2]] = Log[10, z̅[[3]]];
  PlotCI4[[1, 1]] = Resultparameter[[4]];
  PlotCI4[[1, 2]] = Log[10, z̲[[4]]];
  PlotCI4[[2, 1]] = Resultparameter[[4]];
  PlotCI4[[2, 2]] = Log[10, z̅[[4]]];
  PlotCI5[[1, 1]] = Resultparameter[[5]];
  PlotCI5[[1, 2]] = Log[10, z̲[[5]]];
  PlotCI5[[2, 1]] = Resultparameter[[5]];
  PlotCI5[[2, 2]] = Log[10, z̅[[5]]];
  PlotCI6[[1, 1]] = Resultparameter[[6]];
  PlotCI6[[1, 2]] = Log[10, z̲[[6]]];
  PlotCI6[[2, 1]] = Resultparameter[[6]];
  PlotCI6[[2, 2]] = Log[10, z̅[[6]]];
  PlotCI7[[1, 1]] = Resultparameter[[7]];
  PlotCI7[[1, 2]] = Log[10, z̲[[7]]];
  PlotCI7[[2, 1]] = Resultparameter[[7]];
  PlotCI7[[2, 2]] = Log[10, z̅[[7]]];
 ];
gCI1 = ListPlot[PlotCI1, PlotRange → {{MinV, MaxV}, {-4, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2 = ListPlot[PlotCI2, PlotRange → {{MinV, MaxV}, {-4, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3 = ListPlot[PlotCI3, PlotRange → {{MinV, MaxV}, {-4, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4 = ListPlot[PlotCI4, PlotRange → {{MinV, MaxV}, {-4, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5 = ListPlot[PlotCI5, PlotRange → {{MinV, MaxV}, {-4, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI6 = ListPlot[PlotCI6, PlotRange → {{MinV, MaxV}, {-4, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI7 = ListPlot[PlotCI7, PlotRange → {{MinV, MaxV}, {-4, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper]
```



Fixation probability from branching process approximation

```
Clear[y]

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 0;
SELwMM = y;
HETEROw = 0;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\text{SELwRM} = \frac{\text{wRM} - \text{wRR}}{\text{wRR}};$$

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SELwM}\right) + \frac{1}{2} * (\text{SELwMM} * \text{HETEROw})$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR}*\text{wR}}{2}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{avef} = \frac{\text{fM} + \text{fR}}{2};$$

$$\text{W} = \frac{\text{fR} * \text{wR}}{2} * \text{Eq}\rho\text{H} + \text{wRR} * \text{Eq}\rho\text{D};$$

$$\text{EqD4a} = 1 - \text{PH} - \text{Exp}\left[-\text{PD} * \frac{\text{avef} * \text{wM}}{\text{W}}\right];$$

$$\text{EqD4b} = 1 - \text{PD} - \text{Exp}\left[-\text{PH} * \frac{\text{wRM}}{2\,\text{W}}\right];$$

```
RepeatMax = 200;
ResultH = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
ResultD = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
ResultBPfix = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
LogResultBPfix = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
repeat = 0;
For[repeat1 = 1, repeat1 ≤ RepeatMax + 1, repeat1 = repeat1 + 1,
  repeat = repeat + 1;
  x1 = 0 + (MaxV - 0)/RepeatMax * (repeat1 - 1);
  CondH = EqD4a /. y → x1;
  CondD = EqD4b /. y → x1;
  sol = Solve[CondH == 0 && CondD == 0 && 0 < PH && PH < 1 && 0 < PD && PD < 1, {PH, PD}, Reals];
  BPPaiH = Part[(PH /. sol), 1];
  BPPaiD = Part[(PD /. sol), 1];
  ResultH[[repeat, 1]] = x1;
  ResultD[[repeat, 1]] = x1;
```

```
    ResultBPfix[[repeat, 1]] = x1;
    LogResultBPfix[[repeat, 1]] = x1;
    ResultH[[repeat, 2]] = BPPaiH;
    ResultD[[repeat, 2]] = BPPaiD;
    ResultBPfix[[repeat, 2]] = EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD;
    LogResultBPfix[[repeat, 2]] = Log[10, EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD];
  ];
gBPfix = ListPlot[ResultBPfix, Joined → True, PlotRange → {{0, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gLogBPfix = ListPlot[LogResultBPfix, Joined → True, PlotRange → {{0, MaxV}, {-4, 0}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
```

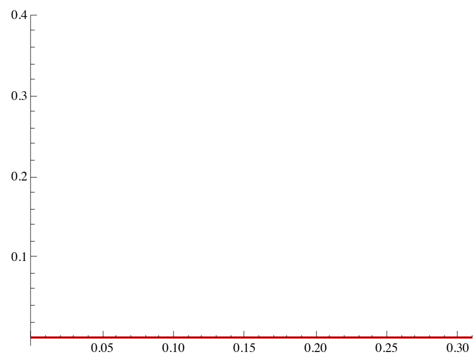
0

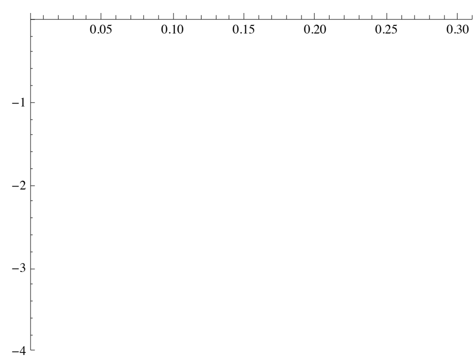
Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

General::stop : この計算中に，Solve::ratnzのこれ以上の出力は表示されません． ≫

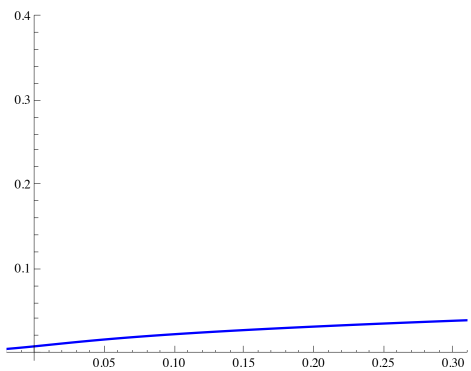


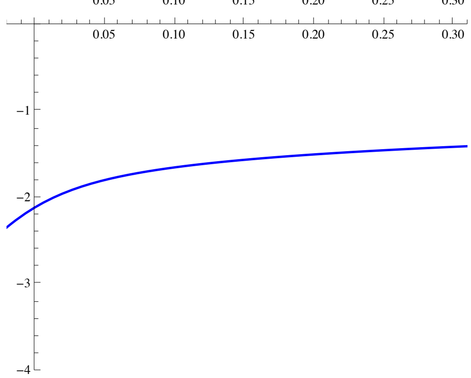


Fixation probability from diffusion approximation

```
Parameters = {SUM → Site, ρ̂H → Sqrt[wRR]/(Sqrt[fR*wR/2] + Sqrt[wRR]), ρ̂D → Sqrt[fR*wR/2]/(Sqrt[fR*wR/2] + Sqrt[wRR]), sfM → SELfM,
   swM → SELwM, swRM → SELwRM, swMM → SELwMM}

snet = sfM/2 + swM + swRM;
Additive = swMM - 2 * swRM;
m = SUM * p * (1 - p) * (snet + p * Additive)/2;
v = (p * (1 - p) * (ρ̂H + 2 * ρ̂D))/(8 * ρ̂H * ρ̂D);
Q = Integrate[m/v /. Parameters, p];
INI = 1/(SUM * (ρ̂H + 2 * ρ̂D)) /. Parameters;
DFfixWS = Integrate[Exp[-2 * Q], {p, 0, INI}]/Integrate[Exp[-2 * Q], {p, 0, 1}];
gDFfixWS = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
   PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS = Plot[Log[10, DFfixWS], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-4, 0}},
   PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

{SUM → 100, ρ̂H → 0.666667, ρ̂D → 0.333333, sfM → 0, swM → 0, swRM → 0, swMM → y}

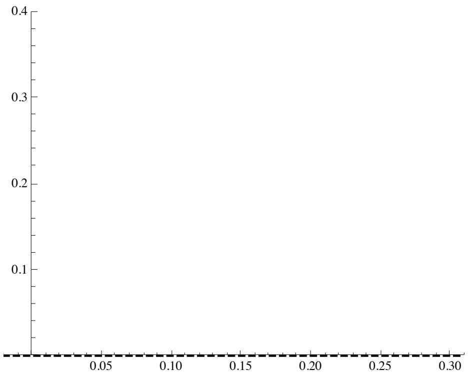


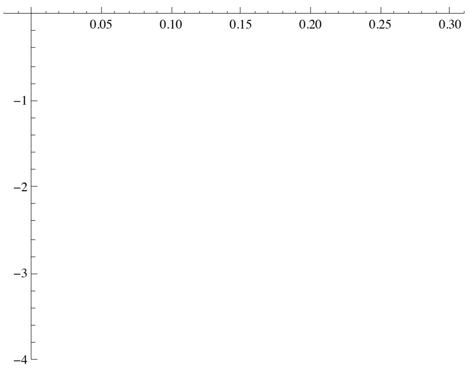


Analytical result (Lineary approximation)

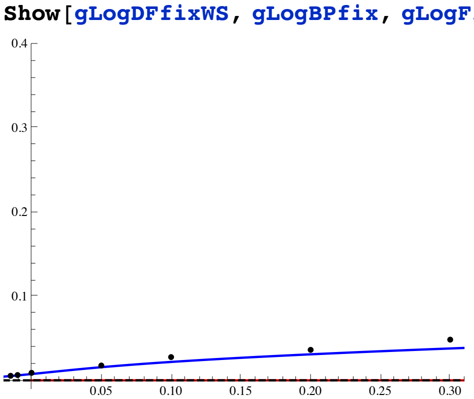
$$\texttt{FixLinear} = \frac{8 * \hat{\rho}_H * \hat{\rho}_D}{\left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)^2} * s_{net};$$

```
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.4}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
gLogFixLinear = Plot[Log[10, FixLinear /. Parameters], {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-4, 0}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```






```
Show[gDFfixWS, gBPfix, gFixLinear, gNumerical]
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical]
```



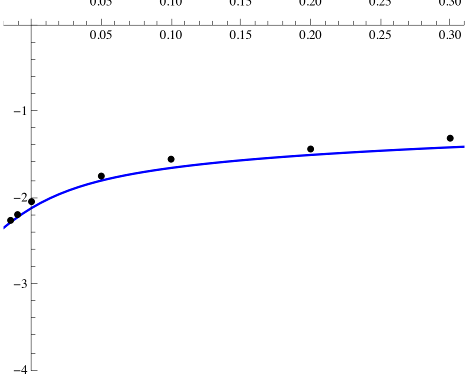
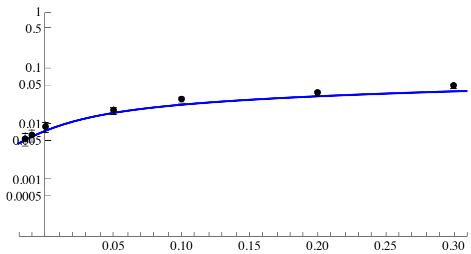


Figure for output

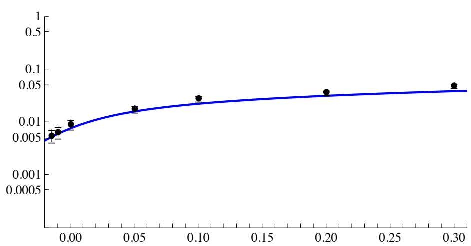
```
axeslabels = {{-5, "0.00001"}, {Log[10, 0.00005], "0.00005"}, {-4, "0.0001"},
   {Log[10, 0.0005], "0.0005"}, {-3, "0.001"}, {Log[10, 0.005], "0.005"},
   {-2, "0.01"}, {Log[10, 0.05], "0.05"}, {-1, "0.1"}, {Log[10, 0.5], "0.5"}, {0, "1"}};
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
 AxesOrigin → {0, -4}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.5]
```

$$\frac{\sqrt{wRR}}{\sqrt{\frac{fR*wR}{2}}+\sqrt{wRR}}$$

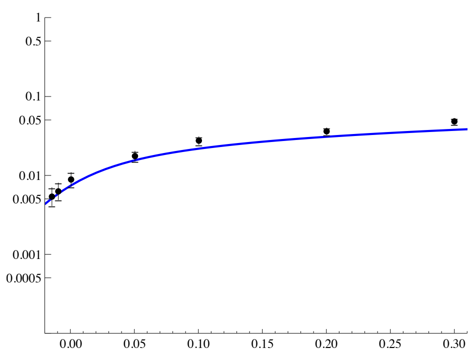


```
0.666667
```

```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
 AxesOrigin → {-0.02, -4}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.5]
```



```
Show[gLogDFfixWS, gLogBPfix, gLogFixLinear, gLogNumerical, gCI1, gCI2, gCI3,
 gCI4, gCI5, gCI6, gCI7, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-4, 0}},
 AxesOrigin → {-0.02, -4}, Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



- **Fig.3(a)**

Parameter values

```
Clear[x]
fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 2 * x;
SELβMM = 0;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$SEL\beta RM = \frac{\beta RM - \beta RR}{\beta RR};$$

$$SELdRM = \frac{dRR - dRM}{dRR};$$

```
MinV = -0.02;
MaxV = 0.21;
```

$$s_{average} = \frac{1}{2} * \left(\frac{SELfM}{2} + SEL\beta M + SELdM\right) + \frac{1}{2} * (SEL\beta MM * HETERO\beta + SELdM * HETEROd)$$

```
0. + x
```

Simulation1 (N = 10)

```
SampleMax = 10 000;
ChangeMax = 4;
OutputData10 = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult10 = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
Site = 10;

VectpM = Table[0, {SampleMax}];
Resultparameter10 = Table[0, {ChangeMax + 1}];
Result10 = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.01];
  If[change == 2, x = 0];
  If[change == 3, x = 0.05];
  If[change == 4, x = 0.1];
  If[change == 5, x = 0.2];

  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,
```

```
EqρH = Sqrt[βRR * dRR] / (Sqrt[(fR*βR*dR)/2] + Sqrt[βRR * dRR]);
xR = Round[Site * EqρH];
xM = 0;
xRR = Site - xR;
xRM = 0;
xMM = 0;
sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];
xR = xR - sampleMutation[[1]];
xM = xM + sampleMutation[[1]];
xRR = xRR - sampleMutation[[2]];
xRM = xRM + sampleMutation[[2]];
xMM = 0;
If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];


step = 2;
While[0 < pM && pM < 1,
  pDeathR = (dR * xR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathM = (dM * xM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathRR = (dRR * xRR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathRM = (dRM * xRM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathMM = (dMM * xMM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  sampleDeath =
   RandomVariate[MultinomialDistribution[1,
     {pDeathR, pDeathM, pDeathRR, pDeathRM, pDeathMM}]];
  xR = xR - sampleDeath[[1]];
  xM = xM - sampleDeath[[2]];
  xRR = xRR - sampleDeath[[3]];
  xRM = xRM - sampleDeath[[4]];
  xMM = xMM - sampleDeath[[5]];

  cR = βRR * xRR + βRM/2 * xRM;
  cM = βMM * xMM + βRM/2 * xRM;
  If[xR == 0 && xM == 0, cRR = 0];
  If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (βR^2 * xR^2)/(βR * xR + βM * xM)];
  If[xR == 0 && xM == 0, cRM = 0];
  If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (βR * βM * xR * xM)/(βR * xR + βM * xM)];
  If[xR == 0 && xM == 0, cMM = 0];
```

```
    If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (βM^2 * xM^2)/(βR * xR + βM * xM)];
    pCellR = cR/(cR + cM + cRR + cRM + cMM);
    pCellM = cM/(cR + cM + cRR + cRM + cMM);
    pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
    pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
    pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
    sampleBirth =
     RandomVariate[MultinomialDistribution[1,
       {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

    xR = xR + sampleBirth[[1]];
    xM = xM + sampleBirth[[2]];
    xRR = xRR + sampleBirth[[3]];
    xRM = xRM + sampleBirth[[4]];
    xMM = xMM + sampleBirth[[5]];

    If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
    If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
    If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

    step = step + 1;
   ];

  VectpM[[sample]] = pM;

  OutputData10[[countData, 1]] = x;
  OutputData10[[countData, 2]] = pM;
  countData = countData + 1;

 ];

 Resultparameter10[[change]] = x;
 Result10[[change]] = Total[VectpM];

 OutputResult10[[countResult, 1]] = x;
 OutputResult10[[countResult, 2]] = Total[VectpM];
 countResult = countResult + 1;

 Print[change]

];
```

1

2

3

4

5

Out put

```
OutputData100 = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult100 = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulation

```
Site = 100;

VectpM = Table[0, {SampleMax}];
Resultparameter100 = Table[0, {ChangeMax + 1}];
Result100 = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.01];
  If[change == 2, x = 0];
  If[change == 3, x = 0.05];
  If[change == 4, x = 0.1];
  If[change == 5, x = 0.2];


  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,
```

$$\text{EqρH} = \frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}};$$

```
   xR = Round[Site * EqρH];
   xM = 0;
   xRR = Site - xR;
   xRM = 0;
   xMM = 0;
```

$$\text{sampleMutation} = \text{RandomVariate}\left[\text{MultinomialDistribution}\left[1, \left\{\frac{xR}{xR + 2 * xRR}, \frac{2 * xRR}{xR + 2 * xRR}\right\}\right]\right];$$

```
   xR = xR - sampleMutation[[1]];
   xM = xM + sampleMutation[[1]];
   xRR = xRR - sampleMutation[[2]];
   xRM = xRM + sampleMutation[[2]];
   xMM = 0;
```

$$\text{If}\left[xRR + xRM + xMM == 0, pM = \frac{xM}{xR + xM}\right];$$

$$\text{If}\left[xR + xM == 0, pM = \frac{\frac{xRM}{2} + xMM}{xRR + xRM + xMM}\right];$$

$$\text{If}\left[xR + xM \neq 0 \;\&\&\; xRR + xRM + xMM \neq 0, pM = \frac{dRR}{dR + dRR} * \frac{\frac{xRM}{2} + xMM}{xRR + xRM + xMM} + \frac{dR}{dR + dRR} * \frac{xM}{xR + xM}\right];$$

```
   step = 2;
   While[0 < pM && pM < 1,
```

$$pDeathR = \frac{dR * xR}{dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM};$$

$$pDeathM = \frac{dM * xM}{dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM};$$

```
pDeathRR = dRR * xRR / (dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
pDeathRM = dRM * xRM / (dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
pDeathMM = dMM * xMM / (dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
sampleDeath =
 RandomVariate[MultinomialDistribution[1,
   {pDeathR, pDeathM, pDeathRR, pDeathRM, pDeathMM}]];
xR = xR - sampleDeath[[1]];
xM = xM - sampleDeath[[2]];
xRR = xRR - sampleDeath[[3]];
xRM = xRM - sampleDeath[[4]];
xMM = xMM - sampleDeath[[5]];

cR = βRR * xRR + βRM/2 * xRM;
cM = βMM * xMM + βRM/2 * xRM;
If[xR == 0 && xM == 0, cRR = 0];
If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * βR^2 * xR^2/(βR * xR + βM * xM)];
If[xR == 0 && xM == 0, cRM = 0];
If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * βR * βM * xR * xM/(βR * xR + βM * xM)];
If[xR == 0 && xM == 0, cMM = 0];
If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * βM^2 * xM^2/(βR * xR + βM * xM)];
pCellR = cR/(cR + cM + cRR + cRM + cMM);
pCellM = cM/(cR + cM + cRR + cRM + cMM);
pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
sampleBirth =
 RandomVariate[MultinomialDistribution[1,
   {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

xR = xR + sampleBirth[[1]];
xM = xM + sampleBirth[[2]];
xRR = xRR + sampleBirth[[3]];
xRM = xRM + sampleBirth[[4]];
xMM = xMM + sampleBirth[[5]];

If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

step = step + 1;
```

```
      ];

      VectpM[[sample]] = pM;

      OutputData100[[countData, 1]] = x;
      OutputData100[[countData, 2]] = pM;
      countData = countData + 1;

     ];

     Resultparameter100[[change]] = x;
     Result100[[change]] = Total[VectpM];

     OutputResult100[[countResult, 1]] = x;
     OutputResult100[[countResult, 2]] = Total[VectpM];
     countResult = countResult + 1;

     Print[change]

    ];
```

1

2

3

4

5

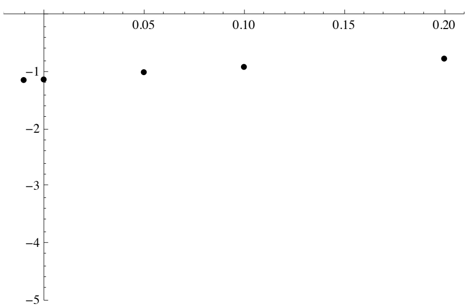
Output of the data

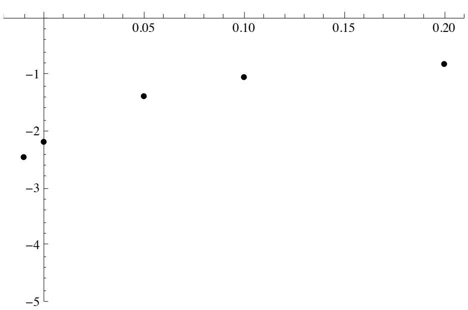
```
Resultparameter10
Result10
Resultparameter100
Result100
Export["/Users/Bessho/Desktop/workstation/OutputDataFig3a_10.txt", OutputData10,
  "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig3a_10.txt", OutputResult10,
  "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputDataFig3a_100.txt", OutputData100,
  "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig3a_100.txt",
  OutputResult100, "Table"];
```

```
{-0.01, 0, 0.05, 0.1, 0.2}
{709., 725., 973., 1204., 1683.}
{-0.01, 0, 0.05, 0.1, 0.2}
{35., 65., 413., 895., 1522.}
```

Plot

```
PlotpM10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpM100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM10[[change, 1]] = Resultparameter10[[change]];
  PlotpM10[[change, 2]] = Result10[[change]] / SampleMax;
  PlotpM100[[change, 1]] = Resultparameter100[[change]];
  PlotpM100[[change, 2]] = Result100[[change]] / SampleMax;
  PlotpMlog10[[change, 1]] = Resultparameter10[[change]];
  PlotpMlog10[[change, 2]] = Log[10, Result10[[change]] / SampleMax];
  PlotpMlog100[[change, 1]] = Resultparameter100[[change]];
  PlotpMlog100[[change, 2]] = Log[10, Result100[[change]] / SampleMax];
 ];
gNumerical10 = ListPlot[PlotpM10, PlotRange → {{MinV, MaxV}, {-0.01, 0.2}},
   PlotStyle → {Thickness[0.005], Black}, AspectRatio → 0.75];
gNumerical100 = ListPlot[PlotpM100, PlotRange → {{MinV, MaxV}, {-0.01, 0.2}},
   PlotStyle → {Thickness[0.005], Black}, AspectRatio → 0.75];
gLogNumerical10 = ListPlot[PlotpMlog10, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Black}]
gLogNumerical100 = ListPlot[PlotpMlog100, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Black}]
PlotpM10
PlotpM100
```






```
{{-0.01, 0.0709}, {0, 0.0725}, {0.05, 0.0973}, {0.1, 0.1204}, {0.2, 0.1683}}
{{-0.01, 0.0035}, {0, 0.0065}, {0.05, 0.0413}, {0.1, 0.0895}, {0.2, 0.1522}}
```

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \bar{z}] \sim 0.95$.

```
z = 1.96;
```

$$\underline{z10} = \frac{n}{n + z^2} \left(\hat{u} + \frac{1}{2 * n} * z^2 - z * \sqrt{\frac{1}{n} * \hat{u} * (1 - \hat{u}) + \frac{1}{4 * n^2} * z^2} \right) /. \; n \to \text{SampleMax} \; /. \; \hat{u} \to \frac{\text{Result10}}{\text{SampleMax}}$$

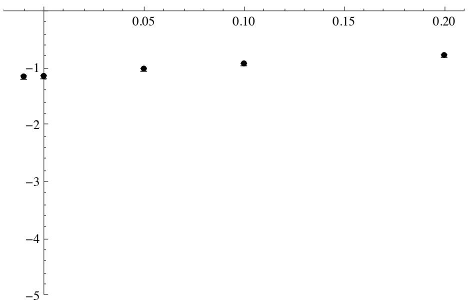
$$\overline{z10} = \frac{n}{n + z^2} \left(\hat{u} + \frac{1}{2 * n} * z^2 + z * \sqrt{\frac{1}{n} * \hat{u} * (1 - \hat{u}) + \frac{1}{4 * n^2} * z^2} \right) /. \; n \to \text{SampleMax} \; /. \; \hat{u} \to \frac{\text{Result10}}{\text{SampleMax}}$$

$$\underline{z100} = \frac{n}{n + z^2} \left(\hat{u} + \frac{1}{2 * n} * z^2 - z * \sqrt{\frac{1}{n} * \hat{u} * (1 - \hat{u}) + \frac{1}{4 * n^2} * z^2} \right) /. \; n \to \text{SampleMax} \; /. \; \hat{u} \to \frac{\text{Result100}}{\text{SampleMax}}$$

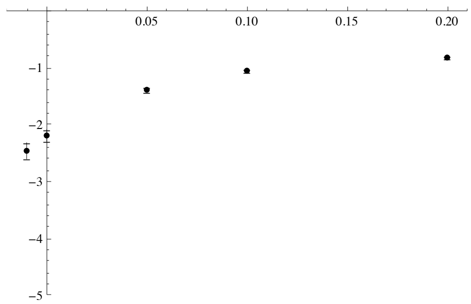
$$\overline{z100} = \frac{n}{n + z^2} \left(\hat{u} + \frac{1}{2 * n} * z^2 + z * \sqrt{\frac{1}{n} * \hat{u} * (1 - \hat{u}) + \frac{1}{4 * n^2} * z^2} \right) /. \; n \to \text{SampleMax} \; /. \; \hat{u} \to \frac{\text{Result100}}{\text{SampleMax}}$$

```
{0.0660326, 0.0675799, 0.0916449, 0.114167, 0.161095}
{0.076097, 0.0777484, 0.103264, 0.126925, 0.17576}
{0.00251776, 0.00510339, 0.0375728, 0.0840614, 0.145293}
{0.00486356, 0.00827563, 0.0453794, 0.0952539, 0.159374}
```

```
PlotCIunder10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder10[[change, 1]] = Resultparameter10[[change]];
  PlotCIupper10[[change, 1]] = Resultparameter10[[change]];
  PlotCIunder10[[change, 2]] = Log[10, z̲1̲0̲[[change]]];
  PlotCIupper10[[change, 2]] = Log[10, z̅1̅0̅[[change]]];
 ];
Log[10, z̅1̅0̅]
N[Log[10, PlotpM10[[All, 2]]]]
Log[10, z̲1̲0̲]
gCIupper10 = ListPlot[PlotCIupper10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder10 = ListPlot[PlotCIunder10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];

PlotCI1♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI2♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI3♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI4♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI5♦10 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1♦10[[1, 1]] = Resultparameter10[[1]];
  PlotCI1♦10[[1, 2]] = Log[10, z̲1̲0̲[[1]]];
  PlotCI1♦10[[2, 1]] = Resultparameter10[[1]];
  PlotCI1♦10[[2, 2]] = Log[10, z̅1̅0̅[[1]]];
  PlotCI2♦10[[1, 1]] = Resultparameter10[[2]];
  PlotCI2♦10[[1, 2]] = Log[10, z̲1̲0̲[[2]]];
  PlotCI2♦10[[2, 1]] = Resultparameter10[[2]];
  PlotCI2♦10[[2, 2]] = Log[10, z̅1̅0̅[[2]]];
  PlotCI3♦10[[1, 1]] = Resultparameter10[[3]];
  PlotCI3♦10[[1, 2]] = Log[10, z̲1̲0̲[[3]]];
  PlotCI3♦10[[2, 1]] = Resultparameter10[[3]];
  PlotCI3♦10[[2, 2]] = Log[10, z̅1̅0̅[[3]]];
  PlotCI4♦10[[1, 1]] = Resultparameter10[[4]];
  PlotCI4♦10[[1, 2]] = Log[10, z̲1̲0̲[[4]]];
  PlotCI4♦10[[2, 1]] = Resultparameter10[[4]];
  PlotCI4♦10[[2, 2]] = Log[10, z̅1̅0̅[[4]]];
  PlotCI5♦10[[1, 1]] = Resultparameter10[[5]];
  PlotCI5♦10[[1, 2]] = Log[10, z̲1̲0̲[[5]]];
  PlotCI5♦10[[2, 1]] = Resultparameter10[[5]];
  PlotCI5♦10[[2, 2]] = Log[10, z̅1̅0̅[[5]]];
 ];
gCI1♦10 = ListPlot[PlotCI1♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2♦10 = ListPlot[PlotCI2♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3♦10 = ListPlot[PlotCI3♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4♦10 = ListPlot[PlotCI4♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5♦10 = ListPlot[PlotCI5♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical10, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10]
{-1.11863, -1.10931, -0.98605, -0.896454, -0.75508}
{-1.14935, -1.13966, -1.01189, -0.919374, -0.773916}
{-1.18024, -1.17018, -1.03789, -0.94246, -0.792919}
```


0.05
0.10
0.15
0.20
−1
−2
−3
−4
−5

```
PlotCIunder100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder100[[change, 1]] = Resultparameter100[[change]];
  PlotCIupper100[[change, 1]] = Resultparameter100[[change]];
  PlotCIunder100[[change, 2]] = Log[10, z100[[change]]];
  PlotCIupper100[[change, 2]] = Log[10, z100[[change]]];
 ];
Log[10, z100]
N[Log[10, PlotpM100[[All, 2]]]]
Log[10, z100]
gCIupper100 = ListPlot[PlotCIupper100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder100 = ListPlot[PlotCIunder100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];


PlotCI1♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI2♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI3♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI4♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI5♦100 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1♦100[[1, 1]] = Resultparameter100[[1]];
  PlotCI1♦100[[1, 2]] = Log[10, z100[[1]]];
  PlotCI1♦100[[2, 1]] = Resultparameter100[[1]];
  PlotCI1♦100[[2, 2]] = Log[10, z100[[1]]];
  PlotCI2♦100[[1, 1]] = Resultparameter100[[2]];
  PlotCI2♦100[[1, 2]] = Log[10, z100[[2]]];
  PlotCI2♦100[[2, 1]] = Resultparameter100[[2]];
  PlotCI2♦100[[2, 2]] = Log[10, z100[[2]]];
  PlotCI3♦100[[1, 1]] = Resultparameter100[[3]];
  PlotCI3♦100[[1, 2]] = Log[10, z100[[3]]];
  PlotCI3♦100[[2, 1]] = Resultparameter100[[3]];
  PlotCI3♦100[[2, 2]] = Log[10, z100[[3]]];
  PlotCI4♦100[[1, 1]] = Resultparameter100[[4]];
  PlotCI4♦100[[1, 2]] = Log[10, z100[[4]]];
  PlotCI4♦100[[2, 1]] = Resultparameter100[[4]];
  PlotCI4♦100[[2, 2]] = Log[10, z100[[4]]];
  PlotCI5♦100[[1, 1]] = Resultparameter100[[5]];
  PlotCI5♦100[[1, 2]] = Log[10, z100[[5]]];
  PlotCI5♦100[[2, 1]] = Resultparameter100[[5]];
  PlotCI5♦100[[2, 2]] = Log[10, z100[[5]]];
 ];
gCI1♦100 = ListPlot[PlotCI1♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2♦100 = ListPlot[PlotCI2♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3♦100 = ListPlot[PlotCI3♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4♦100 = ListPlot[PlotCI4♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5♦100 = ListPlot[PlotCI5♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical100, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100]
{-2.31305, -2.0822, -1.34314, -1.02112, -0.797582}
{-2.45593, -2.18709, -1.38405, -1.04818, -0.817585}
{-2.59899, -2.29214, -1.42513, -1.0754, -0.837755}
```



Analytical result (Branching process approximation)

```
Clear[y]

fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 2 * y;
SELβMM = 0;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SEL}\beta\text{M} + \text{SELdM}\right) + \frac{1}{2} * (\text{SEL}\beta\text{MM} * \text{HETERO}\beta + \text{SELdM} * \text{HETEROd})$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\beta\text{RR} * \text{dRR}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

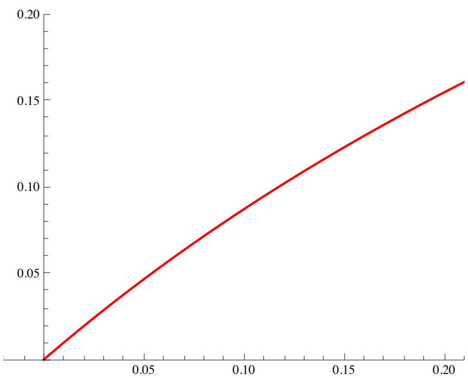
$$\text{FaiR} = \frac{\beta\text{R}}{\text{dR}};$$

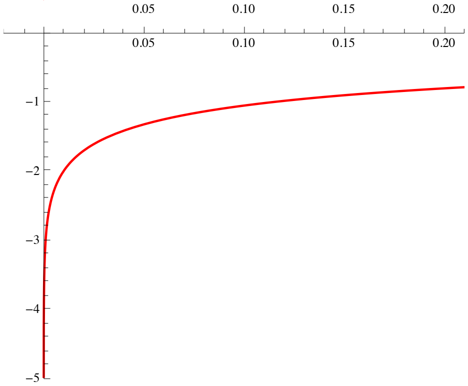
$$\text{FaiM} = \frac{\beta\text{M}}{\text{dM}};$$

$$\text{FaiRR} = \frac{\beta\text{RR}}{\text{dRR}};$$

```
FaiRM = βRM / dRM;
avef = (fM + fR) / 2;
FaiR2 = fR / 2 * FaiR;
FaiM2 = avef / 2 * FaiM;
BPPaiH = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + FaiRM / 2 * Sqrt[FaiR2 * FaiRR]);
BPPaiD = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + 2 * FaiM2 * Sqrt[FaiR2 * FaiRR]);
BPfix = EqρH / (EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD) / (EqρH + 2 * EqρD) * BPPaiD;
gBPfix = Plot[BPfix, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {0, 0.2}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gLogBPfix = Plot[Log[10, BPfix], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
```

```
0. + y
```





Analytical result (Diffusion approximation)

```
Site = 10;

Parameters = {SUM → Site, d_R → dR, d_RR → dRR, ρ̂_H → Sqrt[βRR * dRR] / (Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]),

  ρ̂_D → Sqrt[fR*βR*dR/2] / (Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]), sfM → SELfM, sβM → SELβM, sβRM → SELβRM, sβMM → SELβMM,

  sdM → SELdM, sdRM → SELdRM, sdMM → SELdMM}

Additive = sβMM - 2 * sβRM + sdMM - 2 * sdRM;
```

$$d_{harmonic} = \frac{2}{\frac{1}{d_R} + \frac{1}{d_{RR}}};$$

$$\hat{d} = d_R * \hat{\rho}_H + d_{RR} * (1 - \hat{\rho}_H);$$

$$\overline{d} = \frac{d_R + d_{RR}}{2};$$

$$m = SUM * p * (1 - p) * \frac{2 * s_{average} + p * Additive}{2} * \frac{d_{harmonic}}{\hat{d}};$$

$$v = \frac{p * (1 - p) * (d_R * \hat{\rho}_H + 2 * d_{RR} * \hat{\rho}_D)}{\hat{\rho}_H * \hat{\rho}_D} * \frac{d_{harmonic}}{4 * \hat{d} * \overline{d}};$$

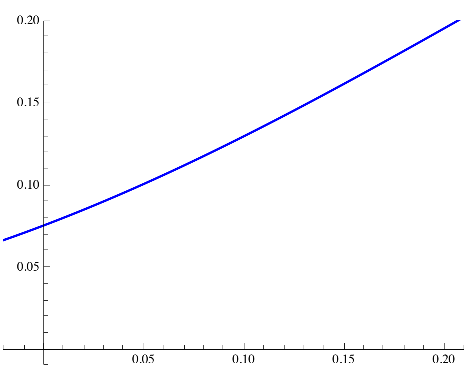
$$Q = Integrate\left[\frac{m}{v} \text{ /. } Parameters, p\right];$$

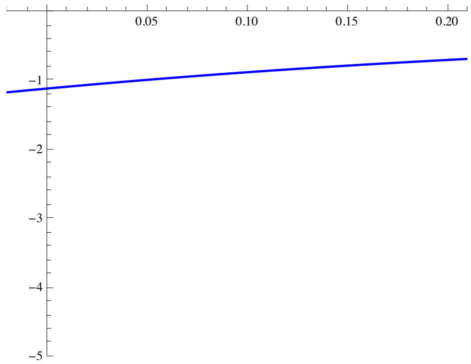
$$INI = \frac{1}{SUM * (\hat{\rho}_H + 2 * \hat{\rho}_D)} \text{ /. } Parameters;$$

$$DFfixWS10 = \frac{Integrate[Exp[-2 * Q], \{p, 0, INI\}]}{Integrate[Exp[-2 * Q], \{p, 0, 1\}]};$$

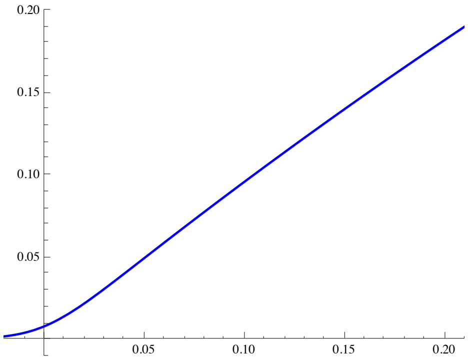
```
gDFfixWS10 = Plot[DFfixWS10, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.2}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS10 = Plot[Log[10, DFfixWS10], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

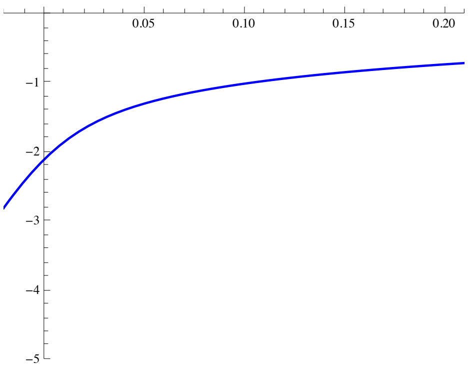
$\{SUM \to 10,\ d_R \to 0.005,\ d_{RR} \to 0.005,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,$
$sfM \to 0,\ s\beta M \to 2\,y,\ s\beta RM \to 0.,\ s\beta MM \to 0,\ sdM \to 0,\ sdRM \to 0.,\ sdMM \to 0\}$






```
Site = 100;

Parameters = {SUM → Site, dR → dR, dRR → dRR, ρ̂H → Sqrt[βRR * dRR] / (Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]),
  ρ̂D → Sqrt[fR*βR*dR/2] / (Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]), sfM → SELfM, sβM → SELβM, sβRM → SELβRM, sβMM → SELβMM,
  sdM → SELdM, sdRM → SELdRM, sdMM → SELdMM}

Additive = sβMM - 2 * sβRM + sdMM - 2 * sdRM;
dharmonic = 2 / (1/dR + 1/dRR);
d̂ = dR * ρ̂H + dRR * (1 - ρ̂H);
d̄ = (dR + dRR) / 2;

m = SUM * p * (1 - p) * (2 * saverage + p * Additive) / 2;
v = p * (1 - p) * (dR/d̄ * ρ̂H + 2 * dRR/d̄ * ρ̂D) / (4 * ρ̂H * ρ̂D);

Q = Integrate[m/v /. Parameters, p];
INI = 1 / (SUM * (ρ̂H + 2 * ρ̂D)) /. Parameters;
DFfixWS100 = Integrate[Exp[-2 * Q], {p, 0, INI}] / Integrate[Exp[-2 * Q], {p, 0, 1}];
gDFfixWS100 = Plot[DFfixWS100, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.2}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS100 = Plot[Log[10, DFfixWS100], {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-5, 0}}, PlotStyle → {Thickness[0.005], Blue},
  AspectRatio → 0.75]
{SUM → 100, dR → 0.005, dRR → 0.005, ρ̂H → 0.666667, ρ̂D → 0.333333,
 sfM → 0, sβM → 2 y, sβRM → 0., sβMM → 0, sdM → 0, sdRM → 0., sdMM → 0}
```

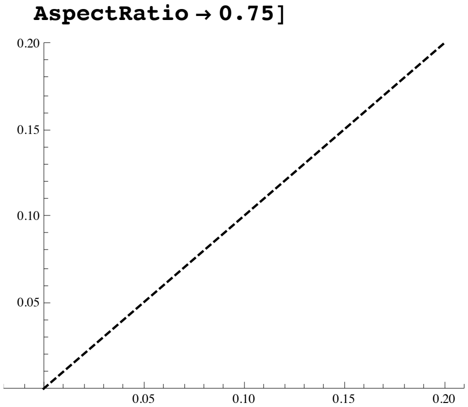


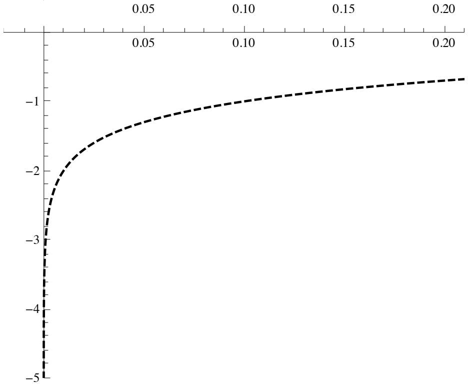


Analytical result (Lineary approximation)

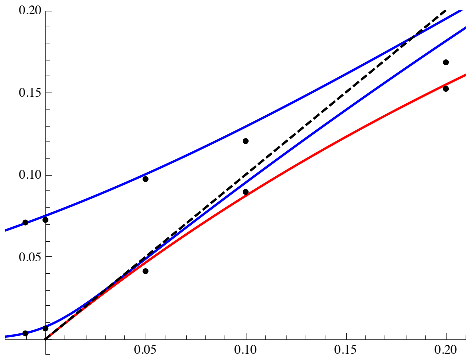
$$\text{FixLinear} = \frac{8 * \hat{\rho}_H * \hat{\rho}_D}{\left(\hat{\rho}_H + 2 * \hat{\rho}_D\right) * \left(\frac{d_R}{\bar{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\bar{d}} * \hat{\rho}_D\right)} * s_{average};$$

```
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.2}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
gLogFixLinear = Plot[Log[10, FixLinear] /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-5, 0}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```





```
Show[gDFfixWS10, gDFfixWS100, gBPfix, gFixLinear, gNumerical10, gNumerical100]
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100]
```



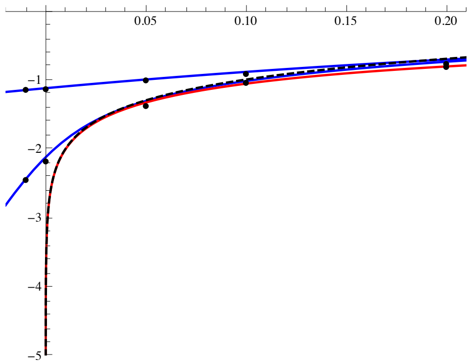
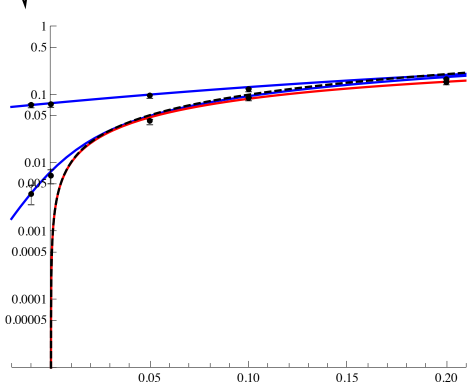


Figure for output

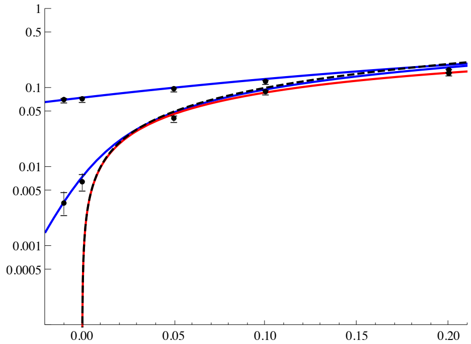
```
axeslabels = {{-5, "0.00001"}, {Log[10, 0.00005], "0.00005"}, {-4, "0.0001"},
   {Log[10, 0.0005], "0.0005"}, {-3, "0.001"}, {Log[10, 0.005], "0.005"},
   {-2, "0.01"}, {Log[10, 0.05], "0.05"}, {-1, "0.1"}, {Log[10, 0.5], "0.5"}, {0, "1"}};

Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100, PlotRange → {{MinV, MaxV}, {-5, 0}}, AxesOrigin → {0, -5},
 Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```

$$\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}}$$

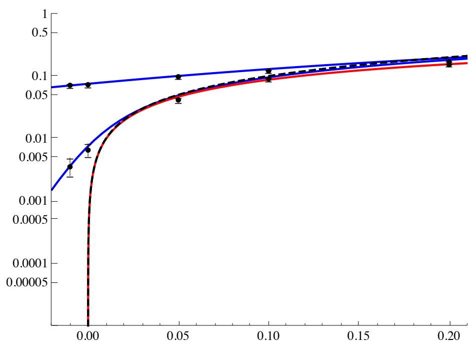



```
0.666667
```

```
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100, PlotRange → {{MinV, MaxV}, {-4, 0}}, AxesOrigin → {-0.02, -4},
 Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```




```
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100, PlotRange → {{MinV, MaxV}, {-5, 0}}, AxesOrigin → {-0.02, -5},
 Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



$N\left[\frac{\text{Result10}}{\text{SampleMax}}\right]$

$N\left[\frac{\text{Result100}}{\text{SampleMax}}\right]$

{0.0709, 0.0725, 0.0973, 0.1204, 0.1683}

{0.0035, 0.0065, 0.0413, 0.0895, 0.1522}

- **Fig.3(b)**

Parameter values

```
Clear[x]
fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 0;
SELβMM = 4 * x;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$SEL\beta RM = \frac{\beta RM - \beta RR}{\beta RR};$$

$$SELdRM = \frac{dRR - dRM}{dRR};$$

```
MinV = -0.02;
MaxV = 0.21;
```

$$s_{average} = \frac{1}{2} * \left(\frac{SELfM}{2} + SEL\beta M + SELdM\right) + \frac{1}{2} * (SEL\beta MM * HETERO\beta + SELdM * HETEROd)$$

```
1. x
```

Simulation1 (N = 10)

```
SampleMax = 10 000;
ChangeMax = 4;
OutputData10 = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult10 = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
Site = 10;

VectpM = Table[0, {SampleMax}];
Resultparameter10 = Table[0, {ChangeMax + 1}];
Result10 = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.01];
  If[change == 2, x = 0];
  If[change == 3, x = 0.05];
  If[change == 4, x = 0.1];
  If[change == 5, x = 0.2];

  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,
```

```
EqρH = Sqrt[βRR * dRR] / (Sqrt[(fR * βR * dR)/2] + Sqrt[βRR * dRR]);
xR = Round[Site * EqρH];
xM = 0;
xRR = Site - xR;
xRM = 0;
xMM = 0;
sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];
xR = xR - sampleMutation[[1]];
xM = xM + sampleMutation[[1]];
xRR = xRR - sampleMutation[[2]];
xRM = xRM + sampleMutation[[2]];
xMM = 0;
If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];


step = 2;
While[0 < pM && pM < 1,
 pDeathR = (dR * xR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 pDeathM = (dM * xM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 pDeathRR = (dRR * xRR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 pDeathRM = (dRM * xRM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 pDeathMM = (dMM * xMM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 sampleDeath =
  RandomVariate[MultinomialDistribution[1,
    {pDeathR, pDeathM, pDeathRR, pDeathRM, pDeathMM}]];
 xR = xR - sampleDeath[[1]];
 xM = xM - sampleDeath[[2]];
 xRR = xRR - sampleDeath[[3]];
 xRM = xRM - sampleDeath[[4]];
 xMM = xMM - sampleDeath[[5]];

 cR = βRR * xRR + βRM/2 * xRM;
 cM = βMM * xMM + βRM/2 * xRM;
 If[xR == 0 && xM == 0, cRR = 0];
 If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (βR^2 * xR^2)/(βR * xR + βM * xM)];
 If[xR == 0 && xM == 0, cRM = 0];
 If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (βR * βM * xR * xM)/(βR * xR + βM * xM)];
 If[xR == 0 && xM == 0, cMM = 0];
```

```
      If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (βM^2 * xM^2)/(βR * xR + βM * xM)];
      pCellR = cR/(cR + cM + cRR + cRM + cMM);
      pCellM = cM/(cR + cM + cRR + cRM + cMM);
      pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
      pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
      pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
      sampleBirth =
       RandomVariate[MultinomialDistribution[1,
          {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

      xR = xR + sampleBirth[[1]];
      xM = xM + sampleBirth[[2]];
      xRR = xRR + sampleBirth[[3]];
      xRM = xRM + sampleBirth[[4]];
      xMM = xMM + sampleBirth[[5]];

      If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
      If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
      If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

      step = step + 1;
     ];

    VectpM[[sample]] = pM;

    OutputData10[[countData, 1]] = x;
    OutputData10[[countData, 2]] = pM;
    countData = countData + 1;

   ];

  Resultparameter10[[change]] = x;
  Result10[[change]] = Total[VectpM];

  OutputResult10[[countResult, 1]] = x;
  OutputResult10[[countResult, 2]] = Total[VectpM];
  countResult = countResult + 1;

  Print[change]

 ];
```

1

2

3

4

5

Out put

```
OutputData100 = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult100 = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulation

```
Site = 100;

VectpM = Table[0, {SampleMax}];
Resultparameter100 = Table[0, {ChangeMax + 1}];
Result100 = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.01];
  If[change == 2, x = 0];
  If[change == 3, x = 0.05];
  If[change == 4, x = 0.1];
  If[change == 5, x = 0.2];


  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,
```

$$\text{EqρH} = \frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}};$$

```
   xR = Round[Site * EqρH];
   xM = 0;
   xRR = Site - xR;
   xRM = 0;
   xMM = 0;
```

$$\text{sampleMutation} = \text{RandomVariate}\left[\text{MultinomialDistribution}\left[1, \left\{\frac{xR}{xR + 2 * xRR}, \frac{2 * xRR}{xR + 2 * xRR}\right\}\right]\right];$$

```
   xR = xR - sampleMutation[[1]];
   xM = xM + sampleMutation[[1]];
   xRR = xRR - sampleMutation[[2]];
   xRM = xRM + sampleMutation[[2]];
   xMM = 0;
```

$$\text{If}\left[xRR + xRM + xMM == 0, pM = \frac{xM}{xR + xM}\right];$$

$$\text{If}\left[xR + xM == 0, pM = \frac{\frac{xRM}{2} + xMM}{xRR + xRM + xMM}\right];$$

$$\text{If}\left[xR + xM \neq 0 \,\&\&\, xRR + xRM + xMM \neq 0, pM = \frac{dRR}{dR + dRR} * \frac{\frac{xRM}{2} + xMM}{xRR + xRM + xMM} + \frac{dR}{dR + dRR} * \frac{xM}{xR + xM}\right];$$

```
   step = 2;
   While[0 < pM && pM < 1,
```

$$pDeathR = \frac{dR * xR}{dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM};$$

$$pDeathM = \frac{dM * xM}{dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM};$$

```
pDeathRR = (dRR * xRR) / (dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
pDeathRM = (dRM * xRM) / (dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
pDeathMM = (dMM * xMM) / (dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
sampleDeath =
 RandomVariate[MultinomialDistribution[1,
   {pDeathR, pDeathM, pDeathRR, pDeathRM, pDeathMM}]];
xR = xR - sampleDeath[[1]];
xM = xM - sampleDeath[[2]];
xRR = xRR - sampleDeath[[3]];
xRM = xRM - sampleDeath[[4]];
xMM = xMM - sampleDeath[[5]];

cR = βRR * xRR + βRM/2 * xRM;
cM = βMM * xMM + βRM/2 * xRM;
If[xR == 0 && xM == 0, cRR = 0];
If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (βR^2 * xR^2)/(βR * xR + βM * xM)];
If[xR == 0 && xM == 0, cRM = 0];
If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (βR * βM * xR * xM)/(βR * xR + βM * xM)];
If[xR == 0 && xM == 0, cMM = 0];
If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (βM^2 * xM^2)/(βR * xR + βM * xM)];
pCellR = cR / (cR + cM + cRR + cRM + cMM);
pCellM = cM / (cR + cM + cRR + cRM + cMM);
pCellRR = cRR / (cR + cM + cRR + cRM + cMM);
pCellRM = cRM / (cR + cM + cRR + cRM + cMM);
pCellMM = cMM / (cR + cM + cRR + cRM + cMM);
sampleBirth =
 RandomVariate[MultinomialDistribution[1,
   {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

xR = xR + sampleBirth[[1]];
xM = xM + sampleBirth[[2]];
xRR = xRR + sampleBirth[[3]];
xRM = xRM + sampleBirth[[4]];
xMM = xMM + sampleBirth[[5]];

If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

step = step + 1;
```

```
      ];

      VectpM[[sample]] = pM;

      OutputData100[[countData, 1]] = x;
      OutputData100[[countData, 2]] = pM;
      countData = countData + 1;

     ];

     Resultparameter100[[change]] = x;
     Result100[[change]] = Total[VectpM];

     OutputResult100[[countResult, 1]] = x;
     OutputResult100[[countResult, 2]] = Total[VectpM];
     countResult = countResult + 1;

     Print[change]

    ];
```

1

2

3

4

5

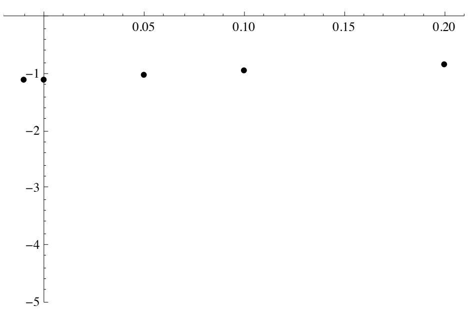
Output of the data

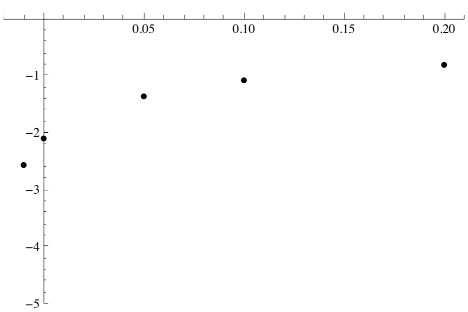
```
Resultparameter10
Result10
Resultparameter100
Result100
Export["/Users/Bessho/Desktop/workstation/OutputDataFig3b_10.txt", OutputData10,
  "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig3b_10.txt", OutputResult10,
  "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputDataFig3b_100.txt", OutputData100,
  "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig3b_100.txt",
  OutputResult100, "Table"];
```

```
{-0.01, 0, 0.05, 0.1, 0.2}
{779., 780., 949., 1135., 1449.}
{-0.01, 0, 0.05, 0.1, 0.2}
{27., 79., 432., 828., 1542.}
```

Plot

```
PlotpM10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpM100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM10[[change, 1]] = Resultparameter10[[change]];
  PlotpM10[[change, 2]] = Result10[[change]] / SampleMax;
  PlotpM100[[change, 1]] = Resultparameter100[[change]];
  PlotpM100[[change, 2]] = Result100[[change]] / SampleMax;
  PlotpMlog10[[change, 1]] = Resultparameter10[[change]];
  PlotpMlog10[[change, 2]] = Log[10, Result10[[change]] / SampleMax];
  PlotpMlog100[[change, 1]] = Resultparameter100[[change]];
  PlotpMlog100[[change, 2]] = Log[10, Result100[[change]] / SampleMax];
 ];
gNumerical10 = ListPlot[PlotpM10, PlotRange → {{MinV, MaxV}, {-0.01, 0.2}},
   PlotStyle → {Thickness[0.005], Black}, AspectRatio → 0.75];
gNumerical100 = ListPlot[PlotpM100, PlotRange → {{MinV, MaxV}, {-0.01, 0.2}},
   PlotStyle → {Thickness[0.005], Black}, AspectRatio → 0.75];
gLogNumerical10 = ListPlot[PlotpMlog10, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Black}]
gLogNumerical100 = ListPlot[PlotpMlog100, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Black}]
PlotpM10
PlotpM100
```






```
{{-0.01, 0.0779}, {0, 0.078}, {0.05, 0.0949}, {0.1, 0.1135}, {0.2, 0.1449}}
{{-0.01, 0.0027}, {0, 0.0079}, {0.05, 0.0432}, {0.1, 0.0828}, {0.2, 0.1542}}
```

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \overline{z}] \sim 0.95$.

z = 1.96;

$$\underline{z10} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*\left(1-\hat{u}\right) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to \text{SampleMax} \text{ /. } \hat{u} \to \frac{\text{Result10}}{\text{SampleMax}}$$

$$\overline{z10} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*\left(1-\hat{u}\right) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to \text{SampleMax} \text{ /. } \hat{u} \to \frac{\text{Result10}}{\text{SampleMax}}$$

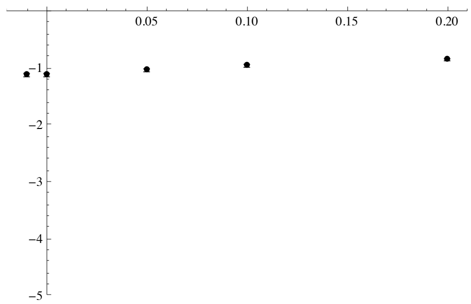
$$\underline{z100} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*\left(1-\hat{u}\right) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to \text{SampleMax} \text{ /. } \hat{u} \to \frac{\text{Result100}}{\text{SampleMax}}$$

$$\overline{z100} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*\left(1-\hat{u}\right) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to \text{SampleMax} \text{ /. } \hat{u} \to \frac{\text{Result100}}{\text{SampleMax}}$$

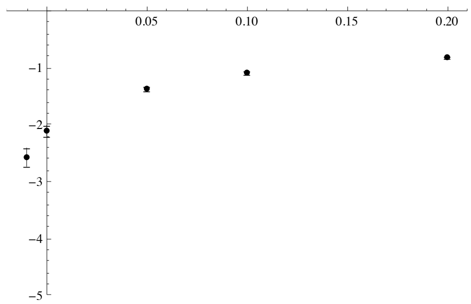
```
{0.0728075, 0.0729044, 0.0893103, 0.107431, 0.138137}
{0.0833167, 0.0834197, 0.100801, 0.119866, 0.151936}
{0.00185632, 0.00634385, 0.0393875, 0.0775575, 0.147255}
{0.00392562, 0.00983409, 0.0473633, 0.0883629, 0.161411}
```

```
PlotCIunder10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder10[[change, 1]] = Resultparameter10[[change]];
  PlotCIupper10[[change, 1]] = Resultparameter10[[change]];
  PlotCIunder10[[change, 2]] = Log[10, UnderBar[z10][[change]]];
  PlotCIupper10[[change, 2]] = Log[10, OverBar[z10][[change]]];
 ];
Log[10, OverBar[z10]]
N[Log[10, PlotpM10[[All, 2]]]]
Log[10, UnderBar[z10]]
gCIupper10 = ListPlot[PlotCIupper10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder10 = ListPlot[PlotCIunder10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];

PlotCI1♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI2♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI3♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI4♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI5♦10 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1♦10[[1, 1]] = Resultparameter10[[1]];
  PlotCI1♦10[[1, 2]] = Log[10, UnderBar[z10][[1]]];
  PlotCI1♦10[[2, 1]] = Resultparameter10[[1]];
  PlotCI1♦10[[2, 2]] = Log[10, OverBar[z10][[1]]];
  PlotCI2♦10[[1, 1]] = Resultparameter10[[2]];
  PlotCI2♦10[[1, 2]] = Log[10, UnderBar[z10][[2]]];
  PlotCI2♦10[[2, 1]] = Resultparameter10[[2]];
  PlotCI2♦10[[2, 2]] = Log[10, OverBar[z10][[2]]];
  PlotCI3♦10[[1, 1]] = Resultparameter10[[3]];
  PlotCI3♦10[[1, 2]] = Log[10, UnderBar[z10][[3]]];
  PlotCI3♦10[[2, 1]] = Resultparameter10[[3]];
  PlotCI3♦10[[2, 2]] = Log[10, OverBar[z10][[3]]];
  PlotCI4♦10[[1, 1]] = Resultparameter10[[4]];
  PlotCI4♦10[[1, 2]] = Log[10, UnderBar[z10][[4]]];
  PlotCI4♦10[[2, 1]] = Resultparameter10[[4]];
  PlotCI4♦10[[2, 2]] = Log[10, OverBar[z10][[4]]];
  PlotCI5♦10[[1, 1]] = Resultparameter10[[5]];
  PlotCI5♦10[[1, 2]] = Log[10, UnderBar[z10][[5]]];
  PlotCI5♦10[[2, 1]] = Resultparameter10[[5]];
  PlotCI5♦10[[2, 2]] = Log[10, OverBar[z10][[5]]];
 ];
gCI1♦10 = ListPlot[PlotCI1♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2♦10 = ListPlot[PlotCI2♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3♦10 = ListPlot[PlotCI3♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4♦10 = ListPlot[PlotCI4♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5♦10 = ListPlot[PlotCI5♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical10, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10]
```

```
{-1.07927, -1.07873, -0.996536, -0.921303, -0.81834}
{-1.10846, -1.10791, -1.02273, -0.945004, -0.838932}
{-1.13782, -1.13725, -1.0491, -0.968872, -0.85969}
```


0.05
0.10
0.15
0.20
−1
−2
−3
−4
−5

```
PlotCIunder100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder100[[change, 1]] = Resultparameter100[[change]];
  PlotCIupper100[[change, 1]] = Resultparameter100[[change]];
  PlotCIunder100[[change, 2]] = Log[10, z100[[change]]];
  PlotCIupper100[[change, 2]] = Log[10, z100[[change]]];
 ];
Log[10, z100]
N[Log[10, PlotpM100[[All, 2]]]]
Log[10, z100]
gCIupper100 = ListPlot[PlotCIupper100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder100 = ListPlot[PlotCIunder100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];


PlotCI1♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI2♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI3♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI4♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI5♦100 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1♦100[[1, 1]] = Resultparameter100[[1]];
  PlotCI1♦100[[1, 2]] = Log[10, z100[[1]]];
  PlotCI1♦100[[2, 1]] = Resultparameter100[[1]];
  PlotCI1♦100[[2, 2]] = Log[10, z100[[1]]];
  PlotCI2♦100[[1, 1]] = Resultparameter100[[2]];
  PlotCI2♦100[[1, 2]] = Log[10, z100[[2]]];
  PlotCI2♦100[[2, 1]] = Resultparameter100[[2]];
  PlotCI2♦100[[2, 2]] = Log[10, z100[[2]]];
  PlotCI3♦100[[1, 1]] = Resultparameter100[[3]];
  PlotCI3♦100[[1, 2]] = Log[10, z100[[3]]];
  PlotCI3♦100[[2, 1]] = Resultparameter100[[3]];
  PlotCI3♦100[[2, 2]] = Log[10, z100[[3]]];
  PlotCI4♦100[[1, 1]] = Resultparameter100[[4]];
  PlotCI4♦100[[1, 2]] = Log[10, z100[[4]]];
  PlotCI4♦100[[2, 1]] = Resultparameter100[[4]];
  PlotCI4♦100[[2, 2]] = Log[10, z100[[4]]];
  PlotCI5♦100[[1, 1]] = Resultparameter100[[5]];
  PlotCI5♦100[[1, 2]] = Log[10, z100[[5]]];
  PlotCI5♦100[[2, 1]] = Resultparameter100[[5]];
  PlotCI5♦100[[2, 2]] = Log[10, z100[[5]]];
 ];
gCI1♦100 = ListPlot[PlotCI1♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2♦100 = ListPlot[PlotCI2♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3♦100 = ListPlot[PlotCI3♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4♦100 = ListPlot[PlotCI4♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5♦100 = ListPlot[PlotCI5♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical100, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100]
{-2.40609, -2.00727, -1.32456, -1.05373, -0.792067}
{-2.56864, -2.10237, -1.36452, -1.08197, -0.811916}
{-2.73135, -2.19765, -1.40464, -1.11038, -0.831931}
```



Analytical result (Branching process approximation)

```
Clear[y]

fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 0;
SELβMM = 4 * y;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

$$\text{s}_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SEL}\beta\text{M} + \text{SELdM}\right) + \frac{1}{2} * (\text{SEL}\beta\text{MM} * \text{HETERO}\beta + \text{SELdM} * \text{HETEROd})$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\beta\text{RR} * \text{dRR}}}{\sqrt{\frac{\text{fR} * \beta\text{R} * \text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

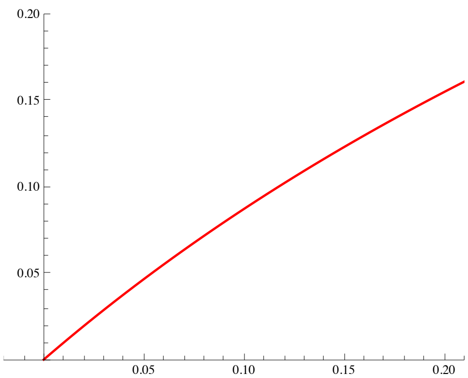
$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR} * \beta\text{R} * \text{dR}}{2}}}{\sqrt{\frac{\text{fR} * \beta\text{R} * \text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

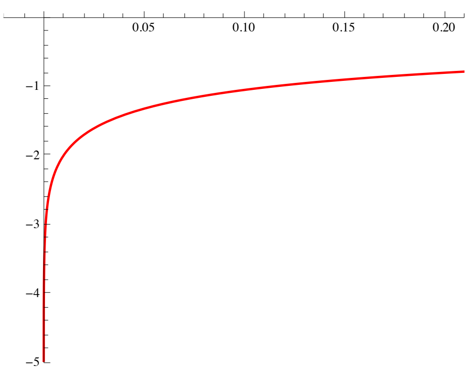
$$\text{FaiR} = \frac{\beta\text{R}}{\text{dR}};$$

$$\text{FaiM} = \frac{\beta\text{M}}{\text{dM}};$$

$$\text{FaiRR} = \frac{\beta\text{RR}}{\text{dRR}};$$

```
FaiRM = βRM/dRM;
avef = (fM + fR)/2;
FaiR2 = fR/2 * FaiR;
FaiM2 = avef/2 * FaiM;
BPPaiH = (FaiM2 * FaiRM - FaiR2 * FaiRR)/(FaiM2 * FaiRM + FaiRM/2 * Sqrt[FaiR2 * FaiRR]);
BPPaiD = (FaiM2 * FaiRM - FaiR2 * FaiRR)/(FaiM2 * FaiRM + 2 * FaiM2 * Sqrt[FaiR2 * FaiRR]);
BPfix = EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD;
gBPfix = Plot[BPfix, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {0, 0.2}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gLogBPfix = Plot[Log[10, BPfix], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
1. y
```





Analytical result (Diffusion approximation)

Site = 10;

$$\text{Parameters} = \left\{\text{SUM} \to \text{Site},\ d_R \to \text{dR},\ d_{RR} \to \text{dRR},\ \hat{\rho}_H \to \frac{\sqrt{\beta\text{RR} * \text{dRR}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}},\ \hat{\rho}_D \to \frac{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}}}{\sqrt{\frac{\text{fR}*\beta\text{R}*\text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}},\ \text{sfM} \to \text{SELfM},\ s\beta\text{M} \to \text{SEL}\beta\text{M},\ s\beta\text{RM} \to \text{SEL}\beta\text{RM},\ s\beta\text{MM} \to \text{SEL}\beta\text{MM},\ \text{sdM} \to \text{SELdM},\ \text{sdRM} \to \text{SELdRM},\ \text{sdMM} \to \text{SELdMM}\right\}$$

Additive = sβMM - 2 * sβRM + sdMM - 2 * sdRM;

$$d_{harmonic} = \frac{2}{\frac{1}{d_R} + \frac{1}{d_{RR}}};$$

$$\hat{d} = d_R * \hat{\rho}_H + d_{RR} * (1 - \hat{\rho}_H);$$

$$\overline{d} = \frac{d_R + d_{RR}}{2};$$

$$m = \text{SUM} * p * (1 - p) * \frac{2 * s_{average} + p * \text{Additive}}{2} * \frac{d_{harmonic}}{\hat{d}};$$

$$v = \frac{p * (1 - p) * (d_R * \hat{\rho}_H + 2 * d_{RR} * \hat{\rho}_D)}{\hat{\rho}_H * \hat{\rho}_D} * \frac{d_{harmonic}}{4 * \hat{d} * \overline{d}};$$

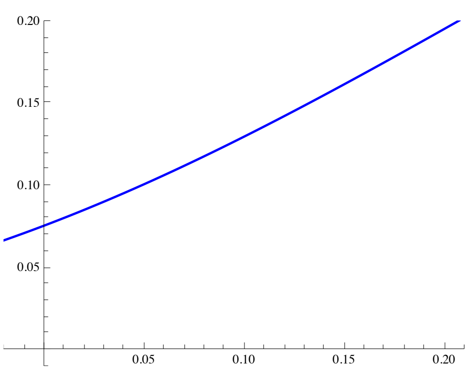
$$Q = \text{Integrate}\left[\frac{m}{v} \text{ /. Parameters},\ p\right];$$

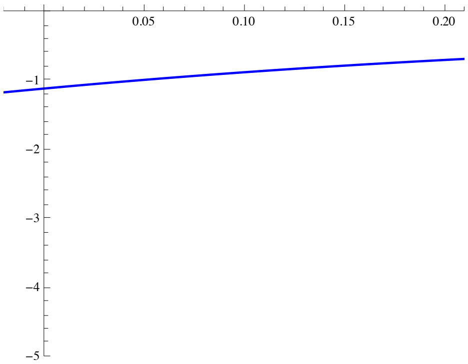
$$\text{INI} = \frac{1}{\text{SUM} * (\hat{\rho}_H + 2 * \hat{\rho}_D)} \text{ /. Parameters};$$

$$\text{DFfixWS10} = \frac{\text{Integrate}[\text{Exp}[-2 * Q], \{p, 0, \text{INI}\}]}{\text{Integrate}[\text{Exp}[-2 * Q], \{p, 0, 1\}]};$$

```
gDFfixWS10 = Plot[DFfixWS10, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.2}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS10 = Plot[Log[10, DFfixWS10], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

$$\left\{\text{SUM} \to 10,\ d_R \to 0.005,\ d_{RR} \to 0.005,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,\ \text{sfM} \to 0,\ s\beta\text{M} \to 0,\ s\beta\text{RM} \to \frac{-1000 + 1000\ (1 + 2.\ y)}{1000},\ s\beta\text{MM} \to 4\ y,\ \text{sdM} \to 0,\ \text{sdRM} \to 0.,\ \text{sdMM} \to 0\right\}$$






```
Site = 100;
```

$$\text{Parameters} = \left\{\text{SUM} \to \text{Site},\ d_R \to \text{dR},\ d_{RR} \to \text{dRR},\ \hat{\rho}_H \to \frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}},\ \hat{\rho}_D \to \frac{\sqrt{\frac{fR*\beta R*dR}{2}}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}},\ sfM \to SELfM,\ s\beta M \to SEL\beta M,\ s\beta RM \to SEL\beta RM,\ s\beta MM \to SEL\beta MM,\ sdM \to SELdM,\ sdRM \to SELdRM,\ sdMM \to SELdMM\right\}$$

$$\text{Additive} = s\beta MM - 2 * s\beta RM + sdMM - 2 * sdRM;$$

$$d_{harmonic} = \frac{2}{\frac{1}{d_R} + \frac{1}{d_{RR}}};$$

$$\hat{d} = d_R * \hat{\rho}_H + d_{RR} * \left(1 - \hat{\rho}_H\right);$$

$$\overline{d} = \frac{d_R + d_{RR}}{2};$$

$$m = \text{SUM} * p * (1 - p) * \frac{2 * s_{average} + p * \text{Additive}}{2};$$

$$v = \frac{p * (1 - p) * \left(\frac{d_R}{\hat{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\hat{d}} * \hat{\rho}_D\right)}{4 * \hat{\rho}_H * \hat{\rho}_D};$$

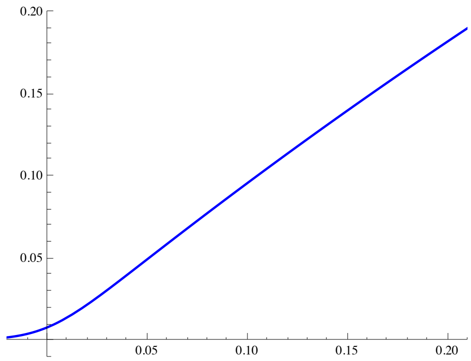
$$Q = \text{Integrate}\left[\frac{m}{v} \text{ /. Parameters},\ p\right];$$

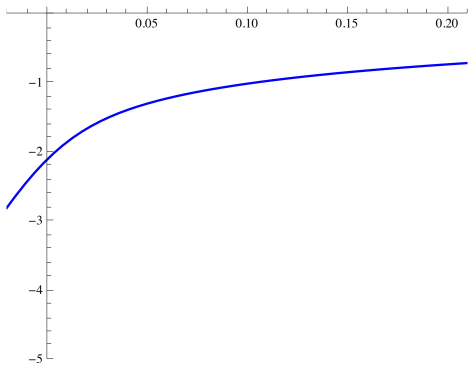
$$\text{INI} = \frac{1}{\text{SUM} * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)} \text{ /. Parameters};$$

$$\text{DFfixWS100} = \frac{\text{Integrate}[\text{Exp}[-2 * Q],\ \{p,\ 0,\ \text{INI}\}]}{\text{Integrate}[\text{Exp}[-2 * Q],\ \{p,\ 0,\ 1\}]};$$

```
gDFfixWS100 = Plot[DFfixWS100, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.2}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS100 = Plot[Log[10, DFfixWS100], {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-5, 0}}, PlotStyle → {Thickness[0.005], Blue},
  AspectRatio → 0.75]
```

$$\left\{\text{SUM} \to 100,\ d_R \to 0.005,\ d_{RR} \to 0.005,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,\ sfM \to 0,\ s\beta M \to 0,\ s\beta RM \to \frac{-1000 + 1000\ (1 + 2.\ y)}{1000},\ s\beta MM \to 4\ y,\ sdM \to 0,\ sdRM \to 0.,\ sdMM \to 0\right\}$$

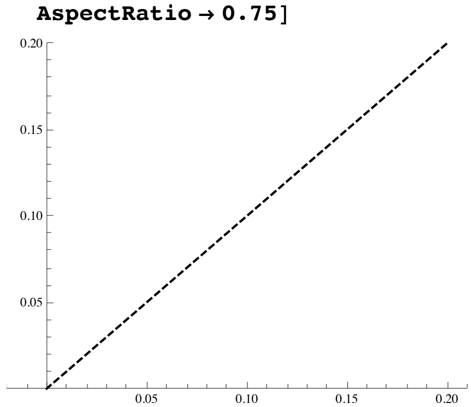


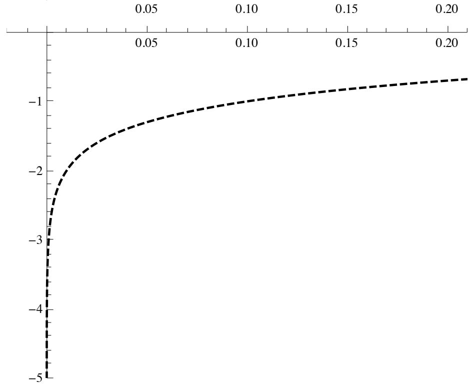


Analytical result (Lineary approximation)

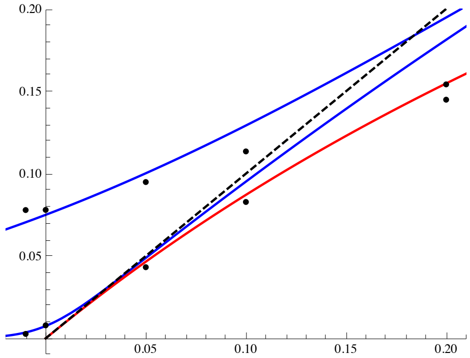
$$\mathbf{FixLinear} = \frac{8 * \hat{\rho}_H * \hat{\rho}_D}{\left(\hat{\rho}_H + 2 * \hat{\rho}_D\right) * \left(\frac{d_R}{d} * \hat{\rho}_H + 2 * \frac{d_{RR}}{d} * \hat{\rho}_D\right)} * s_{average};$$

```
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.2}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
gLogFixLinear = Plot[Log[10, FixLinear] /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-5, 0}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```





```
Show[gDFfixWS10, gDFfixWS100, gBPfix, gFixLinear, gNumerical10, gNumerical100]
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100]
```



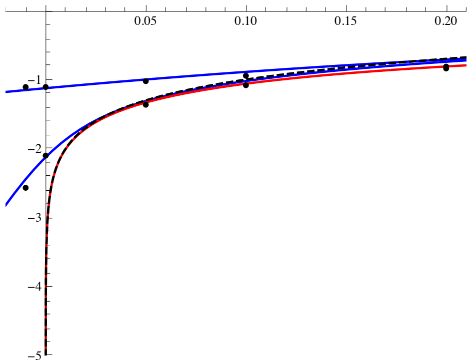
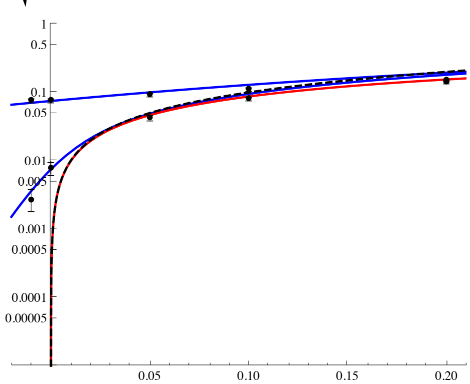


Figure for output

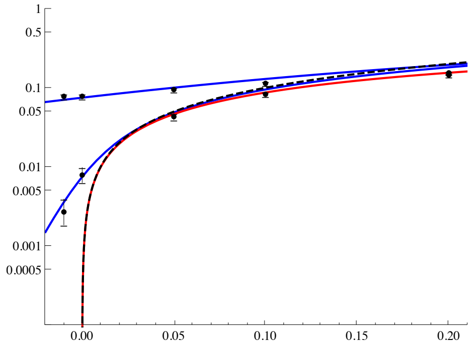
```
axeslabels = {{-5, "0.00001"}, {Log[10, 0.00005], "0.00005"}, {-4, "0.0001"},
   {Log[10, 0.0005], "0.0005"}, {-3, "0.001"}, {Log[10, 0.005], "0.005"},
   {-2, "0.01"}, {Log[10, 0.05], "0.05"}, {-1, "0.1"}, {Log[10, 0.5], "0.5"}, {0, "1"}};
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100, PlotRange → {{MinV, MaxV}, {-5, 0}}, AxesOrigin → {0, -5},
 Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```

$$\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR * \beta R * dR}{2}} + \sqrt{\beta RR * dRR}}$$

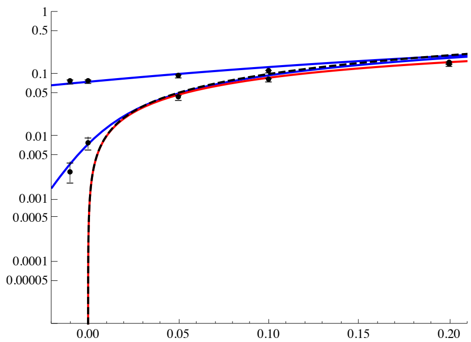


```
0.666667
```

```
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100, PlotRange → {{MinV, MaxV}, {-4, 0}}, AxesOrigin → {-0.02, -4},
 Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```




```
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100, PlotRange → {{MinV, MaxV}, {-5, 0}}, AxesOrigin → {-0.02, -5},
 Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



- **Fig.3(c)**

Parameter values

```
Clear[x]

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 2 * x;
SELwMM = 0;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\text{SELwRM} = \frac{\text{wRM} - \text{wRR}}{\text{wRR}};$$

```
MinV = -0.02;
MaxV = 0.21;
```

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SELwM}\right) + \frac{1}{2} * (\text{SELwMM} * \text{HETEROw})$$

```
0. + x
```

Simulation1 (N = 10)

```
Site = 10;

SampleMax = 10 000;
ChangeMax = 4;
OutputData10 = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult10 = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
VectpM = Table[0, {SampleMax}];
Resultparameter10 = Table[0, {ChangeMax + 1}];
Result10 = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.01];
  If[change == 2, x = 0];
  If[change == 3, x = 0.05];
  If[change == 4, x = 0.1];
  If[change == 5, x = 0.2];

  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,

   EqρH = Sqrt[wRR] / (Sqrt[(fR*wR)/2] + Sqrt[wRR]);

   xR = Round[Site * EqρH];
   xM = 0;
   xRR = Site - xR;
   xRM = 0;
   xMM = 0;
   sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];
   xR = xR - sampleMutation[[1]];
   xM = xM + sampleMutation[[1]];
   xRR = xRR - sampleMutation[[2]];
   xRM = xRM + sampleMutation[[2]];
   xMM = 0;

   If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
   If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
   If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

   step = 2;
   While[0 < pM && pM < 1,
    cR = wRR * xRR + wRM/2 * xRM;
    cM = wMM * xMM + wRM/2 * xRM;
    If[xR == 0 && xM == 0, cRR = 0];
```

```
If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (wR^2 * xR^2)/(wR * xR + wM * xM)];
If[xR == 0 && xM == 0, cRM = 0];
If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (wR * wM * xR * xM)/(wR * xR + wM * xM)];
If[xR == 0 && xM == 0, cMM = 0];
If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (wM^2 * xM^2)/(wR * xR + wM * xM)];
pCellR = cR/(cR + cM + cRR + cRM + cMM);
pCellM = cM/(cR + cM + cRR + cRM + cMM);
pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
sampleBirth = RandomVariate[MultinomialDistribution[Site,
    {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

xR = sampleBirth[[1]];
xM = sampleBirth[[2]];
xRR = sampleBirth[[3]];
xRM = sampleBirth[[4]];
xMM = sampleBirth[[5]];
If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

step = step + 1;
];

VectpM[[sample]] = pM;

OutputData10[[countData, 1]] = x;
OutputData10[[countData, 2]] = pM;
countData = countData + 1;

];

Resultparameter10[[change]] = x;
Result10[[change]] = Total[VectpM];

OutputResult10[[countResult, 1]] = x;
OutputResult10[[countResult, 2]] = Total[VectpM];
countResult = countResult + 1;

Print[change]

];
```

1

2

3

4

5

Simulation1 (N = 100)

```
Site = 100;

SampleMax = 10 000;
ChangeMax = 4;
OutputData100 = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult100 = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
VectpM = Table[0, {SampleMax}];
Resultparameter100 = Table[0, {ChangeMax + 1}];
Result100 = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.01];
  If[change == 2, x = 0];
  If[change == 3, x = 0.05];
  If[change == 4, x = 0.1];
  If[change == 5, x = 0.2];

  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,
```

$$\text{EqρH} = \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

```
   xR = Round[Site * EqρH];
   xM = 0;
   xRR = Site - xR;
   xRM = 0;
   xMM = 0;
```

$$\text{sampleMutation} = \text{RandomVariate}\left[\text{MultinomialDistribution}\left[1, \left\{\frac{\text{xR}}{\text{xR} + 2*\text{xRR}}, \frac{2*\text{xRR}}{\text{xR} + 2*\text{xRR}}\right\}\right]\right];$$

```
   xR = xR - sampleMutation[[1]];
   xM = xM + sampleMutation[[1]];
   xRR = xRR - sampleMutation[[2]];
   xRM = xRM + sampleMutation[[2]];
   xMM = 0;
```

$$\text{If}\left[\text{xRR} + \text{xRM} + \text{xMM} == 0, \text{pM} = \frac{\text{xM}}{\text{xR} + \text{xM}}\right];$$

$$\text{If}\left[\text{xR} + \text{xM} == 0, \text{pM} = \frac{\frac{\text{xRM}}{2} + \text{xMM}}{\text{xRR} + \text{xRM} + \text{xMM}}\right];$$

$$\text{If}\left[\text{xR} + \text{xM} \neq 0 \ \&\& \ \text{xRR} + \text{xRM} + \text{xMM} \neq 0, \text{pM} = \frac{1}{2} * \frac{\frac{\text{xRM}}{2} + \text{xMM}}{\text{xRR} + \text{xRM} + \text{xMM}} + \frac{1}{2} * \frac{\text{xM}}{\text{xR} + \text{xM}}\right];$$

```
   step = 2;
   While[0 < pM && pM < 1,
```

```
cR = wRR * xRR + wRM/2 * xRM;
cM = wMM * xMM + wRM/2 * xRM;
If[xR == 0 && xM == 0, cRR = 0];
If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (wR^2 * xR^2)/(wR * xR + wM * xM)];
If[xR == 0 && xM == 0, cRM = 0];
If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (wR * wM * xR * xM)/(wR * xR + wM * xM)];
If[xR == 0 && xM == 0, cMM = 0];
If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (wM^2 * xM^2)/(wR * xR + wM * xM)];
pCellR = cR/(cR + cM + cRR + cRM + cMM);
pCellM = cM/(cR + cM + cRR + cRM + cMM);
pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
sampleBirth = RandomVariate[MultinomialDistribution[Site,
    {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

xR = sampleBirth[[1]];
xM = sampleBirth[[2]];
xRR = sampleBirth[[3]];
xRM = sampleBirth[[4]];
xMM = sampleBirth[[5]];
If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

step = step + 1;
];

VectpM[[sample]] = pM;

OutputData100[[countData, 1]] = x;
OutputData100[[countData, 2]] = pM;
countData = countData + 1;

];

Resultparameter100[[change]] = x;
Result100[[change]] = Total[VectpM];

OutputResult100[[countResult, 1]] = x;
OutputResult100[[countResult, 2]] = Total[VectpM];
countResult = countResult + 1;
```

```
    Print[change]
   ];
```

1

2

3

4

5

Output of the data

```
Resultparameter10
Result10
Export["/Users/Bessho/Desktop/workstation/OutputDataFig3c_10.txt", OutputData10,
  "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig3c_10.txt", OutputResult10,
  "Table"];

Resultparameter100
Result100
Export["/Users/Bessho/Desktop/workstation/OutputDataFig3c_100.txt", OutputData100,
  "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig3c_100.txt",
  OutputResult100, "Table"];
```

{-0.01, 0, 0.05, 0.1, 0.2}

{703, 748, 1197, 1656, 2699}

{-0.01, 0, 0.05, 0.1, 0.2}

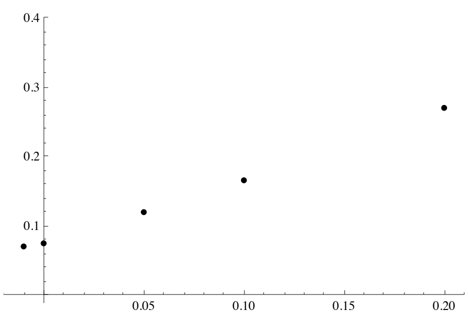
{13, 66, 879, 1615, 2834}

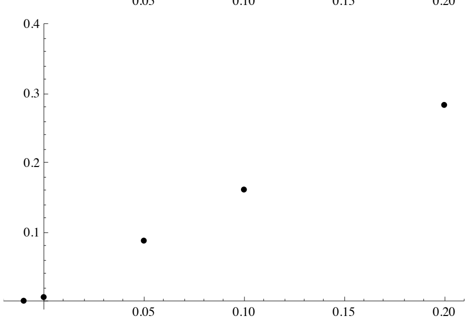
Plot

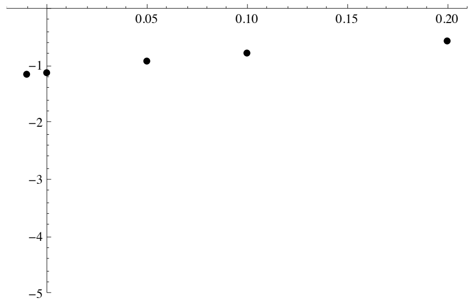
```
PlotpM10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpM100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM10[[change, 1]] = Resultparameter10[[change]];
  PlotpM10[[change, 2]] = Result10[[change]] / SampleMax;
  PlotpMlog10[[change, 1]] = Resultparameter10[[change]];
  PlotpMlog10[[change, 2]] = Log[10, Result10[[change]] / SampleMax];
  PlotpM100[[change, 1]] = Resultparameter100[[change]];
  PlotpM100[[change, 2]] = Result100[[change]] / SampleMax;
  PlotpMlog100[[change, 1]] = Resultparameter100[[change]];
  PlotpMlog100[[change, 2]] = Log[10, Result100[[change]] / SampleMax];
 ];
PlotpM10
PlotpM100
gNumerical10 = ListPlot[PlotpM10, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Black}]
gNumerical100 = ListPlot[PlotpM100, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Black}]
gLogNumerical10 = ListPlot[PlotpMlog10, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {PointSize[0.015], Black}]
gLogNumerical100 = ListPlot[PlotpMlog100, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {PointSize[0.015], Black}]
```

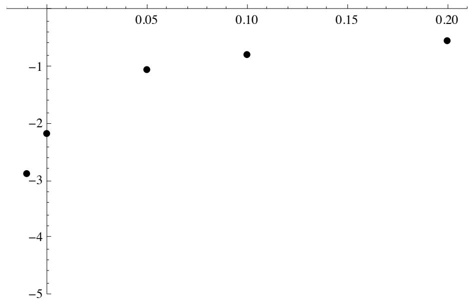
$$\left\{\left\{-0.01, \frac{703}{10000}\right\}, \left\{0, \frac{187}{2500}\right\}, \left\{0.05, \frac{1197}{10000}\right\}, \left\{0.1, \frac{207}{1250}\right\}, \left\{0.2, \frac{2699}{10000}\right\}\right\}$$

$$\left\{\left\{-0.01, \frac{13}{10000}\right\}, \left\{0, \frac{33}{5000}\right\}, \left\{0.05, \frac{879}{10000}\right\}, \left\{0.1, \frac{323}{2000}\right\}, \left\{0.2, \frac{1417}{5000}\right\}\right\}$$









Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u}=\frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \overline{z}] \sim 0.95$.

$$z = 1.96;$$

$$\underline{z10} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /.\ n \to \text{SampleMax} /.\ \hat{u} \to \frac{\text{Result10}}{\text{SampleMax}}$$

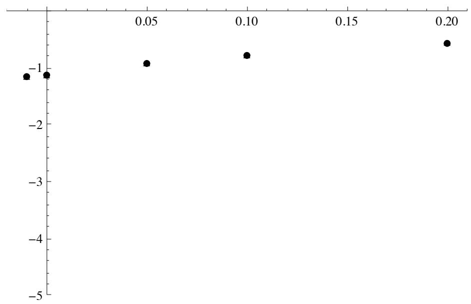
$$\overline{z10} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /.\ n \to \text{SampleMax} /.\ \hat{u} \to \frac{\text{Result10}}{\text{SampleMax}}$$

$$\underline{z100} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /.\ n \to \text{SampleMax} /.\ \hat{u} \to \frac{\text{Result100}}{\text{SampleMax}}$$

$$\overline{z100} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /.\ n \to \text{SampleMax} /.\ \hat{u} \to \frac{\text{Result100}}{\text{SampleMax}}$$

```
{0.0654525, 0.0698055, 0.113483, 0.158443, 0.261289}
{0.0754775, 0.080121, 0.126209, 0.173014, 0.278688}
{0.000759905, 0.00519146, 0.0825073, 0.154418, 0.274652}
{0.00222311, 0.00838749, 0.0936092, 0.168842, 0.292315}
```

```
PlotCIunder10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder10[[change, 1]] = Resultparameter10[[change]];
  PlotCIupper10[[change, 1]] = Resultparameter10[[change]];
  PlotCIunder10[[change, 2]] = Log[10, z10[[change]]];
  PlotCIupper10[[change, 2]] = Log[10, z10[[change]]];
 ];
Log[10, z10]
N[Log[10, PlotpM10[[All, 2]]]]
Log[10, z10]
gCIupper10 = ListPlot[PlotCIupper10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder10 = ListPlot[PlotCIunder10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];

PlotCI1♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI2♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI3♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI4♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI5♦10 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1♦10[[1, 1]] = Resultparameter10[[1]];
  PlotCI1♦10[[1, 2]] = Log[10, z10[[1]]];
  PlotCI1♦10[[2, 1]] = Resultparameter10[[1]];
  PlotCI1♦10[[2, 2]] = Log[10, z10[[1]]];
  PlotCI2♦10[[1, 1]] = Resultparameter10[[2]];
  PlotCI2♦10[[1, 2]] = Log[10, z10[[2]]];
  PlotCI2♦10[[2, 1]] = Resultparameter10[[2]];
  PlotCI2♦10[[2, 2]] = Log[10, z10[[2]]];
  PlotCI3♦10[[1, 1]] = Resultparameter10[[3]];
  PlotCI3♦10[[1, 2]] = Log[10, z10[[3]]];
  PlotCI3♦10[[2, 1]] = Resultparameter10[[3]];
  PlotCI3♦10[[2, 2]] = Log[10, z10[[3]]];
  PlotCI4♦10[[1, 1]] = Resultparameter10[[4]];
  PlotCI4♦10[[1, 2]] = Log[10, z10[[4]]];
  PlotCI4♦10[[2, 1]] = Resultparameter10[[4]];
  PlotCI4♦10[[2, 2]] = Log[10, z10[[4]]];
  PlotCI5♦10[[1, 1]] = Resultparameter10[[5]];
  PlotCI5♦10[[1, 2]] = Log[10, z10[[5]]];
  PlotCI5♦10[[2, 1]] = Resultparameter10[[5]];
  PlotCI5♦10[[2, 2]] = Log[10, z10[[5]]];
 ];
gCI1♦10 = ListPlot[PlotCI1♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2♦10 = ListPlot[PlotCI2♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3♦10 = ListPlot[PlotCI3♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4♦10 = ListPlot[PlotCI4♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5♦10 = ListPlot[PlotCI5♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical10, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10]
{-1.12218, -1.09625, -0.89891, -0.761919, -0.554882}
{-1.15304, -1.1261, -0.921906, -0.78094, -0.568797}
{-1.18407, -1.15611, -0.945068, -0.800127, -0.582879}
```



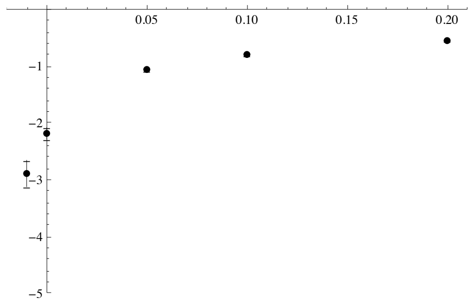
0.05
0.10
0.15
0.20
−1
−2
−3
−4
−5

```
PlotCIunder100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder100[[change, 1]] = Resultparameter100[[change]];
  PlotCIupper100[[change, 1]] = Resultparameter100[[change]];
  PlotCIunder100[[change, 2]] = Log[10, z100[[change]]];
  PlotCIupper100[[change, 2]] = Log[10, z100[[change]]];
 ];
Log[10, z100]
N[Log[10, PlotpM100[[All, 2]]]]
Log[10, z100]
gCIupper100 = ListPlot[PlotCIupper100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder100 = ListPlot[PlotCIunder100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];


PlotCI1♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI2♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI3♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI4♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI5♦100 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1♦100[[1, 1]] = Resultparameter100[[1]];
  PlotCI1♦100[[1, 2]] = Log[10, z100[[1]]];
  PlotCI1♦100[[2, 1]] = Resultparameter100[[1]];
  PlotCI1♦100[[2, 2]] = Log[10, z100[[1]]];
  PlotCI2♦100[[1, 1]] = Resultparameter100[[2]];
  PlotCI2♦100[[1, 2]] = Log[10, z100[[2]]];
  PlotCI2♦100[[2, 1]] = Resultparameter100[[2]];
  PlotCI2♦100[[2, 2]] = Log[10, z100[[2]]];
  PlotCI3♦100[[1, 1]] = Resultparameter100[[3]];
  PlotCI3♦100[[1, 2]] = Log[10, z100[[3]]];
  PlotCI3♦100[[2, 1]] = Resultparameter100[[3]];
  PlotCI3♦100[[2, 2]] = Log[10, z100[[3]]];
  PlotCI4♦100[[1, 1]] = Resultparameter100[[4]];
  PlotCI4♦100[[1, 2]] = Log[10, z100[[4]]];
  PlotCI4♦100[[2, 1]] = Resultparameter100[[4]];
  PlotCI4♦100[[2, 2]] = Log[10, z100[[4]]];
  PlotCI5♦100[[1, 1]] = Resultparameter100[[5]];
  PlotCI5♦100[[1, 2]] = Log[10, z100[[5]]];
  PlotCI5♦100[[2, 1]] = Resultparameter100[[5]];
  PlotCI5♦100[[2, 2]] = Log[10, z100[[5]]];
 ];
gCI1♦100 = ListPlot[PlotCI1♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2♦100 = ListPlot[PlotCI2♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3♦100 = ListPlot[PlotCI3♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4♦100 = ListPlot[PlotCI4♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5♦100 = ListPlot[PlotCI5♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical100, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100]
```

{-2.65304, -2.07637, -1.02868, -0.772518, -0.53415}

{-2.88606, -2.18046, -1.05601, -0.791827, -0.5476}

{-3.11924, -2.28471, -1.08351, -0.811303, -0.561218}



Fixation probability from branching process approximation

```
Clear[y]

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 2 * y;
SELwMM = 0.0;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\text{SELwRM} = \frac{\text{wRM} - \text{wRR}}{\text{wRR}};$$

$$s_{average} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SELwM}\right) + \frac{1}{2} * (\text{SELwMM} * \text{HETEROw})$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR}*\text{wR}}{2}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{avef} = \frac{\text{fM} + \text{fR}}{2};$$

$$\text{W} = \frac{\text{fR} * \text{wR}}{2} * \text{Eq}\rho\text{H} + \text{wRR} * \text{Eq}\rho\text{D};$$

$$\text{EqD4a} = 1 - \text{PH} - \text{Exp}\left[-\text{PD} * \frac{\text{avef} * \text{wM}}{\text{W}}\right];$$

$$\text{EqD4b} = 1 - \text{PD} - \text{Exp}\left[-\text{PH} * \frac{\text{wRM}}{2\,\text{W}}\right];$$

```
RepeatMax = 200;
ResultH = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
ResultD = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
ResultBPfix = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
LogResultBPfix = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
repeat = 0;
For[repeat1 = 1, repeat1 ≤ RepeatMax + 1, repeat1 = repeat1 + 1,
  repeat = repeat + 1;
```

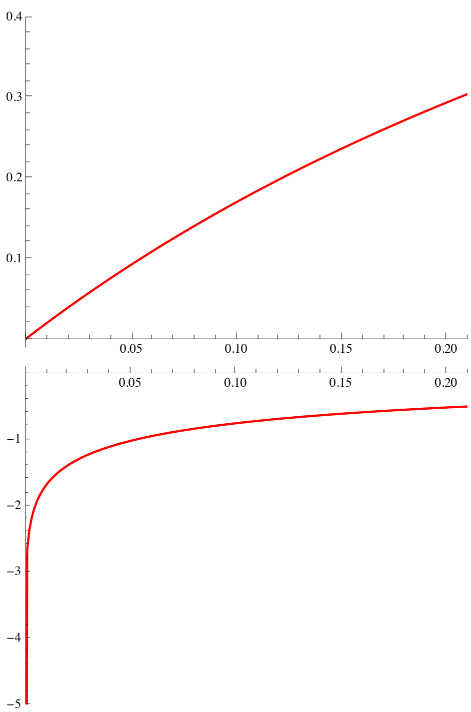
```
x1 = 0 + (MaxV - 0)/RepeatMax * (repeat1 - 1);
CondH = EqD4a /. y → x1;
CondD = EqD4b /. y → x1;
sol = Solve[CondH == 0 && CondD == 0 && 0 < PH && PH < 1 && 0 < PD && PD < 1, {PH, PD}, Reals];
BPPaiH = Part[(PH /. sol), 1];
BPPaiD = Part[(PD /. sol), 1];
ResultH[[repeat, 1]] = x1;
ResultD[[repeat, 1]] = x1;
ResultBPfix[[repeat, 1]] = x1;
LogResultBPfix[[repeat, 1]] = x1;
ResultH[[repeat, 2]] = BPPaiH;
ResultD[[repeat, 2]] = BPPaiD;
ResultBPfix[[repeat, 2]] = EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD;
LogResultBPfix[[repeat, 2]] = Log[10, EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD];
];
gBPfix = ListPlot[ResultBPfix, Joined → True, PlotRange → {{0, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gLogBPfix = ListPlot[LogResultBPfix, Joined → True, PlotRange → {{0, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
0. + y
```

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

General::stop : この計算中に，Solve::ratnzのこれ以上の出力は表示されません． ≫



Fixation probability from diffusion approximation

```
Site = 10;
```

$$\text{Parameters} = \left\{\text{SUM} \to \text{Site},\ \hat{\rho}_H \to \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR*wR}}{2}} + \sqrt{\text{wRR}}},\ \hat{\rho}_D \to \frac{\sqrt{\frac{\text{fR*wR}}{2}}}{\sqrt{\frac{\text{fR*wR}}{2}} + \sqrt{\text{wRR}}},\ \text{sfM} \to \text{SELfM},\ \text{swM} \to \text{SELwM},\ \text{swRM} \to \text{SELwRM},\ \text{swMM} \to \text{SELwMM}\right\}$$

```
Additive = swMM - 2 * swRM;
```

$$m = \text{SUM} * p * (1 - p) * \frac{2 * s_{average} + p * \text{Additive}}{2};$$

$$v = \frac{p * (1 - p) * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)}{8 * \hat{\rho}_H * \hat{\rho}_D};$$

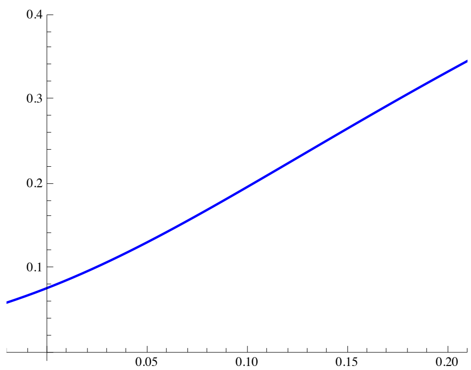
$$Q = \text{Integrate}\left[\frac{m}{v} \text{ /. Parameters}, p\right];$$

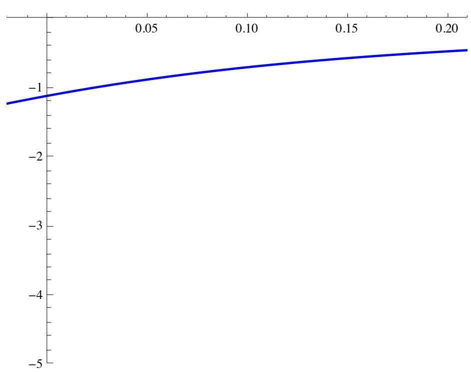
$$\text{INI} = \frac{1}{\text{SUM} * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)} \text{ /. Parameters};$$

$$\text{DFfixWS} = \frac{\text{Integrate}[\text{Exp}[-2 * Q], \{p, 0, \text{INI}\}]}{\text{Integrate}[\text{Exp}[-2 * Q], \{p, 0, 1\}]};$$

```
gDFfixWS10 = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS10 = Plot[Log[10, DFfixWS], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

$\{\text{SUM} \to 10,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,\ \text{sfM} \to 0,\ \text{swM} \to 2\,y,\ \text{swRM} \to 0.,\ \text{swMM} \to 0.\}$





Fixation probability from diffusion approximation

```
Site = 100;
```

$$\texttt{Parameters} = \left\{\texttt{SUM} \to \texttt{Site},\ \hat{\rho}_H \to \frac{\sqrt{\texttt{wRR}}}{\sqrt{\frac{\texttt{fR*wR}}{2}} + \sqrt{\texttt{wRR}}},\ \hat{\rho}_D \to \frac{\sqrt{\frac{\texttt{fR*wR}}{2}}}{\sqrt{\frac{\texttt{fR*wR}}{2}} + \sqrt{\texttt{wRR}}},\ \texttt{sfM} \to \texttt{SELfM},\ \texttt{swM} \to \texttt{SELwM},\ \texttt{swRM} \to \texttt{SELwRM},\ \texttt{swMM} \to \texttt{SELwMM}\right\}$$

```
Additive = swMM - 2 * swRM;
```

$$\texttt{m} = \texttt{SUM} * \texttt{p} * (1 - \texttt{p}) * \frac{2 * s_{average} + \texttt{p} * \texttt{Additive}}{2};$$

$$\texttt{v} = \frac{\texttt{p} * (1 - \texttt{p}) * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)}{8 * \hat{\rho}_H * \hat{\rho}_D};$$

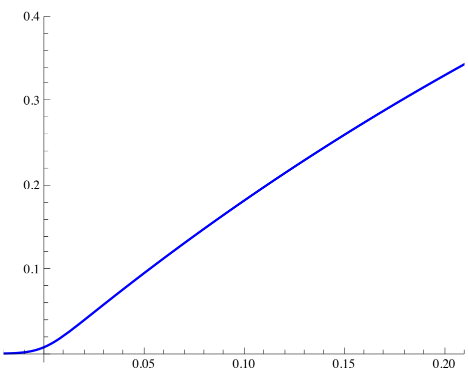
$$\texttt{Q} = \texttt{Integrate}\left[\frac{\texttt{m}}{\texttt{v}} \ /. \ \texttt{Parameters},\ \texttt{p}\right];$$

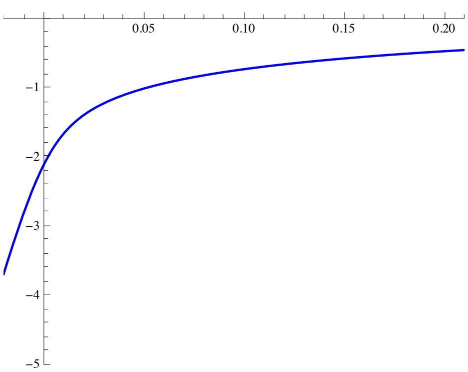
$$\texttt{INI} = \frac{1}{\texttt{SUM} * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)} \ /. \ \texttt{Parameters};$$

$$\texttt{DFfixWS} = \frac{\texttt{Integrate[Exp[-2 * Q], \{p, 0, INI\}]}}{\texttt{Integrate[Exp[-2 * Q], \{p, 0, 1\}]}};$$

```
gDFfixWS100 = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS100 = Plot[Log[10, DFfixWS], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

$\{\mathrm{SUM} \to 100,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,\ \mathrm{sfM} \to 0,\ \mathrm{swM} \to 2\,y,\ \mathrm{swRM} \to 0.,\ \mathrm{swMM} \to 0.\}$

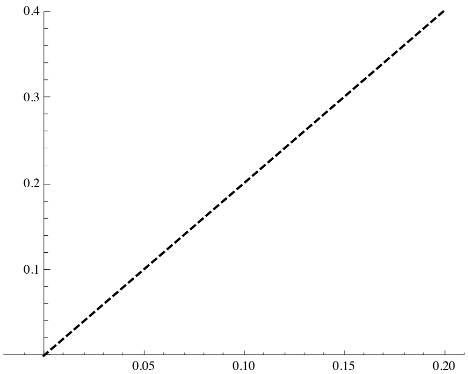


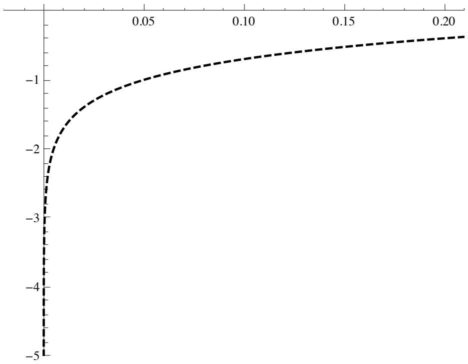


Analytical result (Lineary approximation)

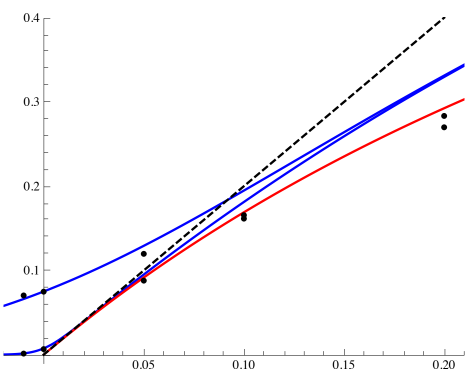
```
FixLinear = (8 * ρ̂H * ρ̂D)/(ρ̂H + 2 * ρ̂D)^2 * 2 * saverage;
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.4}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
gLogFixLinear = Plot[Log[10, FixLinear /. Parameters], {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-5, 0}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```

$$FixLinear = \frac{8 * \hat{\rho}_H * \hat{\rho}_D}{\left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)^2} * 2 * s_{average};$$






```
Show[gDFfixWS10, gDFfixWS100, gBPfix, gFixLinear, gNumerical10, gNumerical100]
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100]
```



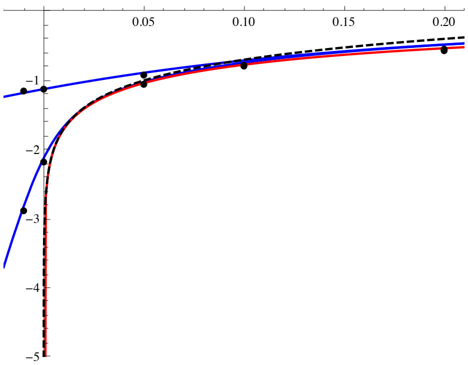
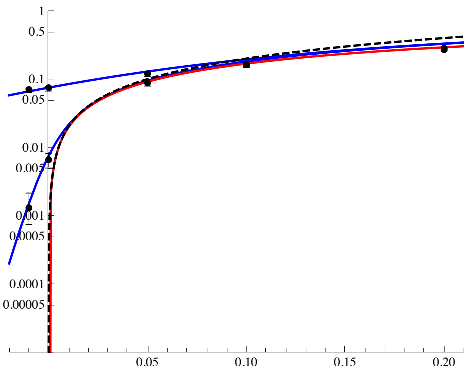


Figure for output

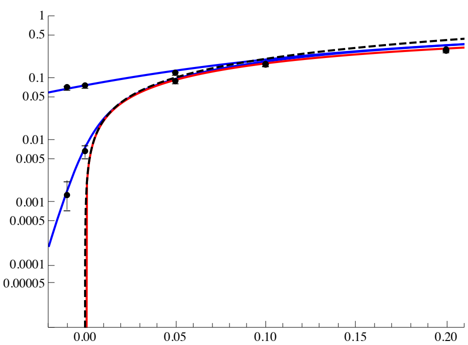
```
axeslabels = {{-5, "0.00001"}, {Log[10, 0.00005], "0.00005"}, {-4, "0.0001"},
   {Log[10, 0.0005], "0.0005"}, {-3, "0.001"}, {Log[10, 0.005], "0.005"},
   {-2, "0.01"}, {Log[10, 0.05], "0.05"}, {-1, "0.1"}, {Log[10, 0.5], "0.5"}, {0, "1"}};
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100, PlotRange → {{MinV, MaxV}, {-5, 0}}, AxesOrigin → {0, -5},
 Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```

$$\frac{\sqrt{\mathrm{wRR}}}{\sqrt{\frac{\mathrm{fR}*\mathrm{wR}}{2}}+\sqrt{\mathrm{wRR}}}$$



0.666667

```
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100, PlotRange → {{MinV, MaxV}, {-5, 0}}, AxesOrigin → {-0.02, -5},
 Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



$N\left[\frac{\mathrm{Result10}}{\mathrm{SampleMax}}\right]$

$N\left[\frac{\mathrm{Result100}}{\mathrm{SampleMax}}\right]$

```
{0.0703, 0.0748, 0.1197, 0.1656, 0.2699}
{0.0013, 0.0066, 0.0879, 0.1615, 0.2834}
```

- **Fig.3(d)**

Parameter values

```
Clear[x]

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 0;
SELwMM = 4 * x;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\text{SELwRM} = \frac{\text{wRM} - \text{wRR}}{\text{wRR}};$$

```
MinV = -0.02;
MaxV = 0.21;
```

$$s_{average} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SELwM}\right) + \frac{1}{2} * (\text{SELwMM} * \text{HETEROw})$$

```
1. x
```

Simulation1 (N = 10)

```
Site = 10;

SampleMax = 10 000;
ChangeMax = 4;
OutputData10 = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult10 = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
VectpM = Table[0, {SampleMax}];
Resultparameter10 = Table[0, {ChangeMax + 1}];
Result10 = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.01];
  If[change == 2, x = 0];
  If[change == 3, x = 0.05];
  If[change == 4, x = 0.1];
  If[change == 5, x = 0.2];

  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,
```

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR} * \text{wR}}{2}} + \sqrt{\text{wRR}}};$$

```
   xR = Round[Site * EqρH];
   xM = 0;
   xRR = Site - xR;
   xRM = 0;
   xMM = 0;
```

```
sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), 2 * xRR/(xR + 2 * xRR)}]];
xR = xR - sampleMutation[[1]];
xM = xM + sampleMutation[[1]];
xRR = xRR - sampleMutation[[2]];
xRM = xRM + sampleMutation[[2]];
xMM = 0;

If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

step = 2;
While[0 < pM && pM < 1,
 cR = wRR * xRR + wRM/2 * xRM;
 cM = wMM * xMM + wRM/2 * xRM;
 If[xR == 0 && xM == 0, cRR = 0];
 If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (wR^2 * xR^2)/(wR * xR + wM * xM)];
 If[xR == 0 && xM == 0, cRM = 0];
 If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (wR * wM * xR * xM)/(wR * xR + wM * xM)];
 If[xR == 0 && xM == 0, cMM = 0];
 If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (wM^2 * xM^2)/(wR * xR + wM * xM)];
 pCellR = cR/(cR + cM + cRR + cRM + cMM);
 pCellM = cM/(cR + cM + cRR + cRM + cMM);
 pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
 pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
 pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
 sampleBirth = RandomVariate[MultinomialDistribution[Site,
    {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

 xR = sampleBirth[[1]];
 xM = sampleBirth[[2]];
 xRR = sampleBirth[[3]];
 xRM = sampleBirth[[4]];
 xMM = sampleBirth[[5]];
 If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
 If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
 If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];
```

```
      step = step + 1;
     ];

    VectpM[[sample]] = pM;

    OutputData10[[countData, 1]] = x;
    OutputData10[[countData, 2]] = pM;
    countData = countData + 1;

   ];

   Resultparameter10[[change]] = x;
   Result10[[change]] = Total[VectpM];

   OutputResult10[[countResult, 1]] = x;
   OutputResult10[[countResult, 2]] = Total[VectpM];
   countResult = countResult + 1;

   Print[change]

  ];
```

1

2

3

4

5

Simulation1 (N = 100)

```
Site = 100;

SampleMax = 10 000;
ChangeMax = 4;
OutputData100 = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult100 = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
VectpM = Table[0, {SampleMax}];
Resultparameter100 = Table[0, {ChangeMax + 1}];
Result100 = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.01];
  If[change == 2, x = 0];
  If[change == 3, x = 0.05];
  If[change == 4, x = 0.1];
  If[change == 5, x = 0.2];

  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,
```

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

```
   xR = Round[Site * EqρH];
```

```
xM = 0;
xRR = Site - xR;
xRM = 0;
xMM = 0;
sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];
xR = xR - sampleMutation[[1]];
xM = xM + sampleMutation[[1]];
xRR = xRR - sampleMutation[[2]];
xRM = xRM + sampleMutation[[2]];
xMM = 0;

If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

step = 2;
While[0 < pM && pM < 1,
 cR = wRR * xRR + wRM/2 * xRM;
 cM = wMM * xMM + wRM/2 * xRM;
 If[xR == 0 && xM == 0, cRR = 0];
 If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (wR^2 * xR^2)/(wR * xR + wM * xM)];
 If[xR == 0 && xM == 0, cRM = 0];
 If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (wR * wM * xR * xM)/(wR * xR + wM * xM)];
 If[xR == 0 && xM == 0, cMM = 0];
 If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (wM^2 * xM^2)/(wR * xR + wM * xM)];
 pCellR = cR/(cR + cM + cRR + cRM + cMM);
 pCellM = cM/(cR + cM + cRR + cRM + cMM);
 pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
 pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
 pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
 sampleBirth = RandomVariate[MultinomialDistribution[Site,
    {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

 xR = sampleBirth[[1]];
 xM = sampleBirth[[2]];
 xRR = sampleBirth[[3]];
 xRM = sampleBirth[[4]];
 xMM = sampleBirth[[5]];
 If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
```

```
      If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];

      If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

      step = step + 1;
     ];

    VectpM[[sample]] = pM;

    OutputData100[[countData, 1]] = x;
    OutputData100[[countData, 2]] = pM;
    countData = countData + 1;

   ];

   Resultparameter100[[change]] = x;
   Result100[[change]] = Total[VectpM];

   OutputResult100[[countResult, 1]] = x;
   OutputResult100[[countResult, 2]] = Total[VectpM];
   countResult = countResult + 1;

   Print[change]

  ];
```

1

2

3

4

5

Output of the data

```
Resultparameter10
Result10
Export["/Users/Bessho/Desktop/workstation/OutputDataFig3d_10.txt", OutputData10,
  "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig3d_10.txt", OutputResult10,
  "Table"];

Resultparameter100
Result100
Export["/Users/Bessho/Desktop/workstation/OutputDataFig3d_100.txt", OutputData100,
  "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig3d_100.txt",
  OutputResult100, "Table"];
```

{-0.01, 0, 0.05, 0.1, 0.2}

{697, 740, 1364, 1765, 2600}

{-0.01, 0, 0.05, 0.1, 0.2}

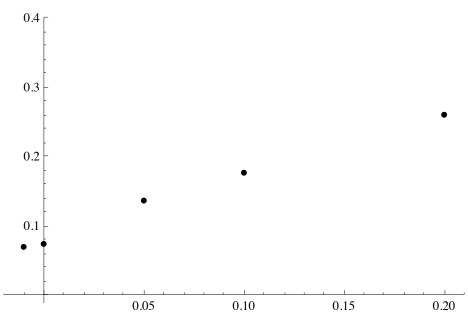
{12, 76, 900, 1699, 2901}

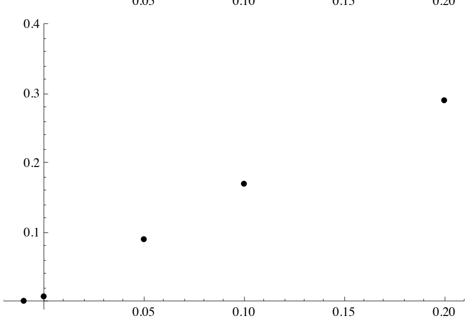
Plot

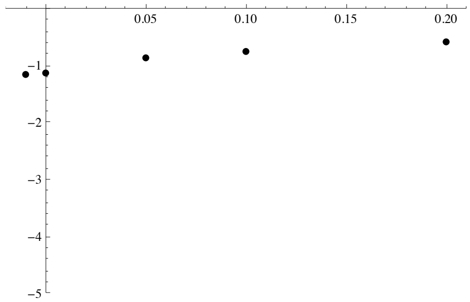
```
PlotpM10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpM100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM10[[change, 1]] = Resultparameter10[[change]];
  PlotpM10[[change, 2]] = Result10[[change]] / SampleMax;
  PlotpMlog10[[change, 1]] = Resultparameter10[[change]];
  PlotpMlog10[[change, 2]] = Log[10, Result10[[change]] / SampleMax];
  PlotpM100[[change, 1]] = Resultparameter100[[change]];
  PlotpM100[[change, 2]] = Result100[[change]] / SampleMax;
  PlotpMlog100[[change, 1]] = Resultparameter100[[change]];
  PlotpMlog100[[change, 2]] = Log[10, Result100[[change]] / SampleMax];
 ];
PlotpM10
PlotpM100
gNumerical10 = ListPlot[PlotpM10, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Black}]
gNumerical100 = ListPlot[PlotpM100, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Black}]
gLogNumerical10 = ListPlot[PlotpMlog10, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {PointSize[0.015], Black}]
gLogNumerical100 = ListPlot[PlotpMlog100, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {PointSize[0.015], Black}]
```

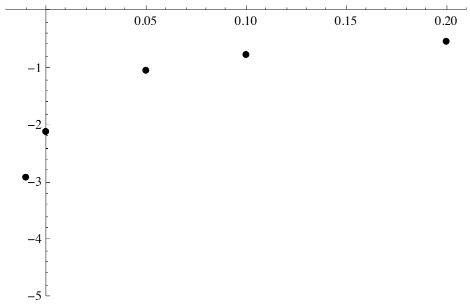
$\left\{\left\{-0.01, \frac{697}{10\,000}\right\}, \left\{0, \frac{37}{500}\right\}, \left\{0.05, \frac{341}{2500}\right\}, \left\{0.1, \frac{353}{2000}\right\}, \left\{0.2, \frac{13}{50}\right\}\right\}$

$\left\{\left\{-0.01, \frac{3}{2500}\right\}, \left\{0, \frac{19}{2500}\right\}, \left\{0.05, \frac{9}{100}\right\}, \left\{0.1, \frac{1699}{10\,000}\right\}, \left\{0.2, \frac{2901}{10\,000}\right\}\right\}$









Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, Pr[$\underline{z}$ < u < $\overline{z}$] ~ 0.95.

$$z = 1.96;$$

$$\underline{z10} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to \text{SampleMax /. } \hat{u} \to \frac{\text{Result10}}{\text{SampleMax}}$$

$$\overline{z10} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to \text{SampleMax /. } \hat{u} \to \frac{\text{Result10}}{\text{SampleMax}}$$

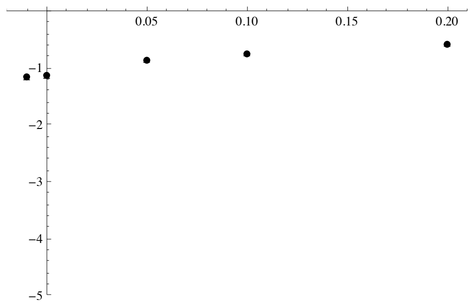
$$\underline{z100} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to \text{SampleMax /. } \hat{u} \to \frac{\text{Result100}}{\text{SampleMax}}$$

$$\overline{z100} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to \text{SampleMax /. } \hat{u} \to \frac{\text{Result100}}{\text{SampleMax}}$$

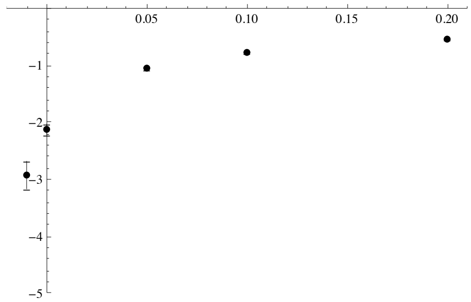
```
{0.0648725, 0.0690313, 0.129812, 0.169152, 0.251496}
{0.074858, 0.0792959, 0.143267, 0.184096, 0.268688}
{0.000686597, 0.00607676, 0.0845472, 0.162666, 0.281287}
{0.00209649, 0.00950142, 0.0957677, 0.177387, 0.299074}
```

```
PlotCIunder10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper10 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder10[[change, 1]] = Resultparameter10[[change]];
  PlotCIupper10[[change, 1]] = Resultparameter10[[change]];
  PlotCIunder10[[change, 2]] = Log[10, z̲1̲0̲[[change]]];
  PlotCIupper10[[change, 2]] = Log[10, z̅1̅0̅[[change]]];
 ];
Log[10, z̅1̅0̅]
N[Log[10, PlotpM10[[All, 2]]]]
Log[10, z̲1̲0̲]
gCIupper10 = ListPlot[PlotCIupper10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder10 = ListPlot[PlotCIunder10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];

PlotCI1♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI2♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI3♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI4♦10 = Table[0, {i, 2}, {j, 2}];
PlotCI5♦10 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1♦10[[1, 1]] = Resultparameter10[[1]];
  PlotCI1♦10[[1, 2]] = Log[10, z̲1̲0̲[[1]]];
  PlotCI1♦10[[2, 1]] = Resultparameter10[[1]];
  PlotCI1♦10[[2, 2]] = Log[10, z̅1̅0̅[[1]]];
  PlotCI2♦10[[1, 1]] = Resultparameter10[[2]];
  PlotCI2♦10[[1, 2]] = Log[10, z̲1̲0̲[[2]]];
  PlotCI2♦10[[2, 1]] = Resultparameter10[[2]];
  PlotCI2♦10[[2, 2]] = Log[10, z̅1̅0̅[[2]]];
  PlotCI3♦10[[1, 1]] = Resultparameter10[[3]];
  PlotCI3♦10[[1, 2]] = Log[10, z̲1̲0̲[[3]]];
  PlotCI3♦10[[2, 1]] = Resultparameter10[[3]];
  PlotCI3♦10[[2, 2]] = Log[10, z̅1̅0̅[[3]]];
  PlotCI4♦10[[1, 1]] = Resultparameter10[[4]];
  PlotCI4♦10[[1, 2]] = Log[10, z̲1̲0̲[[4]]];
  PlotCI4♦10[[2, 1]] = Resultparameter10[[4]];
  PlotCI4♦10[[2, 2]] = Log[10, z̅1̅0̅[[4]]];
  PlotCI5♦10[[1, 1]] = Resultparameter10[[5]];
  PlotCI5♦10[[1, 2]] = Log[10, z̲1̲0̲[[5]]];
  PlotCI5♦10[[2, 1]] = Resultparameter10[[5]];
  PlotCI5♦10[[2, 2]] = Log[10, z̅1̅0̅[[5]]];
 ];
gCI1♦10 = ListPlot[PlotCI1♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2♦10 = ListPlot[PlotCI2♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3♦10 = ListPlot[PlotCI3♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4♦10 = ListPlot[PlotCI4♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5♦10 = ListPlot[PlotCI5♦10, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical10, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10]
```

```
{-1.12576, -1.10075, -0.843855, -0.734955, -0.570751}
{-1.15677, -1.13077, -0.865186, -0.753255, -0.585027}
{-1.18794, -1.16095, -0.886683, -0.771722, -0.599469}
```


0.05
0.10
0.15
0.20
−1
−2
−3
−4
−5

```
PlotCIunder100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper100 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder100[[change, 1]] = Resultparameter100[[change]];
  PlotCIupper100[[change, 1]] = Resultparameter100[[change]];
  PlotCIunder100[[change, 2]] = Log[10, z100[[change]]];
  PlotCIupper100[[change, 2]] = Log[10, z100[[change]]];
 ];
Log[10, z100]
N[Log[10, PlotpM100[[All, 2]]]]
Log[10, z100]
gCIupper100 = ListPlot[PlotCIupper100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder100 = ListPlot[PlotCIunder100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];


PlotCI1♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI2♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI3♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI4♦100 = Table[0, {i, 2}, {j, 2}];
PlotCI5♦100 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1♦100[[1, 1]] = Resultparameter100[[1]];
  PlotCI1♦100[[1, 2]] = Log[10, z100[[1]]];
  PlotCI1♦100[[2, 1]] = Resultparameter100[[1]];
  PlotCI1♦100[[2, 2]] = Log[10, z100[[1]]];
  PlotCI2♦100[[1, 1]] = Resultparameter100[[2]];
  PlotCI2♦100[[1, 2]] = Log[10, z100[[2]]];
  PlotCI2♦100[[2, 1]] = Resultparameter100[[2]];
  PlotCI2♦100[[2, 2]] = Log[10, z100[[2]]];
  PlotCI3♦100[[1, 1]] = Resultparameter100[[3]];
  PlotCI3♦100[[1, 2]] = Log[10, z100[[3]]];
  PlotCI3♦100[[2, 1]] = Resultparameter100[[3]];
  PlotCI3♦100[[2, 2]] = Log[10, z100[[3]]];
  PlotCI4♦100[[1, 1]] = Resultparameter100[[4]];
  PlotCI4♦100[[1, 2]] = Log[10, z100[[4]]];
  PlotCI4♦100[[2, 1]] = Resultparameter100[[4]];
  PlotCI4♦100[[2, 2]] = Log[10, z100[[4]]];
  PlotCI5♦100[[1, 1]] = Resultparameter100[[5]];
  PlotCI5♦100[[1, 2]] = Log[10, z100[[5]]];
  PlotCI5♦100[[2, 1]] = Resultparameter100[[5]];
  PlotCI5♦100[[2, 2]] = Log[10, z100[[5]]];
 ];
gCI1♦100 = ListPlot[PlotCI1♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2♦100 = ListPlot[PlotCI2♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3♦100 = ListPlot[PlotCI3♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4♦100 = ListPlot[PlotCI4♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5♦100 = ListPlot[PlotCI5♦100, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gLogNumerical100, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100]
{-2.67851, -2.02221, -1.01878, -0.751078, -0.524221}
{-2.92082, -2.11919, -1.04576, -0.769807, -0.537452}
{-3.1633, -2.21633, -1.0729, -0.788702, -0.55085}
```



Fixation probability from branching process approximation

```
Clear[y]

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 0.0;
SELwMM = 4 * y;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\text{SELwRM} = \frac{\text{wRM} - \text{wRR}}{\text{wRR}};$$

$$s_{average} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SELwM}\right) + \frac{1}{2} * (\text{SELwMM} * \text{HETEROw})$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR}*\text{wR}}{2}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{avef} = \frac{\text{fM} + \text{fR}}{2};$$

$$\text{W} = \frac{\text{fR} * \text{wR}}{2} * \text{Eq}\rho\text{H} + \text{wRR} * \text{Eq}\rho\text{D};$$

$$\text{EqD4a} = 1 - \text{PH} - \text{Exp}\left[-\text{PD} * \frac{\text{avef} * \text{wM}}{\text{W}}\right];$$

$$\text{EqD4b} = 1 - \text{PD} - \text{Exp}\left[-\text{PH} * \frac{\text{wRM}}{2\,\text{W}}\right];$$

```
RepeatMax = 200;
ResultH = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
ResultD = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
ResultBPfix = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
LogResultBPfix = Table[0, {i, (RepeatMax + 1)}, {j, 2}];
repeat = 0;
For[repeat1 = 1, repeat1 ≤ RepeatMax + 1, repeat1 = repeat1 + 1,
  repeat = repeat + 1;
```

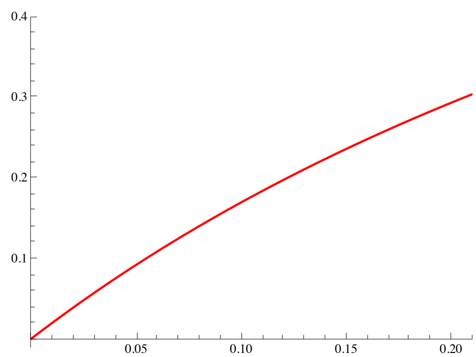
```
    x1 = 0 + (MaxV - 0)/RepeatMax * (repeat1 - 1);
    CondH = EqD4a /. y → x1;
    CondD = EqD4b /. y → x1;
    sol = Solve[CondH == 0 && CondD == 0 && 0 < PH && PH < 1 && 0 < PD && PD < 1, {PH, PD}, Reals];
    BPPaiH = Part[(PH /. sol), 1];
    BPPaiD = Part[(PD /. sol), 1];
    ResultH[[repeat, 1]] = x1;
    ResultD[[repeat, 1]] = x1;
    ResultBPfix[[repeat, 1]] = x1;
    LogResultBPfix[[repeat, 1]] = x1;
    ResultH[[repeat, 2]] = BPPaiH;
    ResultD[[repeat, 2]] = BPPaiD;
    ResultBPfix[[repeat, 2]] = EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD;
    LogResultBPfix[[repeat, 2]] = Log[10, EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD];
  ];
gBPfix = ListPlot[ResultBPfix, Joined → True, PlotRange → {{0, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gLogBPfix = ListPlot[LogResultBPfix, Joined → True, PlotRange → {{0, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
0. + 1. y
```

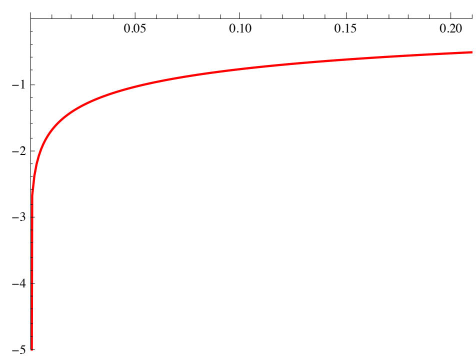
Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

Solve::ratnz : Solveは厳密でない係数の系を解くことができませんでした．解は対応する厳密系を解き，結果を数値に変換することで得られました． ≫

General::stop : この計算中に，Solve::ratnzのこれ以上の出力は表示されません． ≫





Fixation probability from diffusion approximation

```
Site = 10;
```

$$\text{Parameters} = \left\{\text{SUM} \to \text{Site},\ \hat{\rho}_H \to \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR*wR}}{2}} + \sqrt{\text{wRR}}},\ \hat{\rho}_D \to \frac{\sqrt{\frac{\text{fR*wR}}{2}}}{\sqrt{\frac{\text{fR*wR}}{2}} + \sqrt{\text{wRR}}},\ \text{sfM} \to \text{SELfM},\ \text{swM} \to \text{SELwM},\ \text{swRM} \to \text{SELwRM},\ \text{swMM} \to \text{SELwMM}\right\}$$

```
Additive = swMM - 2 * swRM;
```

$$m = \text{SUM} * p * (1 - p) * \frac{2 * s_{average} + p * \text{Additive}}{2};$$

$$v = \frac{p * (1 - p) * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)}{8 * \hat{\rho}_H * \hat{\rho}_D};$$

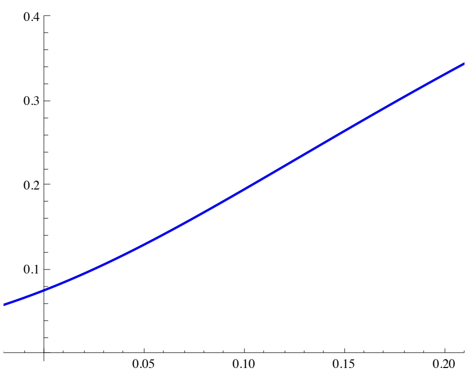
$$Q = \text{Integrate}\left[\frac{m}{v}\ /.\ \text{Parameters},\ p\right];$$

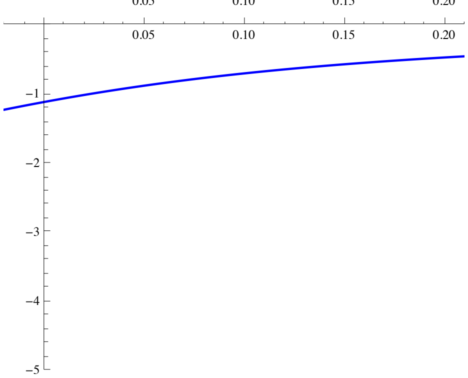
$$\text{INI} = \frac{1}{\text{SUM} * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)}\ /.\ \text{Parameters};$$

$$\text{DFfixWS} = \frac{\text{Integrate}[\text{Exp}[-2 * Q],\ \{p,\ 0,\ \text{INI}\}]}{\text{Integrate}[\text{Exp}[-2 * Q],\ \{p,\ 0,\ 1\}]};$$

```
gDFfixWS10 = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS10 = Plot[Log[10, DFfixWS], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

$$\left\{\text{SUM} \to 10,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,\ \text{sfM} \to 0,\ \text{swM} \to 0.,\ \text{swRM} \to \frac{-1000 + 1000\ (1 + 2.\ y)}{1000},\ \text{swMM} \to 4\ y\right\}$$





Fixation probability from diffusion approximation

$$\texttt{Site} = 100;$$

$$\texttt{Parameters} = \left\{\texttt{SUM} \to \texttt{Site},\ \hat{\rho}_{\text{H}} \to \frac{\sqrt{\texttt{wRR}}}{\sqrt{\frac{\texttt{fR*wR}}{2}} + \sqrt{\texttt{wRR}}},\ \hat{\rho}_{\text{D}} \to \frac{\sqrt{\frac{\texttt{fR*wR}}{2}}}{\sqrt{\frac{\texttt{fR*wR}}{2}} + \sqrt{\texttt{wRR}}},\ \texttt{sfM} \to \texttt{SELfM},\right.$$
$$\left.\texttt{swM} \to \texttt{SELwM},\ \texttt{swRM} \to \texttt{SELwRM},\ \texttt{swMM} \to \texttt{SELwMM}\right\}$$

$$\texttt{Additive} = \texttt{swMM} - 2 * \texttt{swRM};$$

$$\texttt{m} = \texttt{SUM} * \texttt{p} * (1 - \texttt{p}) * \frac{2 * \texttt{s}_{\texttt{average}} + \texttt{p} * \texttt{Additive}}{2};$$

$$\texttt{v} = \frac{\texttt{p} * (1 - \texttt{p}) * \left(\hat{\rho}_{\text{H}} + 2 * \hat{\rho}_{\text{D}}\right)}{8 * \hat{\rho}_{\text{H}} * \hat{\rho}_{\text{D}}};$$

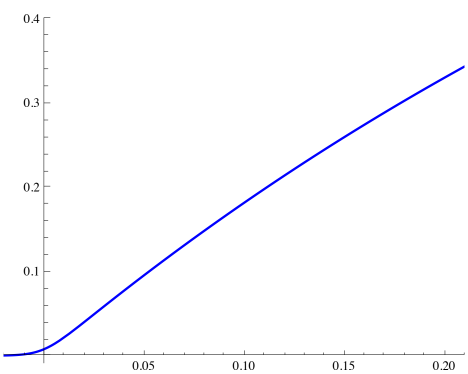
$$\texttt{Q} = \texttt{Integrate}\left[\frac{\texttt{m}}{\texttt{v}}\ \texttt{/.}\ \texttt{Parameters},\ \texttt{p}\right];$$

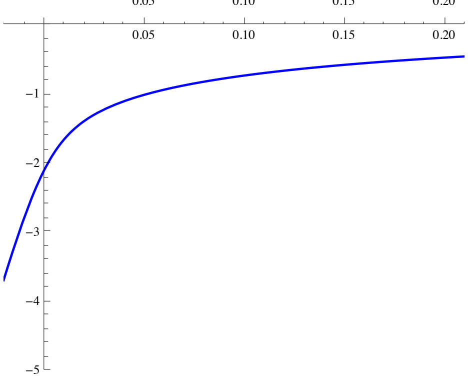
$$\texttt{INI} = \frac{1}{\texttt{SUM} * \left(\hat{\rho}_{\text{H}} + 2 * \hat{\rho}_{\text{D}}\right)}\ \texttt{/.}\ \texttt{Parameters};$$

$$\texttt{DFfixWS} = \frac{\texttt{Integrate[Exp[-2*Q], \{p, 0, INI\}]}}{\texttt{Integrate[Exp[-2*Q], \{p, 0, 1\}]}};$$

```
gDFfixWS100 = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.4}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
gLogDFfixWS100 = Plot[Log[10, DFfixWS], {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-5, 0}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

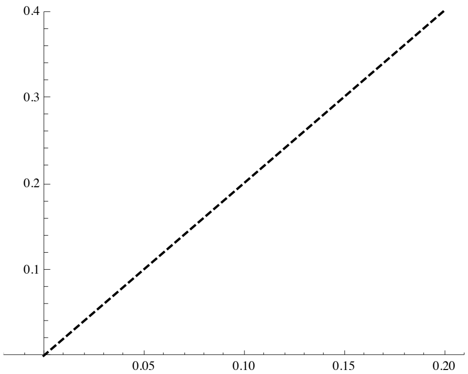
$$\left\{\text{SUM} \to 100,\ \hat{\rho}_{\text{H}} \to 0.666667,\ \hat{\rho}_{\text{D}} \to 0.333333,\ \text{sfM} \to 0,\ \text{swM} \to 0.,\ \text{swRM} \to \frac{-1000 + 1000\ (1 + 2.\ \text{y})}{1000},\ \text{swMM} \to 4\ \text{y}\right\}$$

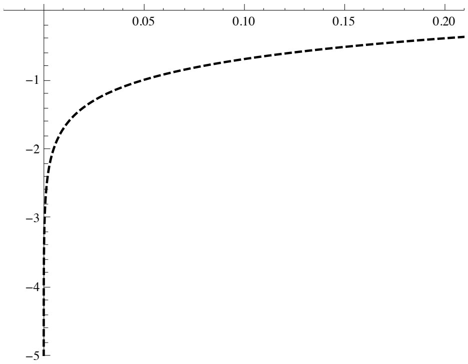




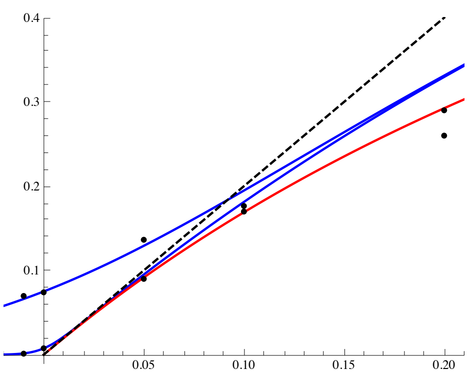
Analytical result (Lineary approximation)

```
FixLinear = (8 * ρ̂_H * ρ̂_D)/(ρ̂_H + 2 * ρ̂_D)^2 * 2 * s_average;
gFixLinear = Plot[FixLinear /. Parameters, {γ, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.4}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
gLogFixLinear = Plot[Log[10, FixLinear /. Parameters], {γ, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {-5, 0}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```






```
Show[gDFfixWS10, gDFfixWS100, gBPfix, gFixLinear, gNumerical10, gNumerical100]
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100]
```



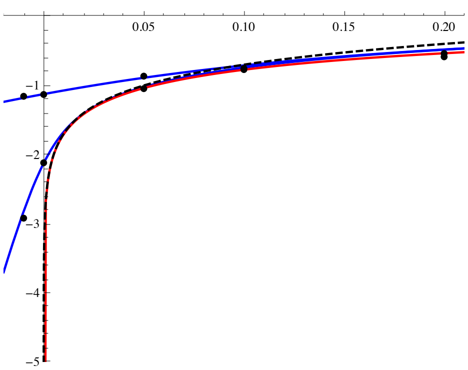
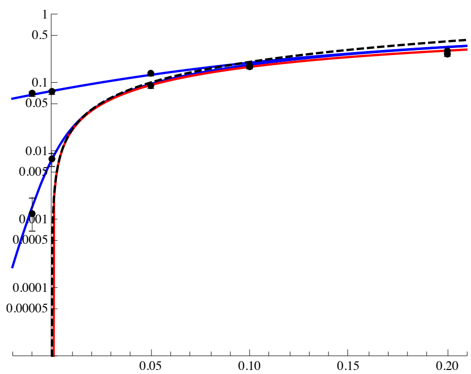


Figure for output

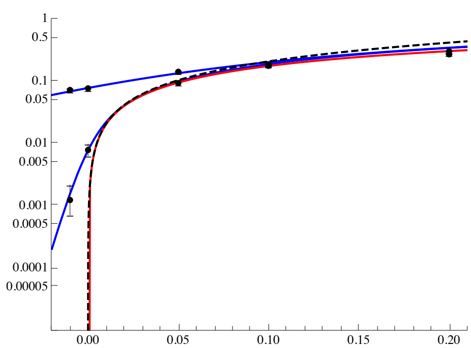
```
axeslabels = {{-5, "0.00001"}, {Log[10, 0.00005], "0.00005"}, {-4, "0.0001"},
   {Log[10, 0.0005], "0.0005"}, {-3, "0.001"}, {Log[10, 0.005], "0.005"},
   {-2, "0.01"}, {Log[10, 0.05], "0.05"}, {-1, "0.1"}, {Log[10, 0.5], "0.5"}, {0, "1"}};
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100, PlotRange → {{MinV, MaxV}, {-5, 0}}, AxesOrigin → {0, -5},
 Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```

$$\frac{\sqrt{\mathrm{wRR}}}{\sqrt{\frac{\mathrm{fR}*\mathrm{wR}}{2}+\sqrt{\mathrm{wRR}}}}$$




```
0.666667
```

```
Show[gLogDFfixWS10, gLogDFfixWS100, gLogBPfix, gLogFixLinear, gLogNumerical10,
 gLogNumerical100, gCI1♦10, gCI2♦10, gCI3♦10, gCI4♦10, gCI5♦10, gCIunder10,
 gCIupper10, gCI1♦100, gCI2♦100, gCI3♦100, gCI4♦100, gCI5♦100, gCIunder100,
 gCIupper100, PlotRange → {{MinV, MaxV}, {-5, 0}}, AxesOrigin → {-0.02, -5},
 Ticks → {Automatic, axeslabels}, AspectRatio → 0.75]
```



- **Fig.4(a)**

Linear approximation (Moran)

Parameter values

```
Clear[x]

Site = 100;
fR = 0.5;
βR = x;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

s = 0.05;

SELfM = 0;
SELβM = s;
SELβMM = 2 * s;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

```
MinV = 0;
MaxV = 1;
```

$$s_{average} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SEL}\beta\text{M} + \text{SELdM}\right) + \frac{1}{2} * (\text{SEL}\beta\text{MM} * \text{HETERO}\beta + \text{SELdM} * \text{HETEROd})$$

```
0.05
```

$$sol1 = Solve\left[\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}} == 0.05,\ x\right]$$

$$sol2 = Solve\left[\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}} == 0.2,\ x\right]$$

$$sol3 = Solve\left[\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}} == 0.4,\ x\right]$$

$$sol4 = Solve\left[\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}} == 0.6,\ x\right]$$

$$sol5 = Solve\left[\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}} == 0.8,\ x\right]$$

$$sol6 = Solve\left[\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}} == 0.95,\ x\right]$$

$\{\{x \to 1.444 \times 10^6\}\}$

$\{\{x \to 64\,000.\}\}$

$\{\{x \to 9000.\}\}$

$\{\{x \to 1777.78\}\}$

$\{\{x \to 250.\}\}$

$\{\{x \to 11.0803\}\}$

Out put

```
SampleMax = 10 000;
ChangeMax = 5;
OutputData = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulation

```
VectpM = Table[0, {SampleMax}];
Resultparameter = Table[0, {ChangeMax + 1}];
Result = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = (x /. sol1[[1]])];
  If[change == 2, x = (x /. sol2[[1]])];
  If[change == 3, x = (x /. sol3[[1]])];
  If[change == 4, x = (x /. sol4[[1]])];
  If[change == 5, x = (x /. sol5[[1]])];
  If[change == 6, x = (x /. sol6[[1]])];

  sample = 1;
```

```
For[sample = 1, sample ≤ SampleMax, sample = sample + 1,

 EqρH = Sqrt[βRR * dRR] / (Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]);
 xR = Round[Site * EqρH];
 xM = 0;
 xRR = Site - xR;
 xRM = 0;
 xMM = 0;
 sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), 2 * xRR/(xR + 2 * xRR)}]];
 xR = xR - sampleMutation[[1]];
 xM = xM + sampleMutation[[1]];
 xRR = xRR - sampleMutation[[2]];
 xRM = xRM + sampleMutation[[2]];
 xMM = 0;
 If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
 If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
 If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

 step = 2;
 While[0 < pM && pM < 1,
  pDeathR = dR * xR/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathM = dM * xM/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathRR = dRR * xRR/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathRM = dRM * xRM/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  pDeathMM = dMM * xMM/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
  sampleDeath =
   RandomVariate[MultinomialDistribution[1,
     {pDeathR, pDeathM, pDeathRR, pDeathRM, pDeathMM}]];
  xR = xR - sampleDeath[[1]];
  xM = xM - sampleDeath[[2]];
  xRR = xRR - sampleDeath[[3]];
  xRM = xRM - sampleDeath[[4]];
  xMM = xMM - sampleDeath[[5]];

  cR = βRR * xRR + βRM/2 * xRM;
  cM = βMM * xMM + βRM/2 * xRM;
  If[xR == 0 && xM == 0, cRR = 0];
  If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * βR^2 * xR^2/(βR * xR + βM * xM)];
  If[xR == 0 && xM == 0, cRM = 0];
```

```
        If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (βR * βM * xR * xM)/(βR * xR + βM * xM)];
        If[xR == 0 && xM == 0, cMM = 0];
        If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (βM^2 * xM^2)/(βR * xR + βM * xM)];
        pCellR = cR/(cR + cM + cRR + cRM + cMM);
        pCellM = cM/(cR + cM + cRR + cRM + cMM);
        pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
        pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
        pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
        sampleBirth =
         RandomVariate[MultinomialDistribution[1,
           {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

        xR = xR + sampleBirth[[1]];
        xM = xM + sampleBirth[[2]];
        xRR = xRR + sampleBirth[[3]];
        xRM = xRM + sampleBirth[[4]];
        xMM = xMM + sampleBirth[[5]];


        If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
        If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
        If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

        step = step + 1;
       ];

      VectpM[[sample]] = pM;

      OutputData[[countData, 1]] = EqρH;
      OutputData[[countData, 2]] = pM;
      countData = countData + 1;

     ];

    Resultparameter[[change]] = EqρH;
    Result[[change]] = Total[VectpM];

    OutputResult[[countResult, 1]] = EqρH;
    OutputResult[[countResult, 2]] = Total[VectpM];
    countResult = countResult + 1;

    Print[change]

   ];
```

1

2

3

4

5

6

Output of the data

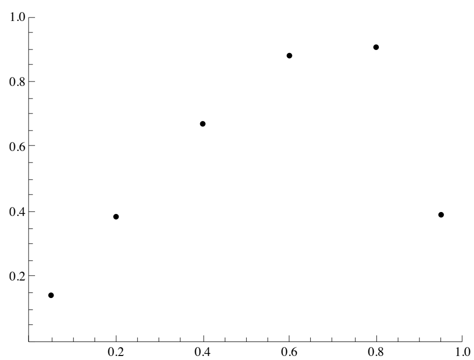
```
Resultparameter
Result
Export["/Users/Bessho/Desktop/workstation/OutputDataFig4a.txt", OutputData, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig4a.txt", OutputResult, "Table"];
```

{0.05, 0.2, 0.4, 0.6, 0.8, 0.95}

{71., 192., 335., 440., 453., 195.}

Plot

```
PlotpM = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM[[change, 1]] = Resultparameter[[change]];
  PlotpM[[change, 2]] = 1/s_average * Result[[change]]/SampleMax;
 ];
gNumerical = ListPlot[PlotpM, PlotRange → {{0, 1}, {0, 1}},
  PlotStyle → {Thickness[0.005], Black}, AspectRatio → 0.75]
PlotpM
```



{{0.05, 0.142}, {0.2, 0.384}, {0.4, 0.67}, {0.6, 0.88}, {0.8, 0.906}, {0.95, 0.39}}

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, Pr[$\underline{z}$ < u < $\bar{z}$] ~ 0.95.

z = 1.96;

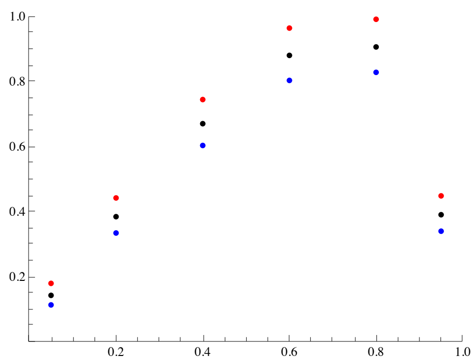
$$\underline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to SampleMax \text{ /. } \hat{u} \to \frac{Result}{SampleMax}$$

$$\overline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to SampleMax \text{ /. } \hat{u} \to \frac{Result}{SampleMax}$$

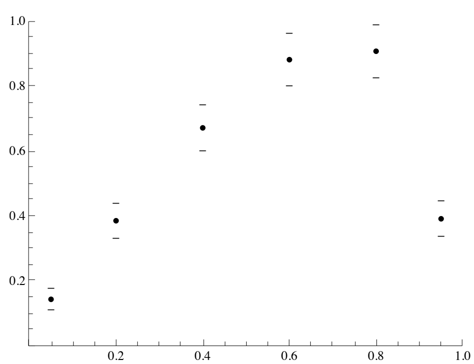
{0.00563309, 0.0166892, 0.0301485, 0.0401522, 0.0413956, 0.0169686}

{0.00894547, 0.0220801, 0.0372098, 0.048198, 0.0495536, 0.0224004}

```
PlotCIunder = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder[[change, 1]] = Resultparameter[[change]];
  PlotCIupper[[change, 1]] = Resultparameter[[change]];
  PlotCIunder[[change, 2]] = 1/s_average * z̲[[change]];
  PlotCIupper[[change, 2]] = 1/s_average * z̄[[change]];
];
gCIupperDot = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.005], Red}];
gCIunderDot = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.005], Blue}];
Show[gNumerical, gCIunderDot, gCIupperDot]
```

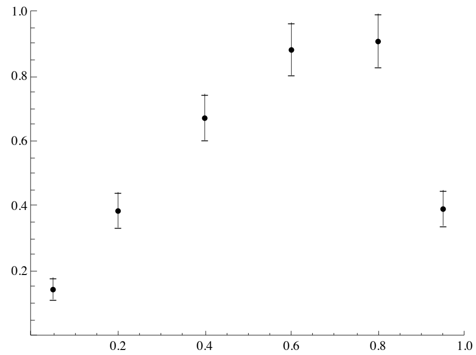



```
gCIupper = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
Show[gNumerical, gCIunder, gCIupper]
```




```
PlotCI1 = Table[0, {i, 2}, {j, 2}];
PlotCI2 = Table[0, {i, 2}, {j, 2}];
PlotCI3 = Table[0, {i, 2}, {j, 2}];
PlotCI4 = Table[0, {i, 2}, {j, 2}];
PlotCI5 = Table[0, {i, 2}, {j, 2}];
PlotCI6 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1[[1, 1]] = Resultparameter[[1]];
  PlotCI1[[1, 2]] = 1/s_average * z̲[[1]];
  PlotCI1[[2, 1]] = Resultparameter[[1]];
```

```
  PlotCI1[[2, 2]] = 1/s_average * z̄[[1]];
  PlotCI2[[1, 1]] = Resultparameter[[2]];
  PlotCI2[[1, 2]] = 1/s_average * z̲[[2]];
  PlotCI2[[2, 1]] = Resultparameter[[2]];
  PlotCI2[[2, 2]] = 1/s_average * z̄[[2]];
  PlotCI3[[1, 1]] = Resultparameter[[3]];
  PlotCI3[[1, 2]] = 1/s_average * z̲[[3]];
  PlotCI3[[2, 1]] = Resultparameter[[3]];
  PlotCI3[[2, 2]] = 1/s_average * z̄[[3]];
  PlotCI4[[1, 1]] = Resultparameter[[4]];
  PlotCI4[[1, 2]] = 1/s_average * z̲[[4]];
  PlotCI4[[2, 1]] = Resultparameter[[4]];
  PlotCI4[[2, 2]] = 1/s_average * z̄[[4]];
  PlotCI5[[1, 1]] = Resultparameter[[5]];
  PlotCI5[[1, 2]] = 1/s_average * z̲[[5]];
  PlotCI5[[2, 1]] = Resultparameter[[5]];
  PlotCI5[[2, 2]] = 1/s_average * z̄[[5]];
  PlotCI6[[1, 1]] = Resultparameter[[6]];
  PlotCI6[[1, 2]] = 1/s_average * z̲[[6]];
  PlotCI6[[2, 1]] = Resultparameter[[6]];
  PlotCI6[[2, 2]] = 1/s_average * z̄[[6]];
 ];
gCI1 = ListPlot[PlotCI1, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2 = ListPlot[PlotCI2, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3 = ListPlot[PlotCI3, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4 = ListPlot[PlotCI4, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5 = ListPlot[PlotCI5, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI6 = ListPlot[PlotCI6, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6, gCIunder, gCIupper]
```



Analytical result (Branching process)

```
Clear[y]

Site = 100;
fR = 0.5;
βR = y;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

s = 0.05;

SELfM = 0;
SELβM = s;
SELβMM = 2 * s;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\beta\text{RR} * \text{dRR}}}{\sqrt{\frac{\text{fR} * \beta\text{R} * \text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR} * \beta\text{R} * \text{dR}}{2}}}{\sqrt{\frac{\text{fR} * \beta\text{R} * \text{dR}}{2}} + \sqrt{\beta\text{RR} * \text{dRR}}};$$

$$\text{FaiR} = \frac{\beta\text{R}}{\text{dR}};$$

$$\text{FaiM} = \frac{\beta\text{M}}{\text{dM}};$$

$$FaiRR = \frac{\beta RR}{dRR};$$

$$FaiRM = \frac{\beta RM}{dRM};$$

$$avef = \frac{fM + fR}{2};$$

$$FaiR2 = \frac{fR}{2} * FaiR;$$

$$FaiM2 = \frac{avef}{2} * FaiM;$$

$$BPPaiH = \frac{FaiM2 * FaiRM - FaiR2 * FaiRR}{FaiM2 * FaiRM + \frac{FaiRM}{2} * \sqrt{FaiR2 * FaiRR}};$$

$$BPPaiD = \frac{FaiM2 * FaiRM - FaiR2 * FaiRR}{FaiM2 * FaiRM + 2 * FaiM2 * \sqrt{FaiR2 * FaiRR}};$$

$$BPfix = \frac{Eq\rho H}{Eq\rho H + 2 * Eq\rho D} * BPPaiH + \frac{2 * Eq\rho D}{Eq\rho H + 2 * Eq\rho D} * BPPaiD;$$

```
MinV = 0;
MaxV = 10;
RepeatMax = 100;
PlotBP = Table[0, {i, RepeatMax}, {j, 2}];
For[repeat = 1, repeat ≤ RepeatMax - 1, repeat = repeat + 1,
```

$$y1 = MinV + \frac{(MaxV - MinV)}{RepeatMax} * (repeat - 1);$$

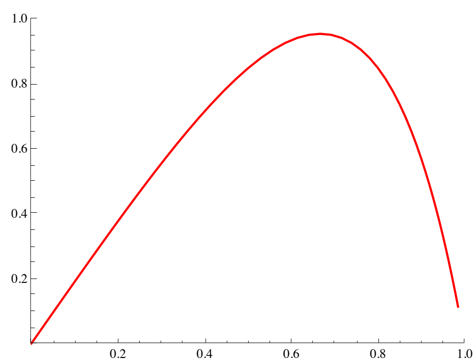
$$fracH = \frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}} \;/.\; y \to 10^{y1};$$

$$U = BPfix \;/.\; y \to 10^{y1};$$

$$PlotBP[[repeat, 1]] = fracH;$$

$$PlotBP[[repeat, 2]] = \frac{U}{s_{average}};$$

```
];
gBPfix = ListPlot[PlotBP, PlotRange → {{0, 1}, {0, 1}}, PlotStyle → {Thickness[0.005], Red},
  Joined → True, AspectRatio → 0.75]
```



Analytical result (Diffusion/Additive gene)

```
Parameters = {SUM → Site, d_R → dR, d_RR → dRR, ρ̂_D → 1 - ρ̂_H, sfM → SELfM, sβM → SELβM,
   sβRM → SELβRM, sβMM → SELβMM, sdM → SELdM, sdRM → SELdRM, sdMM → SELdMM }
```

$$\overline{d} = \frac{d_R + d_{RR}}{2} \text{ /. Parameters;}$$

$$s_e = 2 * s_{average};$$

$$N_e = \frac{2 * \hat{\rho}_H * \hat{\rho}_D}{\frac{d_R}{\overline{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\overline{d}} * \hat{\rho}_D} * \text{SUM};$$

$$\text{INI} = \frac{1}{\text{SUM} * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)} \text{ /. Parameters;}$$

```
EffectiveS = s_e /. Parameters;
EffectiveN = N_e /. Parameters;
```

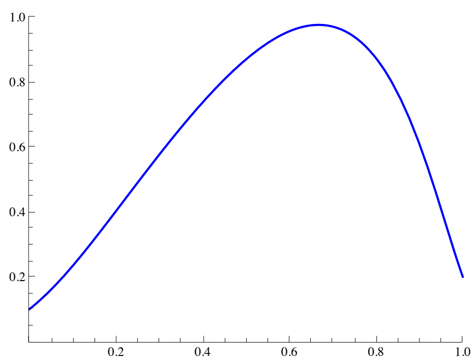
$$\text{DFfixWS2} = \frac{1 - \text{Exp}[-2 * \text{EffectiveS} * \text{EffectiveN} * \text{INI}]}{1 - \text{Exp}[-2 * \text{EffectiveS} * \text{EffectiveN}]}$$

$$\text{gDFfixWS2} = \text{Plot}\left[\frac{\text{DFfixWS2}}{s_{average}}, \{\hat{\rho}_H, 0, 1\}, \text{PlotRange} \to \{\{0, 1\}, \{0, 1\}\},\right.$$
$$\left.\text{PlotStyle} \to \{\text{Thickness}[0.005], \text{Blue}\}, \text{AspectRatio} \to 0.75\right];$$

```
Show[gDFfixWS2]
```

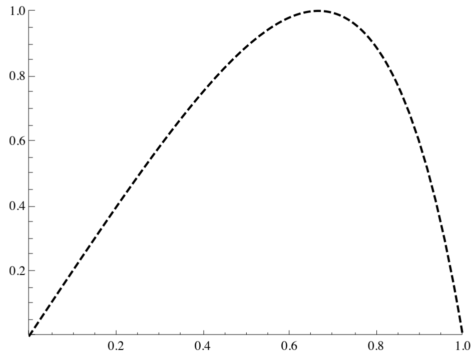
$\{\text{SUM} \to 100, d_R \to 0.005, d_{RR} \to 0.005, \hat{\rho}_D \to 1 - \hat{\rho}_H, \text{sfM} \to 0,$
$\text{s}\beta\text{M} \to 0.05, \text{s}\beta\text{RM} \to 0.05, \text{s}\beta\text{MM} \to 0.1, \text{sdM} \to 0, \text{sdRM} \to 0., \text{sdMM} \to 0\}$

$$\frac{1 - e^{-\frac{0.4 (1-\hat{\rho}_H) \hat{\rho}_H}{(2 (1-\hat{\rho}_H)+\hat{\rho}_H) (2. (1-\hat{\rho}_H)+1. \hat{\rho}_H)}}}{1 - e^{-\frac{40. (1-\hat{\rho}_H) \hat{\rho}_H}{2. (1-\hat{\rho}_H)+1. \hat{\rho}_H}}}$$



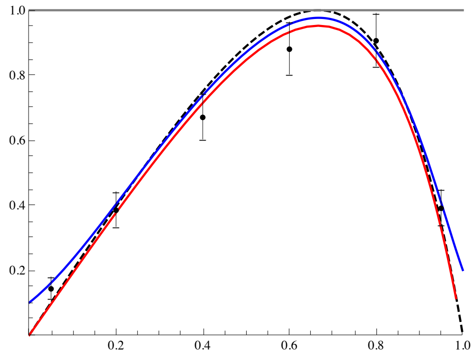
Analytical result (Linear approximation)

```
Ulinear = (8 * ρH * ρD)/(ρH + 2 * ρD)^2 * saverage /. ρD → 1 - ρH;
gUlinear = Plot[Ulinear/saverage /. Parameters, {ρH, 0, 1}, PlotRange → {{0, 1}, {0, 1.0}},
   PlotStyle → {Thickness[0.005], Black, Dashed}, AspectRatio → 0.75];
Show[gUlinear]
```



Fully hap and dip

```
fully = Plot[1 + 0 * ρH /. Parameters, {ρH, 0, 1}, PlotRange → {{0, 1}, {0, 2.0}},
   PlotStyle → {Thickness[0.005], Gray}, AspectRatio → 0.75];
Show[gUlinear, gDFfixWS2, gBPfix, gNumerical, fully, gCI1, gCI2, gCI3, gCI4, gCI5,
 gCI6, gCIunder, gCIupper, AspectRatio → 0.75]
```



- **Fig.4(b)**

Linear approximation (WF)

Parameter values

```
Clear[x]

Site = 100;

fR = 0.5;
wR = x;
wRR = 1000;

s = 0.05;

SELfM = 0;
SELwM = s;
SELwMM = 2 * s;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\text{SELwRM} = \frac{\text{wRM} - \text{wRR}}{\text{wRR}};$$

```
MinV = 0;
MaxV = 1;
```

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SELwM}\right) + \frac{1}{2} * (\text{SELwMM} * \text{HETEROw})$$

```
0.05
```

$$\text{sol1} = \text{Solve}\left[\frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}} == 0.05, x\right]$$

$$\text{sol2} = \text{Solve}\left[\frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}} == 0.2, x\right]$$

$$\text{sol3} = \text{Solve}\left[\frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}} == 0.4, x\right]$$

$$\text{sol4} = \text{Solve}\left[\frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}} == 0.6, x\right]$$

$$\text{sol5} = \text{Solve}\left[\frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}} == 0.8, x\right]$$

$$\text{sol6} = \text{Solve}\left[\frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}} == 0.95, x\right]$$

$\{\{x \to 1.444 \times 10^6\}\}$

$\{\{x \to 64\,000.\}\}$

$\{\{x \to 9000.\}\}$

$\{\{x \to 1777.78\}\}$

$\{\{x \to 250.\}\}$

$\{\{x \to 11.0803\}\}$

Out put

```
SampleMax = 10 000;
ChangeMax = 5;
OutputData = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulaton

```
VectpM = Table[0, {SampleMax}];
Resultparameter = Table[0, {ChangeMax + 1}];
Result = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = (x /. sol1[[1]])];
  If[change == 2, x = (x /. sol2[[1]])];
  If[change == 3, x = (x /. sol3[[1]])];
  If[change == 4, x = (x /. sol4[[1]])];
  If[change == 5, x = (x /. sol5[[1]])];
  If[change == 6, x = (x /. sol6[[1]])];

  sample = 1;
```

```
For[sample = 1, sample ≤ SampleMax, sample = sample + 1,

 EqρH = Sqrt[wRR] / Sqrt[(fR*wR)/2 + Sqrt[wRR]];

 xR = Round[Site * EqρH];
 xM = 0;
 xRR = Site - xR;
 xRM = 0;
 xMM = 0;

 sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];

 xR = xR - sampleMutation[[1]];
 xM = xM + sampleMutation[[1]];
 xRR = xRR - sampleMutation[[2]];
 xRM = xRM + sampleMutation[[2]];
 xMM = 0;

 If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
 If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
 If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

 step = 2;
 While[0 < pM && pM < 1,
  cR = wRR * xRR + wRM/2 * xRM;
  cM = wMM * xMM + wRM/2 * xRM;
  If[xR == 0 && xM == 0, cRR = 0];
  If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (wR^2 * xR^2)/(wR * xR + wM * xM)];
  If[xR == 0 && xM == 0, cRM = 0];
  If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (wR * wM * xR * xM)/(wR * xR + wM * xM)];
  If[xR == 0 && xM == 0, cMM = 0];
  If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (wM^2 * xM^2)/(wR * xR + wM * xM)];
  pCellR = cR/(cR + cM + cRR + cRM + cMM);
  pCellM = cM/(cR + cM + cRR + cRM + cMM);
  pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
  pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
  pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
  sampleBirth = RandomVariate[MultinomialDistribution[Site,
     {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

  xR = sampleBirth[[1]];
  xM = sampleBirth[[2]];
```

```
      xRR = sampleBirth[[3]];
      xRM = sampleBirth[[4]];
      xMM = sampleBirth[[5]];
      If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
      If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
      If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

      step = step + 1;
     ];

    VectpM[[sample]] = pM;

    OutputData[[countData, 1]] = EqρH;
    OutputData[[countData, 2]] = pM;
    countData = countData + 1;

   ];

  Resultparameter[[change]] = EqρH;
  Result[[change]] = Total[VectpM];

  OutputResult[[countResult, 1]] = EqρH;
  OutputResult[[countResult, 2]] = Total[VectpM];
  countResult = countResult + 1;

  Print[change]

 ];
```

1

2

3

4

5

6

Output of the data

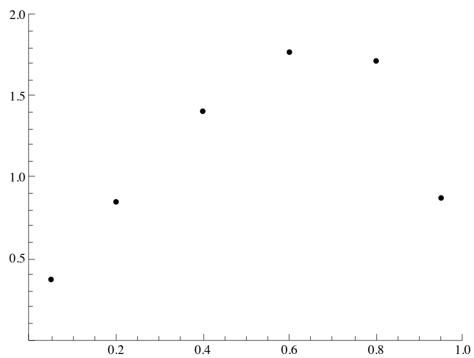
```
Resultparameter
Result
Export["/Users/Bessho/Desktop/workstation/OutputDataFig4b.txt", OutputData, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig4b.txt", OutputResult, "Table"];
```

{0.05, 0.2, 0.4, 0.6, 0.8, 0.95}

{185, 424, 703, 885, 858, 436}

Plot

```
PlotpM = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM[[change, 1]] = Resultparameter[[change]];
  PlotpM[[change, 2]] = 1/s_average * Result[[change]]/SampleMax;
 ];
gNumerical = ListPlot[PlotpM, PlotRange → {{0, 1}, {0, 2.0}},
  PlotStyle → {Thickness[0.005], Black}, AspectRatio → 0.75]
PlotpM
```




```
{{0.05, 0.37}, {0.2, 0.848}, {0.4, 1.406}, {0.6, 1.77}, {0.8, 1.716}, {0.95, 0.872}}
```

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \overline{z}] \sim 0.95$.

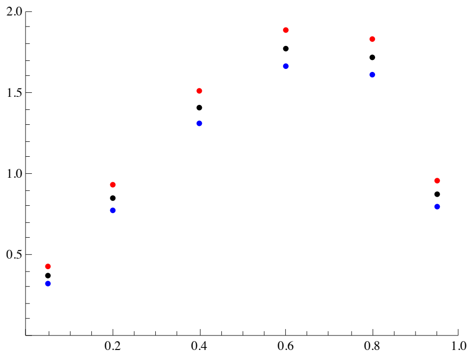
```
z = 1.96;
```

$$\underline{z} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to \text{SampleMax /. } \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

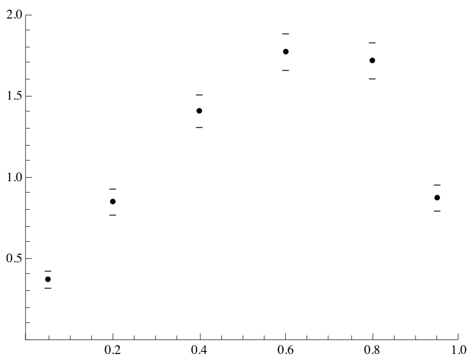
$$\overline{z} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) \text{ /. } n \to \text{SampleMax /. } \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

```
{0.0160378, 0.0386232, 0.0654525, 0.08309, 0.0804685, 0.0397698}
{0.021332, 0.0465283, 0.0754775, 0.094226, 0.0914497, 0.0477807}
```

```
PlotCIunder = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder[[change, 1]] = Resultparameter[[change]];
  PlotCIupper[[change, 1]] = Resultparameter[[change]];
  PlotCIunder[[change, 2]] = 1/s_average * z[[change]];
  PlotCIupper[[change, 2]] = 1/s_average * z̅[[change]];
 ];
gCIupperDot = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.005], Red}];
gCIunderDot = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.005], Blue}];
Show[gNumerical, gCIunderDot, gCIupperDot]
```

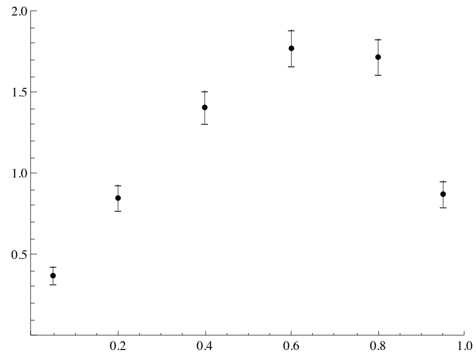


```
gCIupper = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
gCIunder = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"-"}];
Show[gNumerical, gCIunder, gCIupper]
```



```
PlotCI1 = Table[0, {i, 2}, {j, 2}];
PlotCI2 = Table[0, {i, 2}, {j, 2}];
PlotCI3 = Table[0, {i, 2}, {j, 2}];
PlotCI4 = Table[0, {i, 2}, {j, 2}];
PlotCI5 = Table[0, {i, 2}, {j, 2}];
PlotCI6 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1[[1, 1]] = Resultparameter[[1]];
  PlotCI1[[1, 2]] = 1/s_average * z[[1]];
  PlotCI1[[2, 1]] = Resultparameter[[1]];
```

```
    PlotCI1[[2, 2]] = 1/s_average * z̄[[1]];
    PlotCI2[[1, 1]] = Resultparameter[[2]];
    PlotCI2[[1, 2]] = 1/s_average * z̲[[2]];
    PlotCI2[[2, 1]] = Resultparameter[[2]];
    PlotCI2[[2, 2]] = 1/s_average * z̄[[2]];
    PlotCI3[[1, 1]] = Resultparameter[[3]];
    PlotCI3[[1, 2]] = 1/s_average * z̲[[3]];
    PlotCI3[[2, 1]] = Resultparameter[[3]];
    PlotCI3[[2, 2]] = 1/s_average * z̄[[3]];
    PlotCI4[[1, 1]] = Resultparameter[[4]];
    PlotCI4[[1, 2]] = 1/s_average * z̲[[4]];
    PlotCI4[[2, 1]] = Resultparameter[[4]];
    PlotCI4[[2, 2]] = 1/s_average * z̄[[4]];
    PlotCI5[[1, 1]] = Resultparameter[[5]];
    PlotCI5[[1, 2]] = 1/s_average * z̲[[5]];
    PlotCI5[[2, 1]] = Resultparameter[[5]];
    PlotCI5[[2, 2]] = 1/s_average * z̄[[5]];
    PlotCI6[[1, 1]] = Resultparameter[[6]];
    PlotCI6[[1, 2]] = 1/s_average * z̲[[6]];
    PlotCI6[[2, 1]] = Resultparameter[[6]];
    PlotCI6[[2, 2]] = 1/s_average * z̄[[6]];
  ];
gCI1 = ListPlot[PlotCI1, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2 = ListPlot[PlotCI2, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3 = ListPlot[PlotCI3, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4 = ListPlot[PlotCI4, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5 = ListPlot[PlotCI5, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI6 = ListPlot[PlotCI6, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6, gCIunder, gCIupper]
```



Analytical result (Branching process)

```
Clear[y]

Site = 100;
fR = 0.5;
wR = y;
wRR = 1000;

s = 0.05;

SELfM = 0;
SELwM = s;
SELwMM = 2 * s;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\text{SELwRM} = \frac{\text{wRM} - \text{wRR}}{\text{wRR}};$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR}*\text{wR}}{2}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{avef} = \frac{\text{fM} + \text{fR}}{2};$$

$$\text{W} = \frac{\text{fR} * \text{wR}}{2} * \text{Eq}\rho\text{H} + \text{wRR} * \text{Eq}\rho\text{D};$$

$$\text{EqD3a} = 1 - \text{PH} - \text{Exp}\left[-\text{PD} * \frac{\text{avef} * \text{wM}}{\text{W}}\right];$$

$$\text{EqD3b} = 1 - \text{PD} - \text{Exp}\left[-\text{PH} * \frac{\text{wRM}}{2 * \text{W}}\right];$$

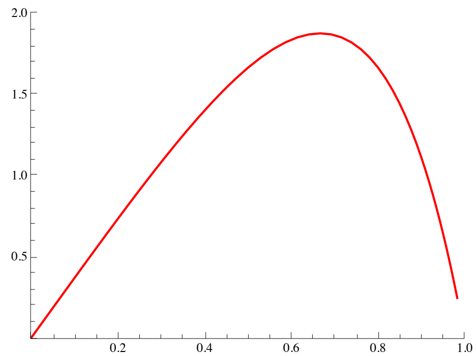
```
MinV = 0;
MaxV = 10;
RepeatMax = 100;
ResultBPfix = Table[0, {i, (RepeatMax)}, {j, 2}];
repeat = 0;
For[repeat1 = 1, repeat1 ≤ RepeatMax, repeat1 = repeat1 + 1,
  repeat = repeat + 1;
  y1 = 0 + (MaxV - MinV)/RepeatMax * repeat1;
  CondH = EqD3a /. y → 10^y1;
  CondD = EqD3b /. y → 10^y1;
  sol = NSolve[CondH == 0 && CondD == 0 && 0 < PH && PH < 1 && 0 < PD && PD < 1, {PH, PD}, Reals];
  BPPaiH = Part[(PH /. sol), 1];
  BPPaiD = Part[(PD /. sol), 1];

  fracH = Sqrt[wRR] / (Sqrt[(fR*wR)/2] + Sqrt[wRR]) /. y → 10^y1;

  U = EqρH/(EqρH + 2*EqρD) * BPPaiH + (2*EqρD)/(EqρH + 2*EqρD) * BPPaiD /. y → 10^y1;

  ResultBPfix[[repeat, 1]] = fracH;
  ResultBPfix[[repeat, 2]] = U/s_average;
 ];
gBPfix = ListPlot[ResultBPfix, Joined → True, PlotRange → {{0, 1}, {0, 2}},
   PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75];
Show[gBPfix]
```

NSolve::ratnz : NSolveは厳密でない係数を持つ系を解くことができませんでした．答は対応する厳密系を解いて，結果を数値化することで得られました． ≫

NSolve::ratnz : NSolveは厳密でない係数を持つ系を解くことができませんでした．答は対応する厳密系を解いて，結果を数値化することで得られました． ≫

NSolve::ratnz : NSolveは厳密でない係数を持つ系を解くことができませんでした．答は対応する厳密系を解いて，結果を数値化することで得られました． ≫

General::stop : この計算中に，NSolve::ratnzのこれ以上の出力は表示されません． ≫



Analytical result (Diffusion/Additive gene)

$$\text{Parameters} = \left\{\text{SUM} \to \text{Site},\ \hat{\rho}_D \to 1 - \hat{\rho}_H,\ \text{sfM} \to \text{SELfM},\ \text{swM} \to \text{SELwM},\ \text{swRM} \to \text{SELwRM},\ \text{swMM} \to \text{SELwMM}\right\}$$

$$s_e = 2 * s_{average};$$

$$N_e = \frac{4 * \hat{\rho}_H * \hat{\rho}_D}{\hat{\rho}_H + 2 * \hat{\rho}_D} * \text{SUM};$$

$$\text{INI} = \frac{1}{\text{SUM} * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)} \text{ /. Parameters};$$

$$\text{EffectiveS} = s_e \text{ /. Parameters};$$

$$\text{EffectiveN} = N_e \text{ /. Parameters};$$

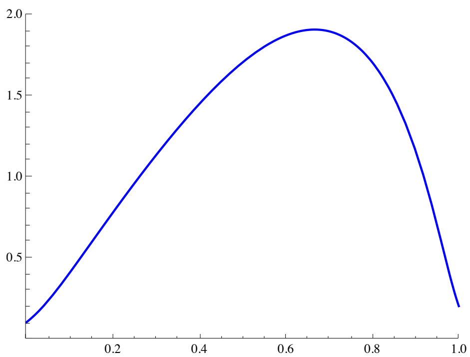
$$\text{DFfixWS2} = \frac{1 - \text{Exp}[-2 * \text{EffectiveS} * \text{EffectiveN} * \text{INI}]}{1 - \text{Exp}[-2 * \text{EffectiveS} * \text{EffectiveN}]}$$

$$\text{gDFfixWS2} = \text{Plot}\left[\frac{\text{DFfixWS2}}{s_{average}},\ \left\{\hat{\rho}_H, 0, 1\right\},\ \text{PlotRange} \to \{\{0, 1\}, \{0, 2\}\},\ \text{PlotStyle} \to \{\text{Thickness}[0.005], \text{Blue}\},\ \text{AspectRatio} \to 0.75\right];$$

$$\text{Show}[\text{gDFfixWS2}]$$

$$\left\{\text{SUM} \to 100,\ \hat{\rho}_D \to 1 - \hat{\rho}_H,\ \text{sfM} \to 0,\ \text{swM} \to 0.05,\ \text{swRM} \to 0.05,\ \text{swMM} \to 0.1\right\}$$

$$\frac{1 - e^{-\frac{0.8\,(1-\hat{\rho}_H)\,\hat{\rho}_H}{(2\,(1-\hat{\rho}_H)+\hat{\rho}_H)^2}}}{1 - e^{-\frac{80.\,(1-\hat{\rho}_H)\,\hat{\rho}_H}{2\,(1-\hat{\rho}_H)+\hat{\rho}_H}}}$$

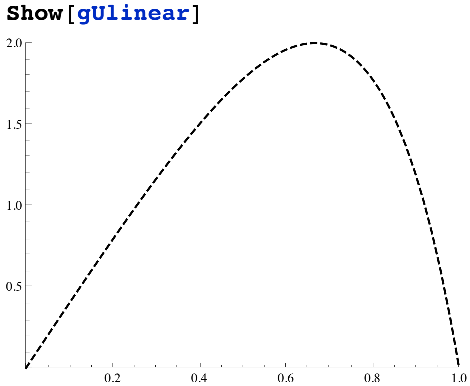


Analytical result (Linear approximation)

$$\text{Ulinear} = \frac{8 * \rho_H * \rho_D}{(\rho_H + 2 * \rho_D)^2} * 2 * s_{average} \text{ /. } \rho_D \to 1 - \rho_H;$$

$$\text{gUlinear} = \text{Plot}\left[\frac{\text{Ulinear}}{s_{average}} \text{ /. Parameters},\ \{\rho_H, 0, 1\},\ \text{PlotRange} \to \{\{0, 1\}, \{0, 2.0\}\},\ \text{PlotStyle} \to \{\text{Thickness}[0.005], \text{Black}, \text{Dashed}\},\ \text{AspectRatio} \to 0.75\right];$$

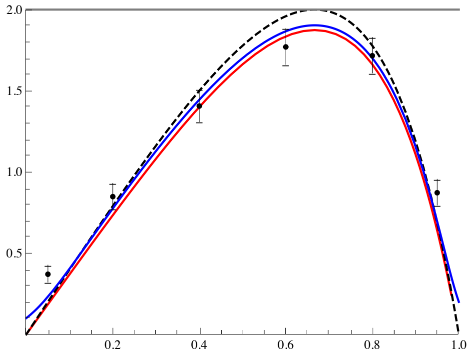
$$\text{Show}[\text{gUlinear}]$$



Fully hap and dip

$$\text{fully} = \text{Plot}[2 + 0 * \rho_H \text{ /. Parameters},\ \{\rho_H, 0, 1\},\ \text{PlotRange} \to \{\{0, 1\}, \{0, 2.0\}\},\ \text{PlotStyle} \to \{\text{Thickness}[0.005], \text{Gray}\},\ \text{AspectRatio} \to 0.75];$$

```
Show[gBPfix, gUlinear, gDFfixWS2, gNumerical, fully, gCI1, gCI2, gCI3, gCI4, gCI5,
 gCI6, gCIunder, gCIupper, AspectRatio → 0.75]
```



- **Fig.5(a)**

Linear approximation (Moran)

Parameter values (dR << dRR)

```
Site = 100;
fR = 0.5;
βR = x;
βRR = 1000;

dR = 0.0005;
dRR = 0.005;

s = 0.05;

SELfM = 0;
SELβM = s;
SELβMM = 2 * s;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$SEL\beta RM = \frac{\beta RM - \beta RR}{\beta RR};$$

$$SELdRM = \frac{dRR - dRM}{dRR};$$

```
MinV = 0;
MaxV = 1;
```

$$\frac{dR}{\frac{dR}{dR+dRR}}$$

$$\frac{dR}{dR + dRR}$$

$$s_{average} = \frac{1}{2} * \left(\frac{SELfM}{2} + SEL\beta M + SELdM\right) + \frac{1}{2} * (SEL\beta MM * HETERO\beta + SELdM * HETEROd)$$

```
0.0055
0.0909091
0.05
```

```
sol1 = Solve[Sqrt[βRR * dRR] / (Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]) == 0.05, x]

sol2 = Solve[Sqrt[βRR * dRR] / (Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]) == 0.2, x]

sol3 = Solve[Sqrt[βRR * dRR] / (Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]) == 0.5, x]

sol4 = Solve[Sqrt[βRR * dRR] / (Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]) == 0.8, x]

sol5 = Solve[Sqrt[βRR * dRR] / (Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]) == 0.95, x]
```

$\{\{x \to 1.444 \times 10^{7}\}\}$

$\{\{x \to 640\,000.\}\}$

$\{\{x \to 40\,000.\}\}$

$\{\{x \to 2500.\}\}$

$\{\{x \to 110.803\}\}$

Out put

```
SampleMax = 10 000;
ChangeMax = 4;
OutputData1 = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult1 = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulation

```
VectpM = Table[0, {SampleMax}];
Resultparameter1 = Table[0, {ChangeMax + 1}];
Result1 = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = (x /. sol1[[1]])];
  If[change == 2, x = (x /. sol2[[1]])];
  If[change == 3, x = (x /. sol3[[1]])];
  If[change == 4, x = (x /. sol4[[1]])];
  If[change == 5, x = (x /. sol5[[1]])];

  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,

   EqρH = Sqrt[βRR * dRR] / (Sqrt[fR*βR*dR/2] + Sqrt[βRR * dRR]);
```

```
xR = Round[Site * EqρH];
xM = 0;
xRR = Site - xR;
xRM = 0;
xMM = 0;
sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), 2 * xRR/(xR + 2 * xRR)}]];
xR = xR - sampleMutation[[1]];
xM = xM + sampleMutation[[1]];
xRR = xRR - sampleMutation[[2]];
xRM = xRM + sampleMutation[[2]];
xMM = 0;
If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

step = 2;
While[0 < pM && pM < 1,
 pDeathR = dR * xR/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 pDeathM = dM * xM/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 pDeathRR = dRR * xRR/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 pDeathRM = dRM * xRM/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 pDeathMM = dMM * xMM/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 sampleDeath =
  RandomVariate[MultinomialDistribution[1,
    {pDeathR, pDeathM, pDeathRR, pDeathRM, pDeathMM}]];
 xR = xR - sampleDeath[[1]];
 xM = xM - sampleDeath[[2]];
 xRR = xRR - sampleDeath[[3]];
 xRM = xRM - sampleDeath[[4]];
 xMM = xMM - sampleDeath[[5]];


 cR = βRR * xRR + βRM/2 * xRM;
 cM = βMM * xMM + βRM/2 * xRM;
 If[xR == 0 && xM == 0, cRR = 0];
 If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (βR^2 * xR^2)/(βR * xR + βM * xM)];
 If[xR == 0 && xM == 0, cRM = 0];
 If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (βR * βM * xR * xM)/(βR * xR + βM * xM)];
 If[xR == 0 && xM == 0, cMM = 0];
 If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (βM^2 * xM^2)/(βR * xR + βM * xM)];
 pCellR = cR/(cR + cM + cRR + cRM + cMM);
```

```
        pCellM = cM / (cR + cM + cRR + cRM + cMM);
        pCellRR = cRR / (cR + cM + cRR + cRM + cMM);
        pCellRM = cRM / (cR + cM + cRR + cRM + cMM);
        pCellMM = cMM / (cR + cM + cRR + cRM + cMM);
        sampleBirth =
         RandomVariate[MultinomialDistribution[1,
           {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

        xR = xR + sampleBirth[[1]];
        xM = xM + sampleBirth[[2]];
        xRR = xRR + sampleBirth[[3]];
        xRM = xRM + sampleBirth[[4]];
        xMM = xMM + sampleBirth[[5]];


        If[xRR + xRM + xMM == 0, pM = xM / (xR + xM)];

        If[xR + xM == 0, pM = (xRM/2 + xMM) / (xRR + xRM + xMM)];

        If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR / (dR + dRR) * (xRM/2 + xMM) / (xRR + xRM + xMM) + dR / (dR + dRR) * xM / (xR + xM)];

        step = step + 1;
       ];

      VectpM[[sample]] = pM;

      OutputData1[[countData, 1]] = EqρH;
      OutputData1[[countData, 2]] = pM;
      countData = countData + 1;

     ];

    Resultparameter1[[change]] = EqρH;
    Result1[[change]] = Total[VectpM];
    OutputResult1[[countResult, 1]] = EqρH;
    OutputResult1[[countResult, 2]] = Total[VectpM];
    countResult = countResult + 1;

    Print[change]

   ];
```

1

2

3

4

5

変数クリア

```
Clear[x]
```

Parameter values (dR << dRR)

```
fR = 0.5;
βR = x;
βRR = 1000;

dR = 0.005;
dRR = 0.0005;

s = 0.05;

SELfM = 0;
SELβM = s;
SELβMM = 2 * s;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

$$\frac{\text{dR}}{\frac{\text{dR}}{\text{dR}+\text{dRR}}}$$

$$\frac{\text{dR}}{\text{dR} + \text{dRR}}$$

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SEL}\beta\text{M} + \text{SELdM}\right) + \frac{1}{2} * (\text{SEL}\beta\text{MM} * \text{HETERO}\beta + \text{SELdM} * \text{HETEROd})$$

```
0.0055
0.909091
0.05
```

$$\text{sol1} = \text{Solve}\left[\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}} == 0.05,\ x\right]$$

$$\text{sol2} = \text{Solve}\left[\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}} == 0.2,\ x\right]$$

$$\text{sol3} = \text{Solve}\left[\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}} == 0.5,\ x\right]$$

$$\text{sol4} = \text{Solve}\left[\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}} == 0.8,\ x\right]$$

$$\text{sol5} = \text{Solve}\left[\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}} == 0.95,\ x\right]$$

```
{{x → 144 400.}}
{{x → 6400.}}
{{x → 400.}}
{{x → 25.}}
{{x → 1.10803}}
```

Out put

```
OutputData2 = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult2 = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulation

```
VectpM = Table[0, {SampleMax}];
Resultparameter2 = Table[0, {ChangeMax + 1}];
Result2 = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = (x /. sol1[[1]])];
  If[change == 2, x = (x /. sol2[[1]])];
  If[change == 3, x = (x /. sol3[[1]])];
  If[change == 4, x = (x /. sol4[[1]])];
  If[change == 5, x = (x /. sol5[[1]])];

  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,
```

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}};$$

```
   xR = Round[Site * EqρH];
   xM = 0;
```

```
xRR = Site - xR;
xRM = 0;
xMM = 0;
sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), 2 * xRR/(xR + 2 * xRR)}]];
xR = xR - sampleMutation[[1]];
xM = xM + sampleMutation[[1]];
xRR = xRR - sampleMutation[[2]];
xRM = xRM + sampleMutation[[2]];
xMM = 0;
pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM);


step = 2;
While[0 < pM && pM < 1,
 pDeathR = dR * xR/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 pDeathM = dM * xM/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 pDeathRR = dRR * xRR/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 pDeathRM = dRM * xRM/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 pDeathMM = dMM * xMM/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
 sampleDeath =
  RandomVariate[MultinomialDistribution[1,
    {pDeathR, pDeathM, pDeathRR, pDeathRM, pDeathMM}]];
 xR = xR - sampleDeath[[1]];
 xM = xM - sampleDeath[[2]];
 xRR = xRR - sampleDeath[[3]];
 xRM = xRM - sampleDeath[[4]];
 xMM = xMM - sampleDeath[[5]];

 cR = βRR * xRR + βRM/2 * xRM;
 cM = βMM * xMM + βRM/2 * xRM;
 If[xR == 0 && xM == 0, cRR = 0];
 If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * βR^2 * xR^2/(βR * xR + βM * xM)];
 If[xR == 0 && xM == 0, cRM = 0];
 If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * βR * βM * xR * xM/(βR * xR + βM * xM)];
 If[xR == 0 && xM == 0, cMM = 0];
 If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * βM^2 * xM^2/(βR * xR + βM * xM)];
 pCellR = cR/(cR + cM + cRR + cRM + cMM);
 pCellM = cM/(cR + cM + cRR + cRM + cMM);
 pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
 pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
```

```
       pCellMM = cMM / (cR + cM + cRR + cRM + cMM);
       sampleBirth =
        RandomVariate[MultinomialDistribution[1,
          {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

       xR = xR + sampleBirth[[1]];
       xM = xM + sampleBirth[[2]];
       xRR = xRR + sampleBirth[[3]];
       xRM = xRM + sampleBirth[[4]];
       xMM = xMM + sampleBirth[[5]];

       pM = dRR / (dR + dRR) * (xRM/2 + xMM) / (xRR + xRM + xMM) + dR / (dR + dRR) * xM / (xR + xM);

       step = step + 1;
      ];

      VectpM[[sample]] = pM;

      OutputData2[[countData, 1]] = EqρH;
      OutputData2[[countData, 2]] = pM;
      countData = countData + 1;

     ];

     Resultparameter2[[change]] = EqρH;
     Result2[[change]] = Total[VectpM];
     OutputResult2[[countResult, 1]] = EqρH;
     OutputResult2[[countResult, 2]] = Total[VectpM];
     countResult = countResult + 1;

     Print[change]

    ];
```

1

2

3

4

5

Output of the data

```
Resultparameter1
Result1
Resultparameter2
Result2
Export["/Users/Bessho/Desktop/workstation/OutputDataFig5a1.txt", OutputData1, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig5a1.txt", OutputResult1,
  "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputDataFig5a2.txt", OutputData2, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig5a2.txt", OutputResult2,
  "Table"];
```

{0.05, 0.2, 0.5, 0.8, 0.95}

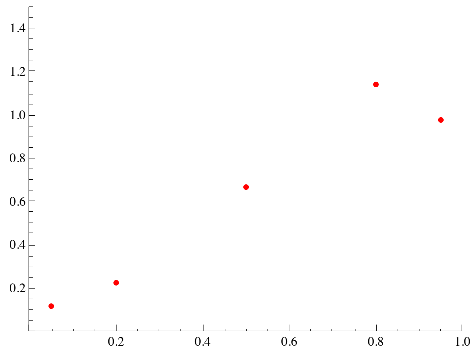
{58., 112., 333., 570., 488.}

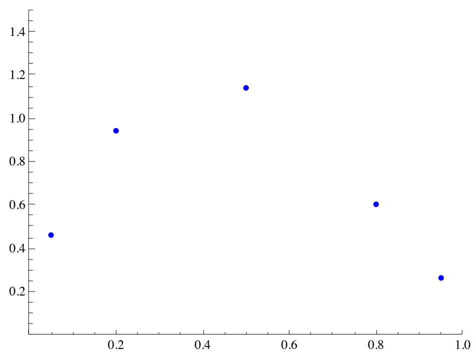
{0.05, 0.2, 0.5, 0.8, 0.95}

{230., 471., 570., 301., 131.}

Plot

```
PlotpM1 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpM2 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM1[[change, 1]] = Resultparameter1[[change]];
  PlotpM1[[change, 2]] = 1/s_average * Result1[[change]]/SampleMax;
  PlotpM2[[change, 1]] = Resultparameter2[[change]];
  PlotpM2[[change, 2]] = 1/s_average * Result2[[change]]/SampleMax;
 ];
gNumerical1 = ListPlot[PlotpM1, PlotRange → {{0, 1}, {0, 1.5}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
gNumerical2 = ListPlot[PlotpM2, PlotRange → {{0, 1}, {0, 1.5}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
PlotpM1
PlotpM2
```





```
{{0.05, 0.116}, {0.2, 0.224}, {0.5, 0.666}, {0.8, 1.14}, {0.95, 0.976}}
{{0.05, 0.46}, {0.2, 0.942}, {0.5, 1.14}, {0.8, 0.602}, {0.95, 0.262}}
```

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \bar{z}] \sim 0.95$.

```
z = 1.96;
```

$$\underline{z1} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2 * n} * z^2 - z * \sqrt{\frac{1}{n} * \hat{u} * \left(1 - \hat{u}\right) + \frac{1}{4 * n^2} * z^2}\right) \text{ /. } n \to \text{SampleMax /. } \hat{u} \to \frac{\text{Result1}}{\text{SampleMax}}$$

$$\overline{z1} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2 * n} * z^2 + z * \sqrt{\frac{1}{n} * \hat{u} * \left(1 - \hat{u}\right) + \frac{1}{4 * n^2} * z^2}\right) \text{ /. } n \to \text{SampleMax /. } \hat{u} \to \frac{\text{Result1}}{\text{SampleMax}}$$

$$\underline{z2} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2 * n} * z^2 - z * \sqrt{\frac{1}{n} * \hat{u} * \left(1 - \hat{u}\right) + \frac{1}{4 * n^2} * z^2}\right) \text{ /. } n \to \text{SampleMax /. } \hat{u} \to \frac{\text{Result2}}{\text{SampleMax}}$$

$$\overline{z2} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2 * n} * z^2 + z * \sqrt{\frac{1}{n} * \hat{u} * \left(1 - \hat{u}\right) + \frac{1}{4 * n^2} * z^2}\right) \text{ /. } n \to \text{SampleMax /. } \hat{u} \to \frac{\text{Result2}}{\text{SampleMax}}$$

```
{0.00448966, 0.00931696, 0.0299587, 0.0526237, 0.0447477}
{0.0074899, 0.0134585, 0.0369997, 0.0617165, 0.0531988}
{0.0202399, 0.0431188, 0.0526237, 0.0269273, 0.011051}
{0.0261264, 0.0514291, 0.0617165, 0.0336336, 0.015523}
PlotCIunder1 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper1 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder1[[change, 1]] = Resultparameter1[[change]];
  PlotCIupper1[[change, 1]] = Resultparameter1[[change]];
```

$$\text{PlotCIunder1[[change, 2]]} = \frac{1}{s_{average}} * \underline{z1}\text{[[change]];}$$

$$\text{PlotCIupper1[[change, 2]]} = \frac{1}{s_{average}} * \overline{z1}\text{[[change]];}$$

```
];
```

$\overline{z1}$

$\frac{\text{Result1}}{\text{SampleMax}}$

$\underline{z1}$

```
gCIupper1 = ListPlot[PlotCIupper1, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.005], Red}, PlotMarkers → {"-"}];
gCIunder1 = ListPlot[PlotCIunder1, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.005], Red}, PlotMarkers → {"-"}];

PlotCI1♦1 = Table[0, {i, 2}, {j, 2}];
PlotCI2♦1 = Table[0, {i, 2}, {j, 2}];
PlotCI3♦1 = Table[0, {i, 2}, {j, 2}];
PlotCI4♦1 = Table[0, {i, 2}, {j, 2}];
PlotCI5♦1 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1♦1[[1, 1]] = Resultparameter1[[1]];
```

$$\text{PlotCI1♦1[[1, 2]]} = \frac{1}{s_{average}} * \underline{z1}\text{[[1]];}$$

```
  PlotCI1♦1[[2, 1]] = Resultparameter1[[1]];
```

$$\text{PlotCI1♦1[[2, 2]]} = \frac{1}{s_{average}} * \overline{z1}\text{[[1]];}$$

```
  PlotCI2♦1[[1, 1]] = Resultparameter1[[2]];
```

$$\text{PlotCI2♦1[[1, 2]]} = \frac{1}{s_{average}} * \underline{z1}\text{[[2]];}$$

```
  PlotCI2♦1[[2, 1]] = Resultparameter1[[2]];
```

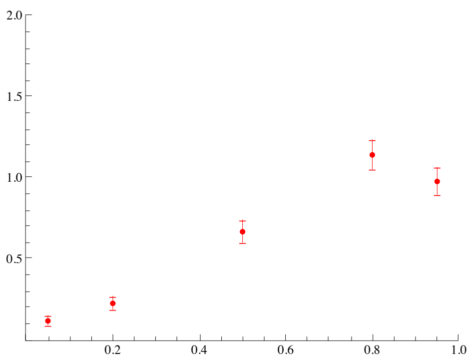
$$\text{PlotCI2♦1[[2, 2]]} = \frac{1}{s_{average}} * \overline{z1}\text{[[2]];}$$

```
  PlotCI3♦1[[1, 1]] = Resultparameter1[[3]];
  PlotCI3♦1[[1, 2]] = 1/s_average * z1[[3]];
  PlotCI3♦1[[2, 1]] = Resultparameter1[[3]];
  PlotCI3♦1[[2, 2]] = 1/s_average * z1‾[[3]];
  PlotCI4♦1[[1, 1]] = Resultparameter1[[4]];
  PlotCI4♦1[[1, 2]] = 1/s_average * z1[[4]];
  PlotCI4♦1[[2, 1]] = Resultparameter1[[4]];
  PlotCI4♦1[[2, 2]] = 1/s_average * z1‾[[4]];
  PlotCI5♦1[[1, 1]] = Resultparameter1[[5]];
  PlotCI5♦1[[1, 2]] = 1/s_average * z1[[5]];
  PlotCI5♦1[[2, 1]] = Resultparameter1[[5]];
  PlotCI5♦1[[2, 2]] = 1/s_average * z1‾[[5]];
 ];
gCI1♦1 = ListPlot[PlotCI1♦1, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Red}, Joined → True];
gCI2♦1 = ListPlot[PlotCI2♦1, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Red}, Joined → True];
gCI3♦1 = ListPlot[PlotCI3♦1, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Red}, Joined → True];
gCI4♦1 = ListPlot[PlotCI4♦1, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Red}, Joined → True];
gCI5♦1 = ListPlot[PlotCI5♦1, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Red}, Joined → True];
Show[gNumerical1, gCI1♦1, gCI2♦1, gCI3♦1, gCI4♦1, gCI5♦1, gCIunder1, gCIupper1]
```

{0.0074899, 0.0134585, 0.0369997, 0.0617165, 0.0531988}

{0.0058, 0.0112, 0.0333, 0.057, 0.0488}

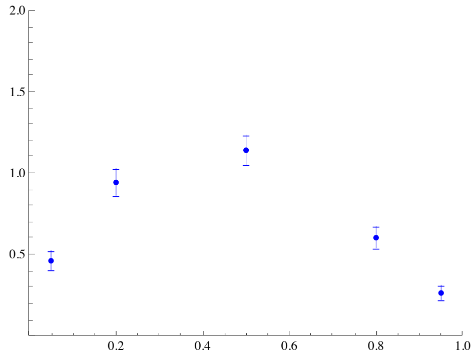
{0.00448966, 0.00931696, 0.0299587, 0.0526237, 0.0447477}




```
PlotCIunder2 = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper2 = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder2[[change, 1]] = Resultparameter2[[change]];
  PlotCIupper2[[change, 1]] = Resultparameter2[[change]];
  PlotCIunder2[[change, 2]] = 1/s_average * z2[[change]];
  PlotCIupper2[[change, 2]] = 1/s_average * z2‾[[change]];
 ];
```

```
z2 (overbar)
Result2
─────────
SampleMax
z2 (underbar)
gCIupper2 = ListPlot[PlotCIupper2, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.005], Blue}, PlotMarkers → {"-"}];
gCIunder2 = ListPlot[PlotCIunder2, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.005], Blue}, PlotMarkers → {"-"}];


PlotCI1♦2 = Table[0, {i, 2}, {j, 2}];
PlotCI2♦2 = Table[0, {i, 2}, {j, 2}];
PlotCI3♦2 = Table[0, {i, 2}, {j, 2}];
PlotCI4♦2 = Table[0, {i, 2}, {j, 2}];
PlotCI5♦2 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1♦2[[1, 1]] = Resultparameter2[[1]];
  PlotCI1♦2[[1, 2]] = 1/s_average * z2(underbar)[[1]];
  PlotCI1♦2[[2, 1]] = Resultparameter2[[1]];
  PlotCI1♦2[[2, 2]] = 1/s_average * z2(overbar)[[1]];
  PlotCI2♦2[[1, 1]] = Resultparameter2[[2]];
  PlotCI2♦2[[1, 2]] = 1/s_average * z2(underbar)[[2]];
  PlotCI2♦2[[2, 1]] = Resultparameter2[[2]];
  PlotCI2♦2[[2, 2]] = 1/s_average * z2(overbar)[[2]];
  PlotCI3♦2[[1, 1]] = Resultparameter2[[3]];
  PlotCI3♦2[[1, 2]] = 1/s_average * z2(underbar)[[3]];
  PlotCI3♦2[[2, 1]] = Resultparameter2[[3]];
  PlotCI3♦2[[2, 2]] = 1/s_average * z2(overbar)[[3]];
  PlotCI4♦2[[1, 1]] = Resultparameter2[[4]];
  PlotCI4♦2[[1, 2]] = 1/s_average * z2(underbar)[[4]];
  PlotCI4♦2[[2, 1]] = Resultparameter2[[4]];
  PlotCI4♦2[[2, 2]] = 1/s_average * z2(overbar)[[4]];
  PlotCI5♦2[[1, 1]] = Resultparameter2[[5]];
  PlotCI5♦2[[1, 2]] = 1/s_average * z2(underbar)[[5]];
  PlotCI5♦2[[2, 1]] = Resultparameter2[[5]];
  PlotCI5♦2[[2, 2]] = 1/s_average * z2(overbar)[[5]];
 ];
gCI1♦2 = ListPlot[PlotCI1♦2, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Blue}, Joined → True];
gCI2♦2 = ListPlot[PlotCI2♦2, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Blue}, Joined → True];
gCI3♦2 = ListPlot[PlotCI3♦2, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Blue}, Joined → True];
gCI4♦2 = ListPlot[PlotCI4♦2, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Blue}, Joined → True];
```

```
gCI5♦2 = ListPlot[PlotCI5♦2, PlotRange → {{MinV, MaxV}, {0, 2}},
   PlotStyle → {Thickness[0.001], Blue}, Joined → True];
Show[gNumerical2, gCI1♦2, gCI2♦2, gCI3♦2, gCI4♦2, gCI5♦2, gCIunder2, gCIupper2]
{0.0261264, 0.0514291, 0.0617165, 0.0336336, 0.015523}
{0.023, 0.0471, 0.057, 0.0301, 0.0131}
{0.0202399, 0.0431188, 0.0526237, 0.0269273, 0.011051}
```



Fixation probability (Moran-model)

$$\bar{d} = \frac{d_R + d_{RR}}{2};$$

$$\text{Ulinear} = \frac{8 * \hat{\rho}_H * \hat{\rho}_D}{\left(\hat{\rho}_H + 2 * \hat{\rho}_D\right) * \left(\frac{d_R}{\bar{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\bar{d}} * \hat{\rho}_D\right)} * s_{average} \;/.\; \hat{\rho}_D \to 1 - \hat{\rho}_H;$$

```
dR = 0.005;
dRR = 0.005;
```

$$\text{gUlinear1} = \text{Plot}\left[\frac{\text{Ulinear}}{s_{average}} \;/.\; d_R \to dR \;/.\; d_{RR} \to dRR, \{\hat{\rho}_H, 0, 1\}, \text{PlotRange} \to \{\{0, 1\}, \{0, 2\}\},\right.$$
$$\left.\text{PlotStyle} \to \{\text{Thickness}[0.005], \text{Black}, \text{Dashed}\}\right];$$

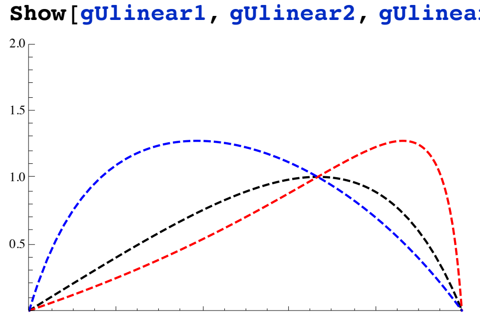
```
dR = 0.0005;
dRR = 0.005;
```

$$\text{gUlinear2} = \text{Plot}\left[\frac{\text{Ulinear}}{s_{average}} \;/.\; d_R \to dR \;/.\; d_{RR} \to dRR, \{\hat{\rho}_H, 0, 1\}, \text{PlotRange} \to \{\{0, 1\}, \{0, 2\}\},\right.$$
$$\left.\text{PlotStyle} \to \{\text{Thickness}[0.005], \text{Red}, \text{Dashed}\}\right];$$

```
dR = 0.005;
dRR = 0.0005;
```

$$\text{gUlinear3} = \text{Plot}\left[\frac{\text{Ulinear}}{s_{average}} \;/.\; d_R \to dR \;/.\; d_{RR} \to dRR, \{\hat{\rho}_H, 0, 1\}, \text{PlotRange} \to \{\{0, 1\}, \{0, 2\}\},\right.$$
$$\left.\text{PlotStyle} \to \{\text{Thickness}[0.005], \text{Blue}, \text{Dashed}\}, \text{AspectRatio} \to 0.75\right];$$

```
Show[gUlinear1, gUlinear2, gUlinear3]
```



Analytical result (Branching process)

```
fR = 0.5;
βR = y;
```

```
βRR = 1000;

dR = 0.0005;
dRR = 0.005;

SELfM = 0;
SELβM = s;
SELβMM = 2 * s;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$SEL\beta RM = \frac{\beta RM - \beta RR}{\beta RR};$$

$$SELdRM = \frac{dRR - dRM}{dRR};$$

$$Eq\rho H = \frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}};$$

$$Eq\rho D = \frac{\sqrt{\frac{fR*\beta R*dR}{2}}}{\sqrt{\frac{fR*\beta R*dR}{2}} + \sqrt{\beta RR * dRR}};$$

$$FaiR = \frac{\beta R}{dR};$$

$$FaiM = \frac{\beta M}{dM};$$

$$FaiRR = \frac{\beta RR}{dRR};$$

$$FaiRM = \frac{\beta RM}{dRM};$$

$$avef = \frac{fM + fR}{2};$$

$$FaiR2 = \frac{fR}{2} * FaiR;$$

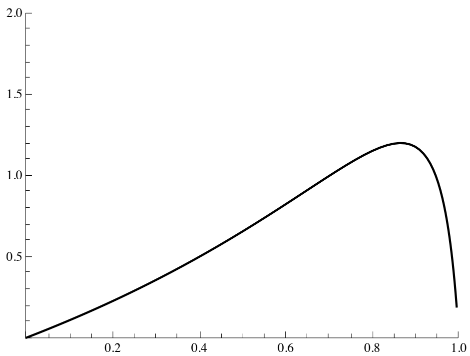
$$FaiM2 = \frac{avef}{2} * FaiM;$$

$$BPPaiH = \frac{FaiM2 * FaiRM - FaiR2 * FaiRR}{FaiM2 * FaiRM + \frac{FaiRM}{2} * \sqrt{FaiR2 * FaiRR}};$$

$$BPPaiD = \frac{FaiM2 * FaiRM - FaiR2 * FaiRR}{FaiM2 * FaiRM + 2 * FaiM2 * \sqrt{FaiR2 * FaiRR}};$$

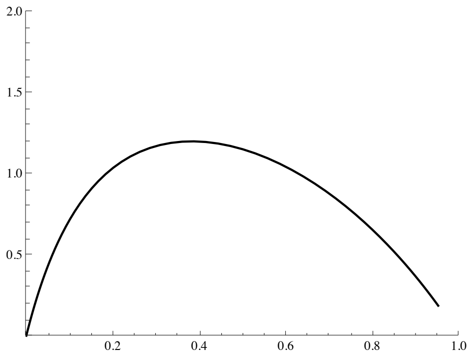
$$BPfix = \frac{Eq\rho H}{Eq\rho H + 2 * Eq\rho D} * BPPaiH + \frac{2 * Eq\rho D}{Eq\rho H + 2 * Eq\rho D} * BPPaiD;$$

```
MinV = 0;
MaxV = 10;
RepeatMax = 100;
PlotBP = Table[0, {i, RepeatMax}, {j, 2}];
For[repeat = 1, repeat ≤ RepeatMax - 1, repeat = repeat + 1,
  y1 = MinV + (MaxV - MinV)/RepeatMax * (repeat - 1);
  fracH = Sqrt[βRR * dRR] / (Sqrt[(fR*βR*dR)/2] + Sqrt[βRR * dRR]) /. y → 10^y1;
  U = BPfix /. y → 10^y1;
  PlotBP[[repeat, 1]] = fracH;
  PlotBP[[repeat, 2]] = U/s_average;
 ];
gBPfix1 = ListPlot[PlotBP, PlotRange → {{0, 1}, {0, 2}}, PlotStyle → {Thickness[0.005], Black},
  Joined → True, AspectRatio → 0.75]
```




```
fR = 0.5;
βR = y;
βRR = 1000;

dR = 0.005;
dRR = 0.0005;

SELfM = 0;
SELβM = s;
SELβMM = 2 * s;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);

SELβRM = (βRM - βRR)/βRR;
SELdRM = (dRR - dRM)/dRR;
```

```
EqρH = Sqrt[βRR * dRR] / (Sqrt[(fR*βR*dR)/2] + Sqrt[βRR * dRR]);

EqρD = Sqrt[(fR*βR*dR)/2] / (Sqrt[(fR*βR*dR)/2] + Sqrt[βRR * dRR]);

FaiR = βR/dR;
FaiM = βM/dM;
FaiRR = βRR/dRR;
FaiRM = βRM/dRM;
avef = (fM + fR)/2;
FaiR2 = fR/2 * FaiR;
FaiM2 = avef/2 * FaiM;
BPPaiH = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + FaiRM/2 * Sqrt[FaiR2 * FaiRR]);
BPPaiD = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + 2 * FaiM2 * Sqrt[FaiR2 * FaiRR]);
BPfix = EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD;

MinV = 0;
MaxV = 10;
RepeatMax = 100;
PlotBP = Table[0, {i, RepeatMax}, {j, 2}];
For[repeat = 1, repeat ≤ RepeatMax - 1, repeat = repeat + 1,
  y1 = MinV + (MaxV - MinV)/RepeatMax * (repeat - 1);

  fracH = Sqrt[βRR * dRR] / (Sqrt[(fR*βR*dR)/2] + Sqrt[βRR * dRR]) /. y → 10^y1;

  U = BPfix /. y → 10^y1;
  PlotBP[[repeat, 1]] = fracH;
  PlotBP[[repeat, 2]] = U / s_average;
 ];
gBPfix2 = ListPlot[PlotBP, PlotRange → {{0, 1}, {0, 2}}, PlotStyle → {Thickness[0.005], Black},
  Joined → True, AspectRatio → 0.75]
```



Analytical result (Diffusion)

```
dR = 0.0005;
dRR = 0.005;

Parameters = {SUM → Site, d_R → dR, d_RR → dRR, ρ̂_D → 1 - ρ̂_H, sfM → SELfM, sβM → SELβM,
    sβRM → SELβRM, sβMM → SELβMM, sdM → SELdM, sdRM → SELdRM, sdMM → SELdMM };
```

$$\bar{d} = \frac{d_R + d_{RR}}{2} \text{ /. Parameters;}$$

$$s_e = 2 * s_{average};$$

$$N_e = \frac{2 * \hat{\rho}_H * \hat{\rho}_D}{\frac{d_R}{\bar{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\bar{d}} * \hat{\rho}_D} * SUM;$$

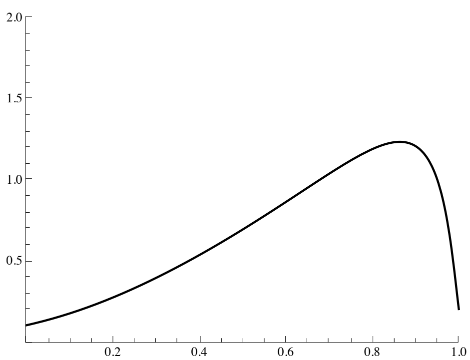
$$INI = \frac{1}{SUM * (\hat{\rho}_H + 2 * \hat{\rho}_D)} \text{ /. Parameters;}$$

```
EffectiveS = s_e /. Parameters;
EffectiveN = N_e /. Parameters;
```

$$DFfixWS1 = \frac{1 - Exp[-2 * EffectiveS * EffectiveN * INI]}{1 - Exp[-2 * EffectiveS * EffectiveN]};$$

$$gDFfix1 = Plot\left[\frac{DFfixWS1}{s_{average}}, \{\hat{\rho}_H, 0, 1\}, PlotRange \to \{\{0, 1\}, \{0, 2\}\},\right.$$
$$\left. PlotStyle \to \{Thickness[0.005], Black\}, AspectRatio \to 0.75\right];$$

```
Show[gDFfix1]
```



```
dR = 0.005;
dRR = 0.0005;

Parameters = {SUM → Site, d_R → dR, d_RR → dRR, ρ̂_D → 1 - ρ̂_H, sfM → SELfM, sβM → SELβM,
   sβRM → SELβRM, sβMM → SELβMM, sdM → SELdM, sdRM → SELdRM, sdMM → SELdMM};
```

$$\bar{d} = \frac{d_R + d_{RR}}{2} \;/.\; \text{Parameters};$$

$$s_e = 2 * s_{average};$$

$$N_e = \frac{2 * \hat{\rho}_H * \hat{\rho}_D}{\frac{d_R}{\bar{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\bar{d}} * \hat{\rho}_D} * \text{SUM};$$

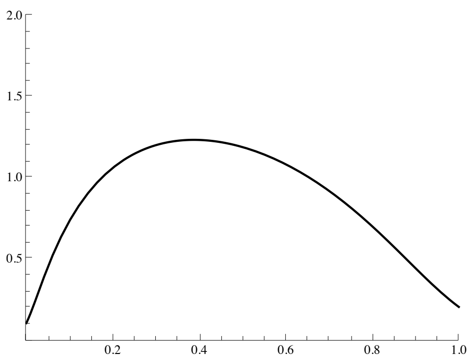
$$\text{INI} = \frac{1}{\text{SUM} * \left(\hat{\rho}_H + 2 * \hat{\rho}_D\right)} \;/.\; \text{Parameters};$$

```
EffectiveS = s_e /. Parameters;
EffectiveN = N_e /. Parameters;
```

$$\text{DFfixWS2} = \frac{1 - \text{Exp}[-2 * \text{EffectiveS} * \text{EffectiveN} * \text{INI}]}{1 - \text{Exp}[-2 * \text{EffectiveS} * \text{EffectiveN}]};$$

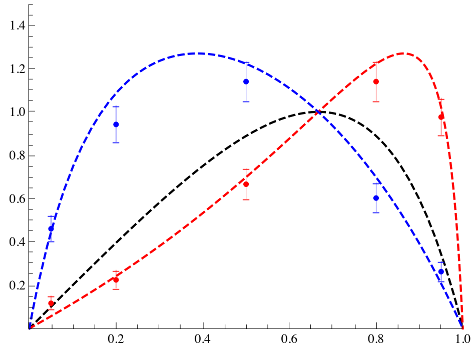
$$\text{gDFfix2} = \text{Plot}\left[\frac{\text{DFfixWS2}}{s_{average}}, \{\hat{\rho}_H, 0, 1\}, \text{PlotRange} \to \{\{0, 1\}, \{0, 2\}\},\right.$$
$$\left.\text{PlotStyle} \to \{\text{Thickness}[0.005], \text{Black}\}, \text{AspectRatio} \to 0.75\right];$$

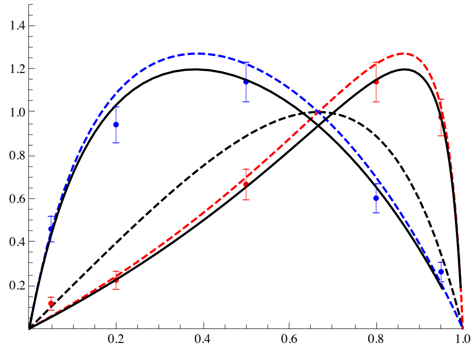
```
Show[gDFfix2]
```

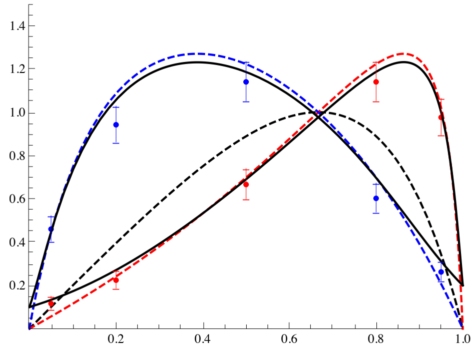


Plot

```
Show[gUlinear1, gUlinear2, gUlinear3, gNumerical1, gNumerical2, gCI1♦1, gCI2♦1,
 gCI3♦1, gCI4♦1, gCI5♦1, gCIunder1, gCIupper1, gCI1♦2, gCI2♦2, gCI3♦2, gCI4♦2,
 gCI5♦2, gCIunder2, gCIupper2, PlotRange → {{0, 1}, {0, 1.5}}, AspectRatio → 0.75]
Show[gUlinear1, gUlinear2, gUlinear3, gNumerical1, gNumerical2, gBPfix1, gBPfix2,
 gCI1♦1, gCI2♦1, gCI3♦1, gCI4♦1, gCI5♦1, gCIunder1, gCIupper1, gCI1♦2, gCI2♦2,
 gCI3♦2, gCI4♦2, gCI5♦2, gCIunder2, gCIupper2, PlotRange → {{0, 1}, {0, 1.5}},
 AspectRatio → 0.75]
Show[gUlinear1, gUlinear2, gUlinear3, gNumerical1, gNumerical2, gDFfix1, gDFfix2,
 gCI1♦1, gCI2♦1, gCI3♦1, gCI4♦1, gCI5♦1, gCIunder1, gCIupper1, gCI1♦2, gCI2♦2,
 gCI3♦2, gCI4♦2, gCI5♦2, gCIunder2, gCIupper2, PlotRange → {{0, 1}, {0, 1.5}},
 AspectRatio → 0.75]
Show[gUlinear1, gUlinear2, gUlinear3, gNumerical1, gNumerical2, gBPfix1, gBPfix2,
 gDFfix1, gDFfix2, gCI1♦1, gCI2♦1, gCI3♦1, gCI4♦1, gCI5♦1, gCIunder1, gCIupper1,
 gCI1♦2, gCI2♦2, gCI3♦2, gCI4♦2, gCI5♦2, gCIunder2, gCIupper2, PlotRange → {{0, 1}, {0, 1.5}},
 AspectRatio → 0.75]
```







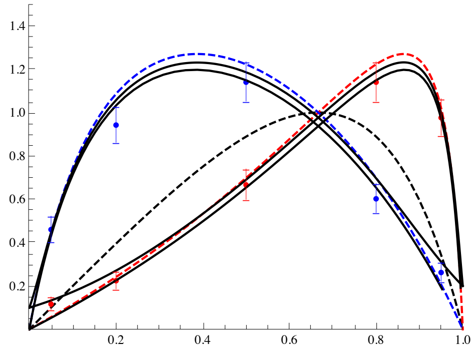
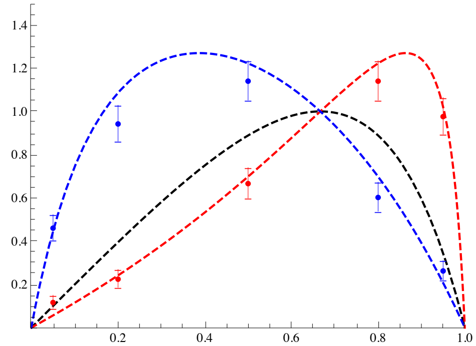


Figure for output

```
Show[gUlinear1, gUlinear2, gUlinear3, gNumerical1, gNumerical2, gCI1♦1, gCI2♦1,
 gCI3♦1, gCI4♦1, gCI5♦1, gCIunder1, gCIupper1, gCI1♦2, gCI2♦2, gCI3♦2, gCI4♦2,
 gCI5♦2, gCIunder2, gCIupper2, PlotRange → {{0, 1}, {0, 1.5}}, AspectRatio → 0.75]
```



- **Fig.5(b)**

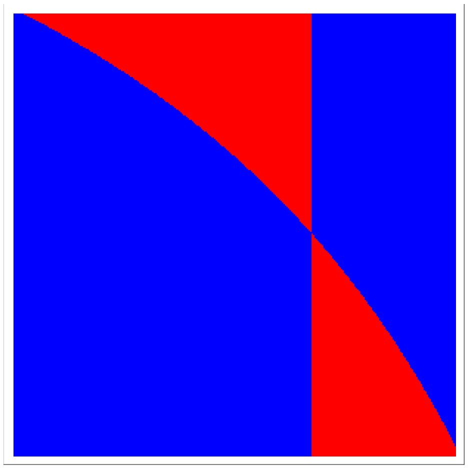
Because of , $\frac{dR}{d} = 2 * \frac{dR}{dR+dRR}$, Eq. (13b) becomes $\frac{8*\hat{\rho}_H*\hat{\rho}_D}{(\hat{\rho}_H+2*\hat{\rho}_D)*\left(2*\frac{dR}{dR+dRR}*\hat{\rho}_H+2*2*\frac{dR}{dR+dRR}*\hat{\rho}_D\right)} = \frac{4*\hat{\rho}_H*\hat{\rho}_D}{(\hat{\rho}_H+2*\hat{\rho}_D)*\left(\frac{dR}{dR+dRR}*\hat{\rho}_H+2*\frac{dR}{dR+dRR}*\hat{\rho}_D\right)}$.

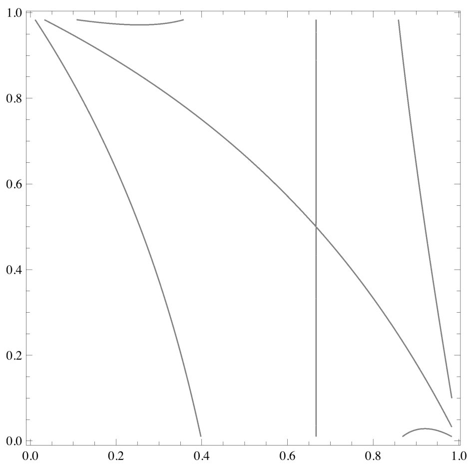
```
MaxV = 0.99;
MinV = 0.01;
RepeatMax = 301;
```

$$\text{Ulinear} = \frac{4 * \hat{\rho}_H * \hat{\rho}_D}{(\hat{\rho}_H + 2 * \hat{\rho}_D) * (R_H * \hat{\rho}_H + 2 * R_D * \hat{\rho}_D)} * s_{average} * \frac{1}{s_{average}} \text{ /. } \hat{\rho}_D \to 1 - \hat{\rho}_H \text{ /. } R_D \to 1 - R_H;$$

```
Omega1 = Table[10, {(RepeatMax - 1) * (RepeatMax - 1)}];
Omega2 = Table[10, {i, (RepeatMax - 1) * (RepeatMax - 1)}, {j, 3}];
repeat = 0;
For[repeat1 = 1, repeat1 ≤ RepeatMax - 1, repeat1 = repeat1 + 1,
  For[repeat2 = 1, repeat2 ≤ RepeatMax - 1, repeat2 = repeat2 + 1,
    repeat = repeat + 1;
    x1 = MinV + (MaxV - MinV)/RepeatMax * (repeat1 - 1);
    x2 = MinV + (MaxV - MinV)/RepeatMax * (repeat2 - 1);
    U = Ulinear /. ρ̂H → x1 /. RH → x2;

    index = -1;
    If[0 < U && U < 1.0,
     index = 1;
    ];
    If[1.0 < U,
     index = 2;
    ];

    Omega1[[repeat]] = index;
    Omega2[[repeat, 1]] = x1;
    Omega2[[repeat, 2]] = x2;
    Omega2[[repeat, 3]] = U;
   ];
 ];
```

```
(*Plot*)
PlotOmega1 = Reverse[Transpose[Partition[Omega1, {(RepeatMax - 1)}]]];
ArrayPlot[PlotOmega1, AspectRatio → 1, ColorRules → {1 → Blue, 2 → Red}]
ListContourPlot[Omega2, Contours → {0.5, 1.0, 1.5, 2.0}, ContourShading → None]
```





- **Fig.6(a)**

Parameter values

```
Clear[x]

Site = 100;
fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 0.2 + x;
SELβMM = -2 * x;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0.5;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\text{SEL}\beta\text{RM} = \frac{\beta\text{RM} - \beta\text{RR}}{\beta\text{RR}};$$

$$\text{SELdRM} = \frac{\text{dRR} - \text{dRM}}{\text{dRR}};$$

```
MinV = -0.51;
MaxV = 0.21;
```

$$s_{average} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SEL}\beta\text{M} + \text{SELdM}\right) + \frac{1}{2} * (\text{SEL}\beta\text{MM} * \text{HETERO}\beta + \text{SELdM} * \text{HETEROd})$$

$$\frac{1}{2}(0. - 1.\,x) + \frac{1}{2}(0.2 + x)$$

Out put

```
SampleMax = 10 000;
ChangeMax = 7;
OutputData = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulation

```
VectpM = Table[0, {SampleMax}];

Resultparameter = Table[0, {ChangeMax + 1}];
Result = Table[0, {ChangeMax + 1}];

change = 1;
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,

  If[change == 1, x = -0.5];
  If[change == 2, x = -0.4];
  If[change == 3, x = -0.3];
  If[change == 4, x = -0.2];
  If[change == 5, x = -0.1];
  If[change == 6, x = 0];
```

```
If[change == 7, x = 0.1];
If[change == 8, x = 0.2];

sample = 1;
For[sample = 1, sample ≤ SampleMax, sample = sample + 1,


  EqρH = Sqrt[βRR * dRR] / (Sqrt[(fR*βR*dR)/2] + Sqrt[βRR * dRR]);
  xR = Round[Site * EqρH];
  xM = 0;
  xRR = Site - xR;
  xRM = 0;
  xMM = 0;
  sampleMutation = RandomVariate[MultinomialDistribution[1, {xR/(xR + 2 * xRR), (2 * xRR)/(xR + 2 * xRR)}]];
  xR = xR - sampleMutation[[1]];
  xM = xM + sampleMutation[[1]];
  xRR = xRR - sampleMutation[[2]];
  xRM = xRM + sampleMutation[[2]];
  xMM = 0;


  If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
  If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
  If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];


  step = 2;
  While[0 < pM && pM < 1,
   pDeathR = (dR * xR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   pDeathM = (dM * xM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   pDeathRR = (dRR * xRR)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   pDeathRM = (dRM * xRM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   pDeathMM = (dMM * xMM)/(dR * xR + dM * xM + dRR * xRR + dRM * xRM + dMM * xMM);
   sampleDeath =
    RandomVariate[MultinomialDistribution[1,
      {pDeathR, pDeathM, pDeathRR, pDeathRM, pDeathMM}]];
   xR = xR - sampleDeath[[1]];
   xM = xM - sampleDeath[[2]];
   xRR = xRR - sampleDeath[[3]];
   xRM = xRM - sampleDeath[[4]];
   xMM = xMM - sampleDeath[[5]];

   cR = βRR * xRR + βRM/2 * xRM;
   cM = βMM * xMM + βRM/2 * xRM;
```

```
If[xR == 0 && xM == 0, cRR = 0];
If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (βR^2 * xR^2)/(βR * xR + βM * xM)];
If[xR == 0 && xM == 0, cRM = 0];
If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (βR * βM * xR * xM)/(βR * xR + βM * xM)];
If[xR == 0 && xM == 0, cMM = 0];
If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (βM^2 * xM^2)/(βR * xR + βM * xM)];
pCellR = cR/(cR + cM + cRR + cRM + cMM);
pCellM = cM/(cR + cM + cRR + cRM + cMM);
pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
sampleBirth =
 RandomVariate[MultinomialDistribution[1,
   {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

xR = xR + sampleBirth[[1]];
xM = xM + sampleBirth[[2]];
xRR = xRR + sampleBirth[[3]];
xRM = xRM + sampleBirth[[4]];
xMM = xMM + sampleBirth[[5]];

If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = dRR/(dR + dRR) * (xRM/2 + xMM)/(xRR + xRM + xMM) + dR/(dR + dRR) * xM/(xR + xM)];

step = step + 1;
];

VectpM[[sample]] = pM;

OutputData[[countData, 1]] = x;
OutputData[[countData, 2]] = pM;
countData = countData + 1;

];

Resultparameter[[change]] = x;
Result[[change]] = Total[VectpM];

OutputResult[[countResult, 1]] = x;
OutputResult[[countResult, 2]] = Total[VectpM];
countResult = countResult + 1;

Print[change]

].
```

1

2

3

4

5

6

7

8

Output of the data

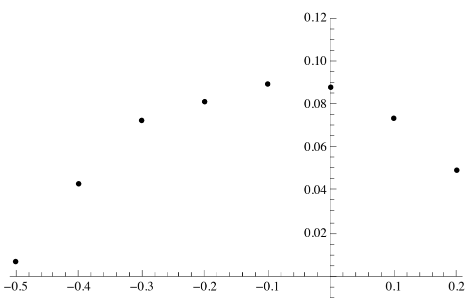
```
Resultparameter
Result
Export["/Users/Bessho/Desktop/workstation/OutputDataFig6b.txt", OutputData, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig6b.txt", OutputResult, "Table"];
```

```
{-0.5, -0.4, -0.3, -0.2, -0.1, 0, 0.1, 0.2}
{67., 428., 722., 809., 891., 876., 732., 491.}
```

```
Resultparameter = {-0.5`, -0.4`, -0.3`, -0.2`, -0.1`, 0, 0.1`, 0.2`};
Result = {67.`, 428.`, 722.`, 809.`, 891.`, 876.`, 732.`, 491.`};
```

Plot

```
PlotpM = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM[[change, 1]] = Resultparameter[[change]];
  PlotpM[[change, 2]] = Result[[change]] / SampleMax;
  PlotpMlog[[change, 1]] = Resultparameter[[change]];
  PlotpMlog[[change, 2]] = Log[10, Result[[change]] / SampleMax];
 ];
gNumerical = ListPlot[PlotpM, PlotRange → {{MinV, MaxV}, {-0.01, 0.12}},
  PlotStyle → {PointSize[0.012], Black}]
```



Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \overline{z}] \sim 0.95$.

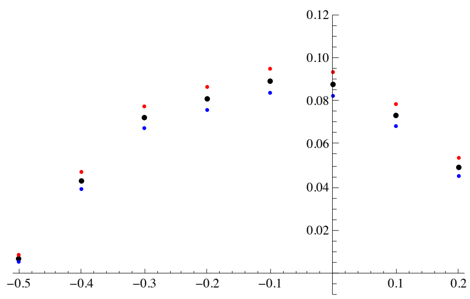
```
z = 1.96;
```

$$\underline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 - z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /.\ n \to SampleMax /.\ \hat{u} \to \frac{Result}{SampleMax}$$

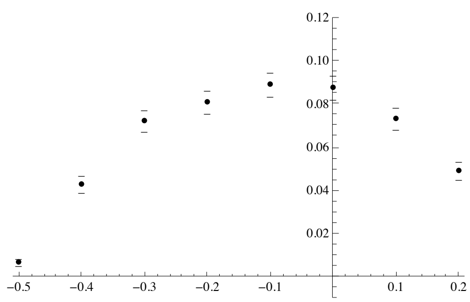
$$\overline{z} = \frac{n}{n+z^2}\left(\hat{u} + \frac{1}{2*n}*z^2 + z*\sqrt{\frac{1}{n}*\hat{u}*(1-\hat{u}) + \frac{1}{4*n^2}*z^2}\right) /.\ n \to SampleMax /.\ \hat{u} \to \frac{Result}{SampleMax}$$

```
{0.00527961, 0.0390053, 0.0672897, 0.075715, 0.0836728, 0.082216, 0.0682571, 0.0450353}
{0.00849926, 0.0469458, 0.0774388, 0.0864069, 0.0948428, 0.0933007, 0.0784707, 0.053511}
```

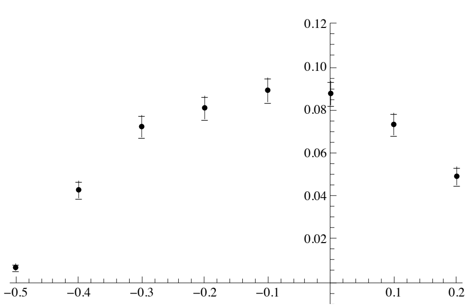
```
PlotCIunder = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder[[change, 1]] = Resultparameter[[change]];
  PlotCIupper[[change, 1]] = Resultparameter[[change]];
  PlotCIunder[[change, 2]] = z[[change]];
  PlotCIupper[[change, 2]] = z̄[[change]];
 ];
gCIupperDot = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Red}];
gCIunderDot = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Blue}];
Show[gNumerical, gCIunderDot, gCIupperDot]
```




```
gCIupper = ListPlot[PlotCIupper, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"—"}];
gCIunder = ListPlot[PlotCIunder, PlotRange → {{MinV, MaxV}, {-5, 0}},
   PlotStyle → {Thickness[0.005], Black}, PlotMarkers → {"—"}];
Show[gNumerical, gCIunder, gCIupper]
```



```
PlotCI1 = Table[0, {i, 2}, {j, 2}];
PlotCI2 = Table[0, {i, 2}, {j, 2}];
PlotCI3 = Table[0, {i, 2}, {j, 2}];
PlotCI4 = Table[0, {i, 2}, {j, 2}];
PlotCI5 = Table[0, {i, 2}, {j, 2}];
PlotCI6 = Table[0, {i, 2}, {j, 2}];
PlotCI7 = Table[0, {i, 2}, {j, 2}];
PlotCI8 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1[[1, 1]] = Resultparameter[[1]];
  PlotCI1[[1, 2]] = z̲[[1]];
  PlotCI1[[2, 1]] = Resultparameter[[1]];
  PlotCI1[[2, 2]] = z̅[[1]];
  PlotCI2[[1, 1]] = Resultparameter[[2]];
  PlotCI2[[1, 2]] = z̲[[2]];
  PlotCI2[[2, 1]] = Resultparameter[[2]];
  PlotCI2[[2, 2]] = z̅[[2]];
  PlotCI3[[1, 1]] = Resultparameter[[3]];
  PlotCI3[[1, 2]] = z̲[[3]];
  PlotCI3[[2, 1]] = Resultparameter[[3]];
  PlotCI3[[2, 2]] = z̅[[3]];
  PlotCI4[[1, 1]] = Resultparameter[[4]];
  PlotCI4[[1, 2]] = z̲[[4]];
  PlotCI4[[2, 1]] = Resultparameter[[4]];
  PlotCI4[[2, 2]] = z̅[[4]];
  PlotCI5[[1, 1]] = Resultparameter[[5]];
  PlotCI5[[1, 2]] = z̲[[5]];
  PlotCI5[[2, 1]] = Resultparameter[[5]];
  PlotCI5[[2, 2]] = z̅[[5]];
  PlotCI6[[1, 1]] = Resultparameter[[6]];
  PlotCI6[[1, 2]] = z̲[[6]];
  PlotCI6[[2, 1]] = Resultparameter[[6]];
  PlotCI6[[2, 2]] = z̅[[6]];
  PlotCI7[[1, 1]] = Resultparameter[[7]];
  PlotCI7[[1, 2]] = z̲[[7]];
  PlotCI7[[2, 1]] = Resultparameter[[7]];
  PlotCI7[[2, 2]] = z̅[[7]];
  PlotCI8[[1, 1]] = Resultparameter[[8]];
  PlotCI8[[1, 2]] = z̲[[8]];
  PlotCI8[[2, 1]] = Resultparameter[[8]];
  PlotCI8[[2, 2]] = z̅[[8]];
 ];
gCI1 = ListPlot[PlotCI1, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2 = ListPlot[PlotCI2, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3 = ListPlot[PlotCI3, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4 = ListPlot[PlotCI4, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5 = ListPlot[PlotCI5, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI6 = ListPlot[PlotCI6, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI7 = ListPlot[PlotCI7, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI8 = ListPlot[PlotCI8, PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6, gCI7, gCI8, gCIunder, gCIupper]
```



Analytical result (Branching process approximation)

```
Clear[y]

fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 0.2 + y;
SELβMM = -2 * y;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$\mathrm{SEL\beta RM} = \frac{\mathrm{\beta RM} - \mathrm{\beta RR}}{\mathrm{\beta RR}};$$

$$\mathrm{SELdRM} = \frac{\mathrm{dRR} - \mathrm{dRM}}{\mathrm{dRR}};$$

$$s_{average} = \frac{1}{2} * \left(\frac{\mathrm{SELfM}}{2} + \mathrm{SEL\beta M} + \mathrm{SELdM}\right) + \frac{1}{2} * (\mathrm{SEL\beta MM} * \mathrm{HETERO\beta} + \mathrm{SELdM} * \mathrm{HETEROd})$$

$$\mathrm{Eq\rho H} = \frac{\sqrt{\mathrm{\beta RR} * \mathrm{dRR}}}{\sqrt{\frac{\mathrm{fR} * \mathrm{\beta R} * \mathrm{dR}}{2}} + \sqrt{\mathrm{\beta RR} * \mathrm{dRR}}};$$

$$\mathrm{Eq\rho D} = \frac{\sqrt{\frac{\mathrm{fR} * \mathrm{\beta R} * \mathrm{dR}}{2}}}{\sqrt{\frac{\mathrm{fR} * \mathrm{\beta R} * \mathrm{dR}}{2}} + \sqrt{\mathrm{\beta RR} * \mathrm{dRR}}};$$

$$\mathrm{FaiR} = \frac{\mathrm{\beta R}}{\mathrm{dR}};$$

$$\mathrm{FaiM} = \frac{\mathrm{\beta M}}{\mathrm{dM}};$$

$$\mathrm{FaiRR} = \frac{\mathrm{\beta RR}}{\mathrm{dRR}};$$

$$\mathrm{FaiRM} = \frac{\mathrm{\beta RM}}{\mathrm{dRM}};$$

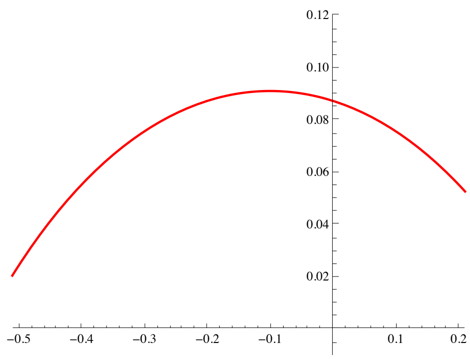
$$\mathrm{avef} = \frac{\mathrm{fM} + \mathrm{fR}}{2};$$

$$\mathrm{FaiR2} = \frac{\mathrm{fR}}{2} * \mathrm{FaiR};$$

$$\mathrm{FaiM2} = \frac{\mathrm{avef}}{2} * \mathrm{FaiM};$$

```
BPPaiH = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + FaiRM/2 * Sqrt[FaiR2 * FaiRR]);
BPPaiD = (FaiM2 * FaiRM - FaiR2 * FaiRR) / (FaiM2 * FaiRM + 2 * FaiM2 * Sqrt[FaiR2 * FaiRR]);
BPfix = EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD;
gBPfix = Plot[BPfix, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.12}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
```

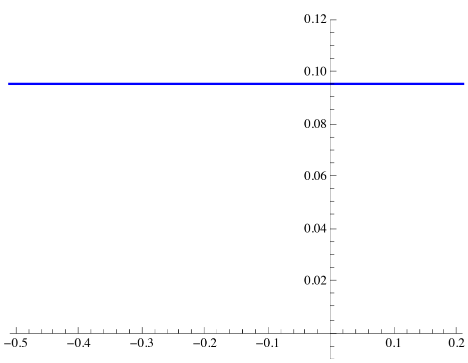
$$-0.5\,y + \frac{1}{2}\,(0.2 + y)$$



Analytical result (Diffusion approximation)

```
Parameters = {SUM → Site, d_R → dR, d_RR → dRR, ρ̂_H → Sqrt[βRR * dRR] / (Sqrt[(fR*βR*dR)/2] + Sqrt[βRR * dRR]),
  ρ̂_D → Sqrt[(fR*βR*dR)/2] / (Sqrt[(fR*βR*dR)/2] + Sqrt[βRR * dRR]), sfM → SELfM, sβM → SELβM, sβRM → SELβRM, sβMM → SELβMM,
  sdM → SELdM, sdRM → SELdRM, sdMM → SELdMM}
Additive = sβMM - 2 * sβRM + sdMM - 2 * sdRM;
d_harmonic = 2 / (1/d_R + 1/d_RR);
d̂ = d_R * ρ̂_H + d_RR * (1 - ρ̂_H);
d̄ = (d_R + d_RR) / 2;

m = SUM * p * (1 - p) * (2 * s_average + p * Additive) / 2;
v = p * (1 - p) * (d_R/d̄ * ρ̂_H + 2 * d_RR/d̄ * ρ̂_D) / (4 * ρ̂_H * ρ̂_D);
Q = Integrate[m/v /. Parameters, p];
INI = 1 / (SUM * (ρ̂_H + 2 * ρ̂_D)) /. Parameters;
DFfixWS = Integrate[Exp[-2 * Q], {p, 0, INI}] / Integrate[Exp[-2 * Q], {p, 0, 1}];
gDFfixWS = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.12}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

$\{SUM \to 100,\ d_R \to 0.005,\ d_{RR} \to 0.005,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,\ sfM \to 0,$
$s\beta M \to 0.2 + y,\ s\beta RM \to \frac{-1000 + 1000\ (1 - 1.\ y)}{1000},\ s\beta MM \to -2\ y,\ sdM \to 0,\ sdRM \to 0.,\ sdMM \to 0\}$



Analytical result (Lineary approximation)

$$\text{FixLinear} = \frac{8 * \hat{\rho}_H * \hat{\rho}_D}{\left(\hat{\rho}_H + 2 * \hat{\rho}_D\right) * \left(\frac{d_R}{\bar{d}} * \hat{\rho}_H + 2 * \frac{d_{RR}}{\bar{d}} * \hat{\rho}_D\right)} * s_{average};$$

```
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
  PlotRange → {{MinV, MaxV}, {0, 0.12}}, PlotStyle → {Thickness[0.005], Black, Dashed},
  AspectRatio → 0.75]
```

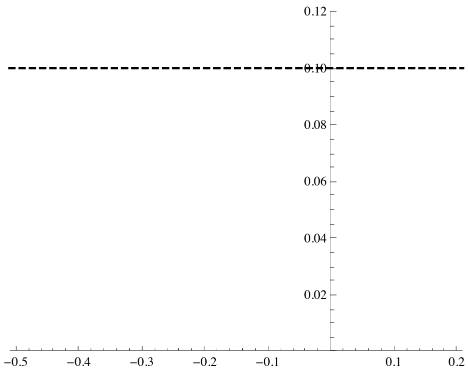
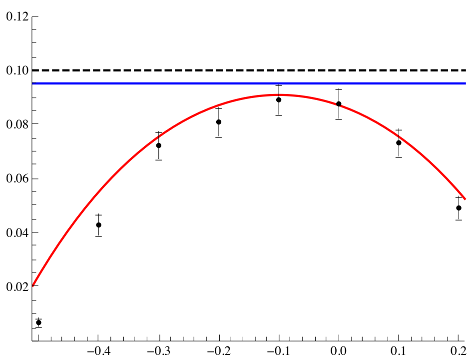


Figure for output

```
Show[gDFfixWS, gBPfix, gFixLinear, gNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6,
 gCI7, gCI8, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {0, 0.12}},
 AxesOrigin → {-0.51, 0}]
```




```
Export["~/Desktop/Fig6a.PDF", %, "PDF"];
```

$$\frac{\sqrt{\beta RR * dRR}}{\sqrt{\frac{fR * \beta R * dR}{2}} + \sqrt{\beta RR * dRR}}$$

0.666667

As ploidally antagonistic selection becomes stronger, however, the M allele stops being beneficial when common. We can measure selection on M as the difference in fitness of M-bearing versus R-bearing individuals relative to the common genotypes (MM and M):

```
Clear[z]

fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 0.2 + z;
SELβMM = -2 * z;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0.5;

fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
dMM = dRR * (1 - SELdMM);
```

$$SEL\beta RM = \frac{\beta RM - \beta RR}{\beta RR};$$

$$SELdRM = \frac{dRR - dRM}{dRR};$$

$$s_{average} = \frac{1}{2} * \left(\frac{SELfM}{2} + SEL\beta M + SELdM\right) + \frac{1}{2} * (SEL\beta MM * HETERO\beta + SELdM * HETEROd) \text{ // Simplify}$$

```
0.1
```

We next derive the selection coefficient of allele R, reversing the effects of allele M and R.
To consider this fixation probability, we now define the rare allele as m (actually allele R) and the resident allele as r (actually allele M).

```
fr = fM;
fm = fR;

βr = βM;
βm = βR;

βrr = βMM;
βrm = βRM;
βmm = βRR;

dr = dM;
dm = dR;

drr = dMM;
drm = dRM;
dmm = dRR;
```

$$\text{SELfm} = \frac{\text{fm} - \text{fr}}{\text{fr}};$$

$$\text{SEL}\beta\text{m} = \frac{\beta\text{m} - \beta\text{r}}{\beta\text{r}};$$

$$\text{SELdm} = \frac{\text{dr} - \text{dm}}{\text{dr}};$$

$$\text{SEL}\beta\text{rm} = \frac{\beta\text{rm} - \beta\text{rr}}{\beta\text{rr}};$$

$$\text{SELdrm} = \frac{\text{drr} - \text{drm}}{\text{drr}};$$

$$\text{aveselMcommon} = \frac{1}{2} * \left(\frac{\text{SELfm}}{2} + \text{SEL}\beta\text{m} + \text{SELdm}\right) + \frac{1}{2} * (\text{SEL}\beta\text{rm} + \text{SELdrm}) \text{ // Simplify}$$

$$0. + \frac{0.5\,z}{1 - 2\,z} - \frac{0.1}{1.2 + z} - \frac{0.5\,z}{1.2 + z}$$

```
NSolve[aveselMcommon == 0, z]
{{z → -0.376887}, {z → 0.176887}}
```

This is when M stops being selected when common.

We next ask when R can actually invade a population fixed on M, according to the Moran model, so that both M and R are predicted to invade when rare.

We first plot the fixation probability of allele M when allele M is rare.

```
fR = 0.5;
βR = 1000;
βRR = 1000;
dR = 0.005;
dRR = 0.005;

SELfM = 0;
SELβM = 0.2 + z;
SELβMM = -2 * z;
HETEROβ = 0.5;
SELdM = 0;
SELdMM = 0;
HETEROd = 0.5;


fM = fR * (1 + SELfM);
βM = βR * (1 + SELβM);
βRM = βRR * (1 + HETEROβ * SELβMM);
βMM = βRR * (1 + SELβMM);
dM = dR * (1 - SELdM);
dRM = dRR * (1 - HETEROd * SELdMM);
```

$$\mathrm{dMM} = \mathrm{dRR} * (1 - \mathrm{SELdMM});$$

$$\mathrm{SEL\beta RM} = \frac{\beta\mathrm{RM} - \beta\mathrm{RR}}{\beta\mathrm{RR}};$$

$$\mathrm{SELdRM} = \frac{\mathrm{dRR} - \mathrm{dRM}}{\mathrm{dRR}};$$

$$\mathrm{Eq\rho H} = \frac{\sqrt{\beta\mathrm{RR} * \mathrm{dRR}}}{\sqrt{\frac{\mathrm{fR} * \beta\mathrm{R} * \mathrm{dR}}{2}} + \sqrt{\beta\mathrm{RR} * \mathrm{dRR}}};$$

$$\mathrm{Eq\rho D} = \frac{\sqrt{\frac{\mathrm{fR} * \beta\mathrm{R} * \mathrm{dR}}{2}}}{\sqrt{\frac{\mathrm{fR} * \beta\mathrm{R} * \mathrm{dR}}{2}} + \sqrt{\beta\mathrm{RR} * \mathrm{dRR}}};$$

$$\mathrm{FaiR} = \frac{\beta\mathrm{R}}{\mathrm{dR}};$$

$$\mathrm{FaiM} = \frac{\beta\mathrm{M}}{\mathrm{dM}};$$

$$\mathrm{FaiRR} = \frac{\beta\mathrm{RR}}{\mathrm{dRR}};$$

$$\mathrm{FaiRM} = \frac{\beta\mathrm{RM}}{\mathrm{dRM}};$$

$$\mathrm{avef} = \frac{\mathrm{fM} + \mathrm{fR}}{2};$$

$$\mathrm{FaiR2} = \frac{\mathrm{fR}}{2} * \mathrm{FaiR};$$

$$\mathrm{FaiM2} = \frac{\mathrm{avef}}{2} * \mathrm{FaiM};$$

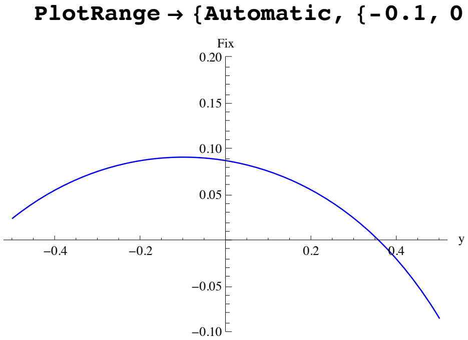
$$\mathrm{BPPaiH} = \frac{\mathrm{FaiM2} * \mathrm{FaiRM} - \mathrm{FaiR2} * \mathrm{FaiRR}}{\mathrm{FaiM2} * \mathrm{FaiRM} + \frac{\mathrm{FaiRM}}{2} * \sqrt{\mathrm{FaiR2} * \mathrm{FaiRR}}};$$

$$\mathrm{BPPaiD} = \frac{\mathrm{FaiM2} * \mathrm{FaiRM} - \mathrm{FaiR2} * \mathrm{FaiRR}}{\mathrm{FaiM2} * \mathrm{FaiRM} + 2 * \mathrm{FaiM2} * \sqrt{\mathrm{FaiR2} * \mathrm{FaiRR}}};$$

$$\mathrm{BPfixM} = \frac{\mathrm{Eq\rho H}}{\mathrm{Eq\rho H} + 2 * \mathrm{Eq\rho D}} * \mathrm{BPPaiH} + \frac{2 * \mathrm{Eq\rho D}}{\mathrm{Eq\rho H} + 2 * \mathrm{Eq\rho D}} * \mathrm{BPPaiD}$$

$$\frac{0.5 \left(-1. \times 10^{10} + 1. \times 10^{10} \ (1 - 1.\ z) \ (1.2 + z)\right)}{1. \times 10^{10} \ (1 - 1.\ z) + 1. \times 10^{10} \ (1 - 1.\ z) \ (1.2 + z)} + \frac{0.5 \left(-1. \times 10^{10} + 1. \times 10^{10} \ (1 - 1.\ z) \ (1.2 + z)\right)}{1. \times 10^{10} \ (1.2 + z) + 1. \times 10^{10} \ (1 - 1.\ z) \ (1.2 + z)}$$

```
plot1 = Plot[BPfixM, {z, -0.5, 0.5}, PlotStyle → Blue, AxesLabel → {"y", "Fix"},
  PlotRange → {Automatic, {-0.1, 0.2}}]
```



We next derive the invasibility of allele R, reversing the effects of allele M and R.
To consider this fixation probability, we now define the rare allele as m (actually allele R) and the resident allele as r (actually allele M).

```
fr = fM;
fm = fR;

βr = βM;
βm = βR;

βrr = βMM;
βrm = βRM;
βmm = βRR;

dr = dM;
dm = dR;

drr = dMM;
drm = dRM;
dmm = dRR;
```

$$\text{EqρH} = \frac{\sqrt{\beta rr * drr}}{\sqrt{\frac{fr*\beta r*dr}{2} + \sqrt{\beta rr * drr}}};$$

$$\text{EqρD} = \frac{\sqrt{\frac{fr*\beta r*dr}{2}}}{\sqrt{\frac{fr*\beta r*dr}{2} + \sqrt{\beta rr * drr}}};$$

$$\text{Fair} = \frac{\beta r}{dr};$$

$$\text{Faim} = \frac{\beta m}{dm};$$

$$\text{Fairr} = \frac{\beta rr}{drr};$$

$$\text{Fairm} = \frac{\beta rm}{drm};$$

$$\text{avef} = \frac{fm + fr}{2};$$

$$\text{Fair2} = \frac{fr}{2} * \text{Fair};$$

$$\text{Faim2} = \frac{\text{avef}}{2} * \text{Faim};$$

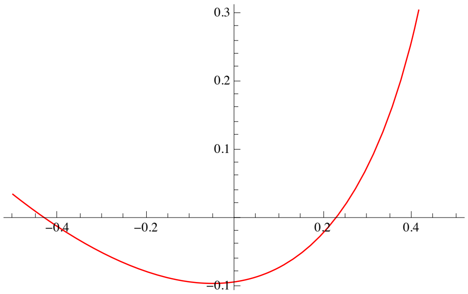
$$\text{BPPaiH} = \frac{\text{Faim2} * \text{Fairm} - \text{Fair2} * \text{Fairr}}{\text{Faim2} * \text{Fairm} + \frac{\text{Fairm}}{2} * \sqrt{\text{Fair2} * \text{Fairr}}};$$

$$\text{BPPaiD} = \frac{\text{Faim2} * \text{Fairm} - \text{Fair2} * \text{Fairr}}{\text{Faim2} * \text{Fairm} + 2 * \text{Faim2} * \sqrt{\text{Fair2} * \text{Fairr}}};$$

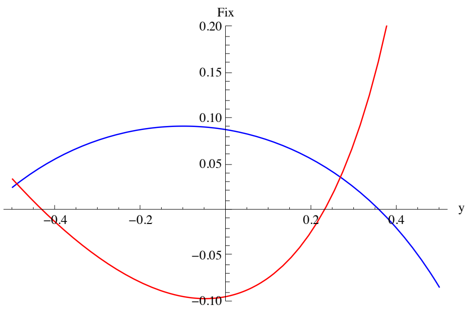
$$\text{BPfixm} = \frac{\text{EqρH}}{\text{EqρH} + 2 * \text{EqρD}} * \text{BPPaiH} + \frac{2 * \text{EqρD}}{\text{EqρH} + 2 * \text{EqρD}} * \text{BPPaiD} \;//\; \text{Simplify}$$

$$\left(\left(-2.\times 10^{9} + 4.\times 10^{9}\, z + 2.\times 10^{10}\, z^{2}\right)\left(\frac{2.23607\sqrt{1-2\,z}}{(1-1.\,z)\left(1.\times 10^{10} + 1.\times 10^{10}\sqrt{(1-2\,z)\,(1.2+z)}\right)} + \frac{2.23607\sqrt{1.2+z}}{1.\times 10^{10} - 1.\times 10^{10}\, z + 1.\times 10^{10}\sqrt{(1-2\,z)\,(1.2+z)}}\right)\right) \Big/ \left(2.23607\sqrt{1-2\,z} + 2.23607\sqrt{1.2+z}\right)$$

```
plot2 = Plot[BPfixm, {z, -0.5, 0.5}, PlotStyle → Red]
```




```
Show[plot1, plot2]
```



We have to hold y between -0.43 and 0.23 for M to invade when rare and to resist invasion by R when common.

```
NSolve[BPfixm == 0, z]
{{z → -0.431662}, {z → 0.231662}}
```

Fig.6(b)

Parameter values

```
Clear[x]

Site = 100;

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 0.2 + x;
SELwMM = -2 * x;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\text{SELwRM} = \frac{\text{wRM} - \text{wRR}}{\text{wRR}};$$

```
MinV = -0.51;
MaxV = 0.21;
```

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SELwM}\right) + \frac{1}{2} * (\text{SELwMM} * \text{HETEROw})$$

$$-0.5\,x + \frac{1}{2}\,(0.2 + x)$$

Out put

```
SampleMax = 10 000;
ChangeMax = 7;
OutputData = Table[0, {i, SampleMax * (ChangeMax + 1)}, {j, 2}];
OutputResult = Table[0, {i, ChangeMax + 1}, {j, 2}];

countData = 1;
countResult = 1;
```

Simulaton

```
VectpM = Table[0, {SampleMax}];

Resultparameter = Table[0, {ChangeMax + 1}];
Result = Table[0, {ChangeMax + 1}];

For[change = 1, change ≤ ChangeMax + 1, change++,

  If[change == 1, x = -0.5];
  If[change == 2, x = -0.4];
  If[change == 3, x = -0.3];
  If[change == 4, x = -0.2];
  If[change == 5, x = -0.1];
  If[change == 6, x = 0];
  If[change == 7, x = 0.1];
  If[change == 8, x = 0.2];


  sample = 1;
  For[sample = 1, sample ≤ SampleMax, sample = sample + 1,
```

$$\text{EqρH} = \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2} + \sqrt{\text{wRR}}}};$$

```
   xR = Round[Site * EqρH];
   xM = 0;
   xRR = Site - xR;
   xRM = 0;
   xMM = 0;
```

$$\text{sampleMutation} = \text{RandomVariate}\left[\text{MultinomialDistribution}\left[1, \left\{\frac{\text{xR}}{\text{xR} + 2*\text{xRR}}, \frac{2*\text{xRR}}{\text{xR} + 2*\text{xRR}}\right\}\right]\right];$$

```
   xR = xR - sampleMutation[[1]];
   xM = xM + sampleMutation[[1]];
   xRR = xRR - sampleMutation[[2]];
   xRM = xRM + sampleMutation[[2]];
   xMM = 0;
```

$$\text{If}\left[\text{xRR} + \text{xRM} + \text{xMM} == 0, \text{pM} = \frac{\text{xM}}{\text{xR} + \text{xM}}\right];$$

$$\text{If}\left[\text{xR} + \text{xM} == 0, \text{pM} = \frac{\frac{\text{xRM}}{2} + \text{xMM}}{\text{xRR} + \text{xRM} + \text{xMM}}\right];$$

$$\text{If}\left[\text{xR} + \text{xM} \neq 0 \;\&\&\; \text{xRR} + \text{xRM} + \text{xMM} \neq 0, \text{pM} = \frac{1}{2} * \frac{\frac{\text{xRM}}{2} + \text{xMM}}{\text{xRR} + \text{xRM} + \text{xMM}} + \frac{1}{2} * \frac{\text{xM}}{\text{xR} + \text{xM}}\right];$$

```
   step = 2;
   While[0 < pM && pM < 1,
```

$$\text{cR} = \text{wRR} * \text{xRR} + \frac{\text{wRM}}{2} * \text{xRM};$$

```
      cM = wMM * xMM + wRM/2 * xRM;
      If[xR == 0 && xM == 0, cRR = 0];
      If[xR ≠ 0 || xM ≠ 0, cRR = fR/2 * (wR^2 * xR^2)/(wR * xR + wM * xM)];
      If[xR == 0 && xM == 0, cRM = 0];
      If[xR ≠ 0 || xM ≠ 0, cRM = (fR + fM)/2 * (wR * wM * xR * xM)/(wR * xR + wM * xM)];
      If[xR == 0 && xM == 0, cMM = 0];
      If[xR ≠ 0 || xM ≠ 0, cMM = fM/2 * (wM^2 * xM^2)/(wR * xR + wM * xM)];
      pCellR = cR/(cR + cM + cRR + cRM + cMM);
      pCellM = cM/(cR + cM + cRR + cRM + cMM);
      pCellRR = cRR/(cR + cM + cRR + cRM + cMM);
      pCellRM = cRM/(cR + cM + cRR + cRM + cMM);
      pCellMM = cMM/(cR + cM + cRR + cRM + cMM);
      sampleBirth = RandomVariate[MultinomialDistribution[Site,
         {pCellR, pCellM, pCellRR, pCellRM, pCellMM}]];

      xR = sampleBirth[[1]];
      xM = sampleBirth[[2]];
      xRR = sampleBirth[[3]];
      xRM = sampleBirth[[4]];
      xMM = sampleBirth[[5]];
      If[xRR + xRM + xMM == 0, pM = xM/(xR + xM)];
      If[xR + xM == 0, pM = (xRM/2 + xMM)/(xRR + xRM + xMM)];
      If[xR + xM ≠ 0 && xRR + xRM + xMM ≠ 0, pM = 1/2 * (xRM/2 + xMM)/(xRR + xRM + xMM) + 1/2 * xM/(xR + xM)];

      step = step + 1;
     ];

    VectpM[[sample]] = pM;

    OutputData[[countData, 1]] = x;
    OutputData[[countData, 2]] = pM;
    countData = countData + 1;

   ];

   Resultparameter[[change]] = x;
   Result[[change]] = Total[VectpM];

   OutputResult[[countResult, 1]] = x;
   OutputResult[[countResult, 2]] = Total[VectpM];
   countResult = countResult + 1;

   Print[change]

  ];
```

1

2

3

4

5

6

7

8

Output of the data

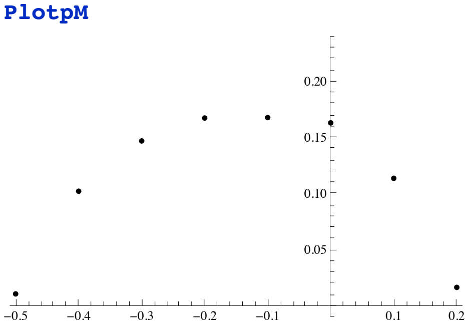
```
Resultparameter
Result
Export["/Users/Bessho/Desktop/workstation/OutputDataFig6b.txt", OutputData, "Table"];
Export["/Users/Bessho/Desktop/workstation/OutputResultFig6b.txt", OutputResult, "Table"];
```

```
{-0.5, -0.4, -0.3, -0.2, -0.1, 0, 0.1, 0.2}
{104, 1019, 1467, 1671, 1675, 1628, 1134, 162}
```

```
Resultparameter = {-0.5`, -0.4`, -0.3`, -0.2`, -0.1`, 0, 0.1`, 0.2`};
Result = {104, 1019, 1467, 1671, 1675, 1628, 1134, 162};
```

Plot

```
PlotpM = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotpMlog = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotpM[[change, 1]] = Resultparameter[[change]];
  PlotpM[[change, 2]] = Result[[change]] / SampleMax;
 ];
gNumerical = ListPlot[PlotpM, PlotRange → {{MinV, MaxV}, {-0.01, 0.24}},
  PlotStyle → {PointSize[0.012], Black}]
PlotpM
```



$$\left\{\left\{-0.5, \frac{13}{1250}\right\}, \left\{-0.4, \frac{1019}{10\,000}\right\}, \left\{-0.3, \frac{1467}{10\,000}\right\}, \left\{-0.2, \frac{1671}{10\,000}\right\}, \left\{-0.1, \frac{67}{400}\right\}, \left\{0, \frac{407}{2500}\right\}, \left\{0.1, \frac{567}{5000}\right\}, \left\{0.2, \frac{81}{5000}\right\}\right\}$$

Wilson score interval (95% Confidence interval)

Because the probability variable $\frac{\hat{u}-u}{\sqrt{u(1-u)/n}}$ (where, estimated fixation probability $\hat{u} = \frac{x}{n}$ and the fixation probability u) approximately depends on the normal distribution, Normal (0,1), we have 95% Confidence interval for the fixation probability, $\Pr[\underline{z} < u < \overline{z}] \sim 0.95$.

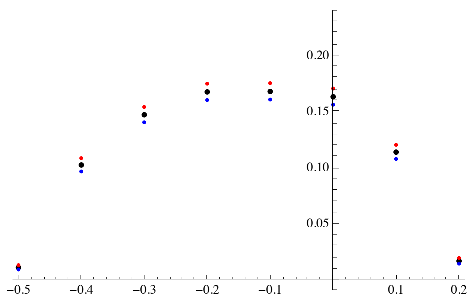
```
z = 1.96;
```

$$\underline{z} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2 * n} * z^2 - z * \sqrt{\frac{1}{n} * \hat{u} * (1 - \hat{u}) + \frac{1}{4 * n^2} * z^2}\right) /.\ n \to \text{SampleMax} /.\ \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

$$\overline{z} = \frac{n}{n + z^2}\left(\hat{u} + \frac{1}{2 * n} * z^2 + z * \sqrt{\frac{1}{n} * \hat{u} * (1 - \hat{u}) + \frac{1}{4 * n^2} * z^2}\right) /.\ n \to \text{SampleMax} /.\ \hat{u} \to \frac{\text{Result}}{\text{SampleMax}}$$

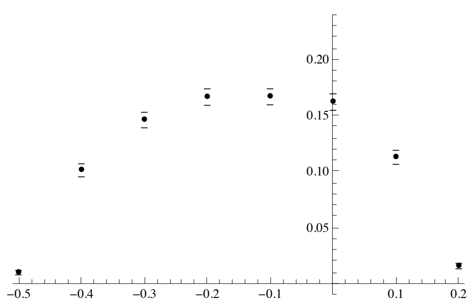
```
{0.00859113, 0.0961227, 0.139901, 0.159916, 0.160309, 0.155694, 0.107333, 0.0139049}
{0.0125849, 0.107983, 0.15377, 0.17454, 0.174946, 0.170165, 0.119764, 0.0188667}
```

```
PlotCIunder = Table[0, {i, ChangeMax + 1}, {j, 2}];
PlotCIupper = Table[0, {i, ChangeMax + 1}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCIunder[[change, 1]] = Resultparameter[[change]];
  PlotCIupper[[change, 1]] = Resultparameter[[change]];
  PlotCIunder[[change, 2]] = z̲[[change]];
  PlotCIupper[[change, 2]] = z̄[[change]];
 ];
gCIupperDot = ListPlot[PlotCIupper, PlotStyle → {Thickness[0.005], Red}];
gCIunderDot = ListPlot[PlotCIunder, PlotStyle → {Thickness[0.005], Blue}];
Show[gNumerical, gCIunderDot, gCIupperDot]
```

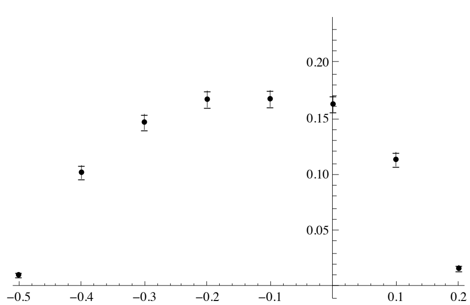



```
z̄
N[PlotpM[[All, 2]]]
z̲
gCIupper = ListPlot[PlotCIupper, PlotStyle → {Thickness[0.005], Black},
   PlotMarkers → {"-"}];
gCIunder = ListPlot[PlotCIunder, PlotStyle → {Thickness[0.005], Black},
   PlotMarkers → {"-"}];
Show[gNumerical, gCIunder, gCIupper]
```

```
{0.0125849, 0.107983, 0.15377, 0.17454, 0.174946, 0.170165, 0.119764, 0.0188667}
{0.0104, 0.1019, 0.1467, 0.1671, 0.1675, 0.1628, 0.1134, 0.0162}
{0.00859113, 0.0961227, 0.139901, 0.159916, 0.160309, 0.155694, 0.107333, 0.0139049}
```



```
PlotCI1 = Table[0, {i, 2}, {j, 2}];
PlotCI2 = Table[0, {i, 2}, {j, 2}];
PlotCI3 = Table[0, {i, 2}, {j, 2}];
PlotCI4 = Table[0, {i, 2}, {j, 2}];
PlotCI5 = Table[0, {i, 2}, {j, 2}];
PlotCI6 = Table[0, {i, 2}, {j, 2}];
PlotCI7 = Table[0, {i, 2}, {j, 2}];
PlotCI8 = Table[0, {i, 2}, {j, 2}];
For[change = 1, change ≤ ChangeMax + 1, change = change + 1,
  PlotCI1[[1, 1]] = Resultparameter[[1]];
  PlotCI1[[1, 2]] = z̲[[1]];
  PlotCI1[[2, 1]] = Resultparameter[[1]];
  PlotCI1[[2, 2]] = z̅[[1]];
  PlotCI2[[1, 1]] = Resultparameter[[2]];
  PlotCI2[[1, 2]] = z̲[[2]];
  PlotCI2[[2, 1]] = Resultparameter[[2]];
  PlotCI2[[2, 2]] = z̅[[2]];
  PlotCI3[[1, 1]] = Resultparameter[[3]];
  PlotCI3[[1, 2]] = z̲[[3]];
  PlotCI3[[2, 1]] = Resultparameter[[3]];
  PlotCI3[[2, 2]] = z̅[[3]];
  PlotCI4[[1, 1]] = Resultparameter[[4]];
  PlotCI4[[1, 2]] = z̲[[4]];
  PlotCI4[[2, 1]] = Resultparameter[[4]];
  PlotCI4[[2, 2]] = z̅[[4]];
  PlotCI5[[1, 1]] = Resultparameter[[5]];
  PlotCI5[[1, 2]] = z̲[[5]];
  PlotCI5[[2, 1]] = Resultparameter[[5]];
  PlotCI5[[2, 2]] = z̅[[5]];
  PlotCI6[[1, 1]] = Resultparameter[[6]];
  PlotCI6[[1, 2]] = z̲[[6]];
  PlotCI6[[2, 1]] = Resultparameter[[6]];
  PlotCI6[[2, 2]] = z̅[[6]];
  PlotCI7[[1, 1]] = Resultparameter[[7]];
  PlotCI7[[1, 2]] = z̲[[7]];
  PlotCI7[[2, 1]] = Resultparameter[[7]];
  PlotCI7[[2, 2]] = z̅[[7]];
  PlotCI8[[1, 1]] = Resultparameter[[8]];
  PlotCI8[[1, 2]] = z̲[[8]];
  PlotCI8[[2, 1]] = Resultparameter[[8]];
  PlotCI8[[2, 2]] = z̅[[8]];
 ];
gCI1 = ListPlot[PlotCI1, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI2 = ListPlot[PlotCI2, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI3 = ListPlot[PlotCI3, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI4 = ListPlot[PlotCI4, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI5 = ListPlot[PlotCI5, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI6 = ListPlot[PlotCI6, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI7 = ListPlot[PlotCI7, PlotStyle → {Thickness[0.001], Black}, Joined → True];
gCI8 = ListPlot[PlotCI8, PlotStyle → {Thickness[0.001], Black}, Joined → True];
Show[gNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6, gCI7, gCI8, gCIunder, gCIupper]
```



Fixation probability from branching process approximation

```
Clear[y]

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 0.2 + y;
SELwMM = -2 * y;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\text{SELwRM} = \frac{\text{wRM} - \text{wRR}}{\text{wRR}};$$

$$s_{\text{average}} = \frac{1}{2} * \left(\frac{\text{SELfM}}{2} + \text{SELwM}\right) + \frac{1}{2} * (\text{SELwMM} * \text{HETEROw})$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR}*\text{wR}}{2}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{avef} = \frac{\text{fM} + \text{fR}}{2};$$

$$\text{W} = \frac{\text{fR} * \text{wR}}{2} * \text{Eq}\rho\text{H} + \text{wRR} * \text{Eq}\rho\text{D};$$

$$\text{EqD4a} = 1 - \text{PH} - \text{Exp}\left[-\text{PD} * \frac{\text{avef} * \text{wM}}{\text{W}}\right];$$

$$\text{EqD4b} = 1 - \text{PD} - \text{Exp}\left[-\text{PH} * \frac{\text{wRM}}{2\ \text{W}}\right];$$

```
RepeatMax = 100;
ResultH = Table[0, {i, RepeatMax}, {j, 2}];
ResultD = Table[0, {i, RepeatMax}, {j, 2}];
ResultBPfix = Table[0, {i, RepeatMax}, {j, 2}];
For[repeat = 0, repeat ≤ RepeatMax, repeat++,
  x1 = MinV + (MaxV - MinV)/RepeatMax * repeat;
  CondH = EqD4a /. y → x1;
  CondD = EqD4b /. y → x1;
  sol = Solve[CondH == 0 && CondD == 0 && 0 < PH && PH < 1 && 0 < PD && PD < 1, {PH, PD}, Reals];
  BPPaiH = Part[(PH /. sol), 1];
  BPPaiD = Part[(PD /. sol), 1];
  ResultH[[repeat, 1]] = x1;
  ResultD[[repeat, 1]] = x1;
  ResultBPfix[[repeat, 1]] = x1;
  ResultH[[repeat, 2]] = BPPaiH;
  ResultD[[repeat, 2]] = BPPaiD;
```

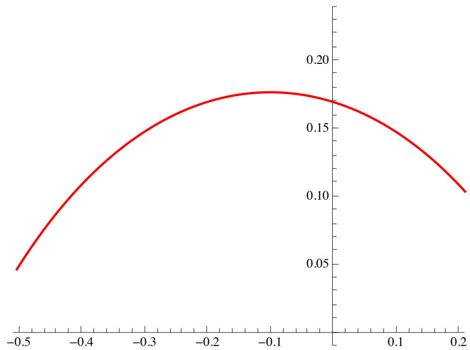
```
    ResultBPfix[[repeat, 2]] = EqρH / (EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD) / (EqρH + 2 * EqρD) * BPPaiD;
  ];
gBPfix = ListPlot[ResultBPfix, Joined → True, PlotRange → {{MinV, MaxV}, {-0.01, 0.24}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
```

$$-0.5\,y + \frac{1}{2}\,(0.2 + y)$$

Solve::ratnz : Solve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result. ≫

Solve::ratnz : Solve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result. ≫

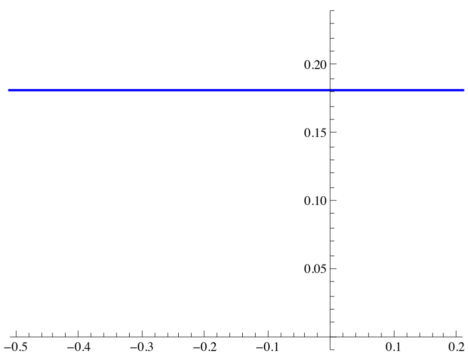
General::stop : Further output of Solve::ratnz will be suppressed during this calculation. ≫



Fixation probability from diffusion approximation

```
Parameters = {SUM → Site, ρ̂H → Sqrt[wRR] / (Sqrt[fR*wR/2] + Sqrt[wRR]), ρ̂D → Sqrt[fR*wR/2] / (Sqrt[fR*wR/2] + Sqrt[wRR]), sfM → SELfM,
   swM → SELwM, swRM → SELwRM, swMM → SELwMM}
Additive = swMM - 2 * swRM;
m = SUM * p * (1 - p) * (2 * s_average + p * Additive) / 2;
v = p * (1 - p) * (ρ̂H + 2 * ρ̂D) / (8 * ρ̂H * ρ̂D);
Q = Integrate[m/v /. Parameters, p];
INI = 1 / (SUM * (ρ̂H + 2 * ρ̂D)) /. Parameters;
DFfixWS = Integrate[Exp[-2 * Q], {p, 0, INI}] / Integrate[Exp[-2 * Q], {p, 0, 1}];
gDFfixWS = Plot[DFfixWS, {y, MinV, MaxV}, PlotRange → {{MinV, MaxV}, {-0.01, 0.24}},
   PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75]
```

$\left\{SUM \to 100,\ \hat{\rho}_H \to 0.666667,\ \hat{\rho}_D \to 0.333333,\ sfM \to 0,\ swM \to 0.2 + y,\ swRM \to \frac{-1000 + 1000\ (1 - 1.\ y)}{1000},\ swMM \to -2\ y\right\}$



Analytical result (Linear approximation)

```
FixLinear = (8 * ρ̂H * ρ̂D) / (ρ̂H + 2 * ρ̂D)^2 * 2 * s_average;
gFixLinear = Plot[FixLinear /. Parameters, {y, MinV, MaxV},
   PlotRange → {{MinV, MaxV}, {-0.01, 0.24}}, PlotStyle → {Thickness[0.005], Black, Dashed},
   AspectRatio → 0.75]
```

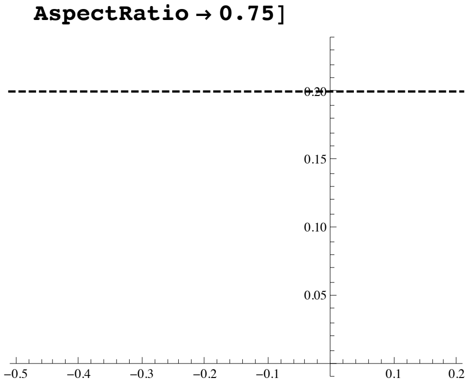
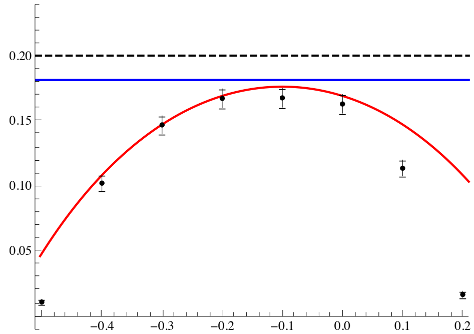


Figure for output

```
Show[gDFfixWS, gBPfix, gFixLinear, gNumerical, gCI1, gCI2, gCI3, gCI4, gCI5, gCI6,
 gCI7, gCI8, gCIunder, gCIupper, PlotRange → {{MinV, MaxV}, {-0.01, 0.24}},
 AxesOrigin → {-0.51, 0}]
```




```
Export["~/Desktop/Fig6b.PDF", %, "PDF"];
```

$$\frac{\sqrt{wRR}}{\sqrt{\frac{fR*wR}{2}}+\sqrt{wRR}}$$

0.666667

As ploidally antagonistic selection becomes stronger, however, the M allele stops being beneficial when common. We can measure selection on M as the difference in fitness of M-bearing versus R-bearing individuals relative to the common genotypes (MM and M):

```
Clear[z]
```

$$\text{aveselMcommon} = \frac{1}{2}\left(1-\frac{1}{1+0.2+z}\right)+\frac{1}{2}\left(1-\frac{1-z}{1-2z}\right)$$

$$\frac{1}{2}\left(1-\frac{1-z}{1-2z}\right)+\frac{1}{2}\left(1-\frac{1}{1.2+z}\right)$$

```
Solve[% == 0, z]
{{z → -0.376887}, {z → 0.176887}}
```

This is when M stops being selected when common.

We next ask when R can actually invade a population fixed on M, according to the Moran model, so that both M and R are predicted to invade when rare.

We first plot the fixation probability of allele M when allele M is rare.

```
Clear[z]
MaxV = 0.6;
MinV = -0.6;

fR = 0.5;
wR = 1000;
wRR = 1000;

SELfM = 0;
SELwM = 0.2 + z;
SELwMM = -2 * z;
HETEROw = 0.5;

fM = fR * (1 + SELfM);
wM = wR * (1 + SELwM);
wRM = wRR * (1 + HETEROw * SELwMM);
wMM = wRR * (1 + SELwMM);
```

$$\text{SELwRM} = \frac{\text{wRM}-\text{wRR}}{\text{wRR}};$$

$$s_{\text{average}} = \frac{1}{2}*\left(\frac{\text{SELfM}}{2}+\text{SELwM}\right)+\frac{1}{2}*(\text{SELwMM}*\text{HETEROw})$$

$$\text{Eq}\rho\text{H} = \frac{\sqrt{\text{wRR}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{Eq}\rho\text{D} = \frac{\sqrt{\frac{\text{fR}*\text{wR}}{2}}}{\sqrt{\frac{\text{fR}*\text{wR}}{2}} + \sqrt{\text{wRR}}};$$

$$\text{avef} = \frac{\text{fM} + \text{fR}}{2};$$

$$\text{W} = \frac{\text{fR} * \text{wR}}{2} * \text{Eq}\rho\text{H} + \text{wRR} * \text{Eq}\rho\text{D};$$

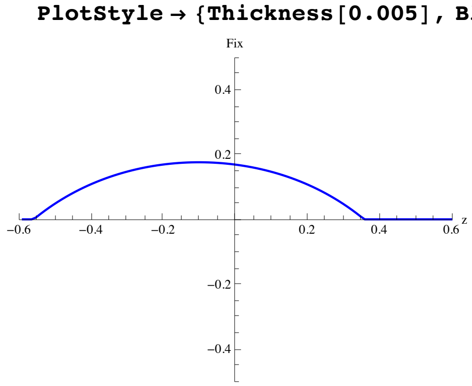
$$\text{EqD4a} = 1 - \text{PH} - \text{Exp}\left[-\text{PD} * \frac{\text{avef} * \text{wM}}{\text{W}}\right];$$

$$\text{EqD4b} = 1 - \text{PD} - \text{Exp}\left[-\text{PH} * \frac{\text{wRM}}{2\ \text{W}}\right];$$

```
RepeatMax = 100;
ResultH = Table[0, {i, RepeatMax}, {j, 2}];
ResultD = Table[0, {i, RepeatMax}, {j, 2}];
ResultBPfixM = Table[0, {i, RepeatMax}, {j, 2}];

For[repeat = 0, repeat ≤ RepeatMax, repeat++,
  x1 = MinV + (MaxV - MinV)/RepeatMax * repeat;
  CondH = EqD4a /. z → x1;
  CondD = EqD4b /. z → x1;
  sol = NSolve[CondH == 0 && CondD == 0, {PH, PD}, Reals];
  BPPaiH = Max[Part[(PH /. sol), 1], Part[(PH /. sol), 2]];
  BPPaiD = Max[Part[(PD /. sol), 1], Part[(PD /. sol), 2]];
  ResultH[[repeat, 1]] = x1;
  ResultD[[repeat, 1]] = x1;
  ResultBPfixM[[repeat, 1]] = x1;

  ResultH[[repeat, 2]] = BPPaiH;
  ResultD[[repeat, 2]] = BPPaiD;
  ResultBPfixM[[repeat, 2]] = EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD;
 ];
```

$$-0.5\ z + \frac{1}{2}\ (0.2 + z)$$

```
ResultBPfixM;
plot1 = ListPlot[ResultBPfixM, Joined → True, PlotRange → {{MinV, MaxV}, {-0.5, 0.5}},
  PlotStyle → {Thickness[0.005], Blue}, AspectRatio → 0.75, AxesLabel → {"z", "Fix"}]
```



We next derive the invasibility of allele R, reversing the effects of allele M and R.
To consider this fixation probability, we now define the rare allele as m (actually allele R) and the resident allele as r (actually allele M).

```
fr = fM;
fm = fR;

wr = wM;
wm = wR;

wrr = wMM;
wrm = wRM;
wmm = wRR;
```

$$\text{EqρH} = \frac{\sqrt{\text{wrr}}}{\sqrt{\frac{\text{fr}*\text{wr}}{2}} + \sqrt{\text{wrr}}};$$

$$\text{EqρD} = \frac{\sqrt{\frac{\text{fr}*\text{wr}}{2}}}{\sqrt{\frac{\text{fr}*\text{wr}}{2}} + \sqrt{\text{wrr}}};$$

$$\text{avef} = \frac{\text{fm} + \text{fr}}{2};$$

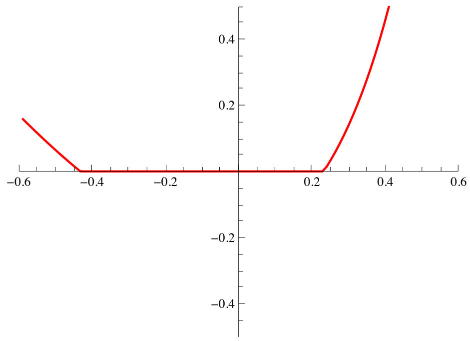
$$\text{W} = \frac{\text{fr} * \text{wr}}{2} * \text{EqρH} + \text{wrr} * \text{EqρD};$$

$$\text{EqD4a} = 1 - \text{PH} - \text{Exp}\left[-\text{PD} * \frac{\text{avef} * \text{wm}}{\text{W}}\right];$$

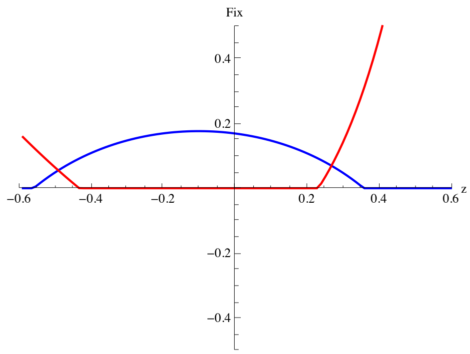
$$\text{EqD4b} = 1 - \text{PD} - \text{Exp}\left[-\text{PH} * \frac{\text{wrm}}{2\,\text{W}}\right];$$

```
RepeatMax = 100;
ResultH = Table[0, {i, RepeatMax}, {j, 2}];
ResultD = Table[0, {i, RepeatMax}, {j, 2}];
ResultBPfixm = Table[0, {i, RepeatMax}, {j, 2}];

For[repeat = 0, repeat ≤ RepeatMax, repeat++,

  x1 = MinV + (MaxV - MinV)/RepeatMax * repeat;
  CondH = EqD4a /. z → x1;
  CondD = EqD4b /. z → x1;
  sol = NSolve[CondH == 0 && CondD == 0, {PH, PD}, Reals];
  BPPaiH = Max[Part[(PH /. sol), 1], Part[(PH /. sol), 2]];
  BPPaiD = Max[Part[(PD /. sol), 1], Part[(PD /. sol), 2]];
  ResultH[[repeat, 1]] = x1;
  ResultD[[repeat, 1]] = x1;
  ResultBPfixm[[repeat, 1]] = x1;

  ResultH[[repeat, 2]] = BPPaiH;
  ResultD[[repeat, 2]] = BPPaiD;
  ResultBPfixm[[repeat, 2]] = EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD /. z → x1;
 ];
```

```
ResultBPfixm;
plot2 = ListPlot[ResultBPfixm, Joined → True, PlotRange → {{MinV, MaxV}, {-0.5, 0.5}},
  PlotStyle → {Thickness[0.005], Red}, AspectRatio → 0.75]
```




```
Show[plot1, plot2]
```



Again, we have to hold z between -0.432 and 0.232 for the allele to be beneficial when rare AND beneficial when common. Otherwise, the sign of selection flips so that M is actually deleterious when common.

```
MaxV = -0.43;
MinV = -0.44;
RepeatMax = 10;

For[repeat = 0, repeat ≤ RepeatMax, repeat++,
  x1 = MinV + (MaxV - MinV)/RepeatMax * repeat;
  CondH = EqD4a /. z → x1;
  CondD = EqD4b /. z → x1;
  sol = NSolve[CondH == 0 && CondD == 0, {PH, PD}, Reals];
  BPPaiH = Max[Part[(PH /. sol), 1], Part[(PH /. sol), 2]];
  BPPaiD = Max[Part[(PD /. sol), 1], Part[(PD /. sol), 2]];

  Print[x1, " ", EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD /. z → x1]
];
```

PolynomialGCD::lrgexp : Exponent is out of bounds for function PolynomialGCD. ≫

PolynomialGCD::lrgexp : Exponent is out of bounds for function PolynomialGCD. ≫

PolynomialGCD::lrgexp : Exponent is out of bounds for function PolynomialGCD. ≫

General::stop : Further output of PolynomialGCD::lrgexp will be suppressed during this calculation. ≫

NSolve::ratnz : NSolve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result. ≫

-0.44 0.0076254

NSolve::ratnz : NSolve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result. ≫

-0.439 0.00670556

NSolve::ratnz : NSolve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result. ≫

General::stop : Further output of NSolve::ratnz will be suppressed during this calculation. ≫

-0.438 0.00578714

-0.437 0.00487015

-0.436 0.00395459

-0.435 0.00304047

-0.434 0.00212778

-0.433 0.00121654

-0.432 0.000306748

-0.431 0.

-0.43 0.

```
MaxV = 0.23;
MinV = 0.24;
RepeatMax = 10;

For[repeat = 0, repeat ≤ RepeatMax, repeat++,
  x1 = MinV + (MaxV - MinV)/RepeatMax * repeat;
  CondH = EqD4a /. z → x1;
  CondD = EqD4b /. z → x1;
  sol = NSolve[CondH == 0 && CondD == 0, {PH, PD}, Reals];
  BPPaiH = Max[Part[(PH /. sol), 1], Part[(PH /. sol), 2]];
  BPPaiD = Max[Part[(PD /. sol), 1], Part[(PD /. sol), 2]];

  Print[x1, " ", EqρH/(EqρH + 2 * EqρD) * BPPaiH + (2 * EqρD)/(EqρH + 2 * EqρD) * BPPaiD /. z → x1]
];
```

NSolve::ratnz : NSolve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result. ≫

```
0.24 0.0145035
```

NSolve::ratnz : NSolve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result. ≫

```
0.239 0.0127283
```

NSolve::ratnz : NSolve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result. ≫

General::stop : Further output of NSolve::ratnz will be suppressed during this calculation. ≫

```
0.238 0.0109628
0.237 0.00920711
0.236 0.00746117
0.235 0.00572494
0.234 0.00399838
0.233 0.00228144
0.232 0.000574104
0.231 0.
0.23 0.
```

We next discuss whether a single copy of allele M could fix with a positive probability, according to a branching process model, if we were to consider M within a population alternating in states between {R,MM} and {M,RR}. The geometric mean fitness of the two alleles is then:

```
RelativeAdvantage = Sqrt[fM/2 * wM * wMM] - Sqrt[fR/2 * wR * wRR]
-500. + 500. √((1 - 2 z) (1.2 + z))
Solve[RelativeAdvantage == 0, z]
{{z → -0.821699}, {z → 0.121699}}
Plot[RelativeAdvantage, {z, -0.5, 0.25}]
```