In[3934]:=
```
Clear["Global`*"]
```

Parameters

In[3935]:=
```
ρmax = 1 (*maximum carbon uptake rate (d^-1)*);
αmax = 1.5 * 10^-9(*attack rate of mixotroph on bacteria (cm^2*d^-1*cellM^-1)*);
b = .15(*conversion rate of bacteria to mixotroph (cellM*cellB^-1)*);
KB = 1 × 10^8; (*carrying capacity of bacteria (cellB*cm^-2)*);
r = .693(*growth rate of bacteria (d^-1)*);
h = 250(*half saturation constant for photosynthesis (μmol quanta*m^2*s^-1)*);
Iin = 100(*incident light (μmol quanta*m^2*s^-1)*);
k = 5 * 10^(-7)(*mixotroph light absorbance constant (cm^2*cellM^-1)*);
l = .05(*mixotroph mortality rate (d^-1)*);
mρ = .1;
(*photosynthetic temeprature sensitivity coefficient (°C^-1)*);
mα = .25;
(*heterotrophic temperature sensitivity coefficient (°C^-1)*);
T0 = 13; (*baseline temperature (°C)*);
T0α = T0 - 1/mα (*minmimum temperature for heterotrophy (°C)*);
T0ρ = T0 - 1/mρ (*minmimum temperature for photosynthesis (°C)*);
kb = 8.62 * 10^-5(*Boltzmann constant (eV*K^-1)*);
Eaρ = .5 (*photosynthetic activation energy (eV)*);
Eaα = .85 (*heterotrophic activation energy (eV)*);
r0ρ = 6.4279909706*^8 (*photosynthetic normalization constant*);
r0α = 9.412997398*^14 (*heterotrophic normalization constant*);
```

Equations/Functions for generating outputs

Equations

In[3954]:=
```
(*temperature-dependent photosynthetic rate*)
ρ[θ_, z_, T_] := ρmax * (1 - θ^(2^z))^(1/2^z) (mρ (T - T0ρ))
```

$$\rho\mathrm{Exp}[\theta\_,\ z\_,\ T\_] := \rho_{max} * \left(1 - \theta^{2^{z}}\right)^{\frac{1}{2^{z}}}\ r0\rho\ E^{\frac{-E_{a\rho}}{k_b\ (273+T)}}$$

(*temperature-dependent grazing rate*)

$$\alpha[\theta\_,\ T\_] := \alpha_{max} * \theta\ \left(m_{\alpha}\ \left(T - T0\alpha\right)\right)$$

$$\alpha\mathrm{Exp}[\theta\_,\ T\_] := \alpha_{max} * \theta\ r0\alpha\ E^{\frac{-E_{a\alpha}}{k_b\ (273+T)}}$$

(*solves for mixotroph and bacteria population density at equilibrium*)

```
eqs[θ_, z_, T_] := FindRoot[{dM[θ, z, T] == 0, dB[θ, T] == 0 },
  {{M, 10^7}, {B, 10^7}}, AccuracyGoal → Infinity]
eqsExp[θ_, z_, T_] := FindRoot[{dMExp[θ, z, T] == 0, dBExp[θ, T] == 0 },
  {{M, 10^7}, {B, 10^7}}, AccuracyGoal → Infinity]
```

(*mixotroph per capita growth rate*)

$$dM[\theta\_,\ z\_,\ T\_] := \left(\frac{\rho[\theta,\ z,\ T]}{k\ M}\ \mathrm{Log}\left[\frac{\left(h + I_{in}\right)}{\left(h + I_{in} * \mathrm{Exp}[-k\ M\ ]\right)}\right] - l + \alpha[\theta,\ T]\ b\ B\right)$$

$$dM\mathrm{Exp}[\theta\_,\ z\_,\ T\_] := \left(\frac{\rho\mathrm{Exp}[\theta,\ z,\ T]}{k\ M}\ \mathrm{Log}\left[\frac{\left(h + I_{in}\right)}{\left(h + I_{in} * \mathrm{Exp}[-k\ M\ ]\right)}\right] - l + \alpha\mathrm{Exp}[\theta,\ T]\ b\ B\right)$$

(*bacteria per capita growth rate*)

$$dB[\theta\_,\ T\_] := \left(r\ \left(1 - \left(\frac{B}{K_B}\right)\right) - \alpha[\theta,\ T]\ M\right)$$

$$dB\mathrm{Exp}[\theta\_,\ T\_] := \left(r\ \left(1 - \left(\frac{B}{K_B}\right)\right) - \alpha\mathrm{Exp}[\theta,\ T]\ M\right)$$

(*mutant fitness equation*)

$$\mathrm{Fitness}[\theta m\_,\ z\_,\ M\_,\ B\_,\ T\_] := -l + \alpha[\theta m,\ T]\ b\ B + \frac{\rho[\theta m,\ z,\ T]\ \mathrm{Log}\left[\frac{h+I_{in}}{h+ e^{-(k\,M)}\ I_{in}}\right]}{k\ M}$$

$$\mathrm{FitnessExp}[\theta m\_,\ z\_,\ M\_,\ B\_,\ T\_] := -l + \alpha\mathrm{Exp}[\theta m,\ T]\ b\ B + \frac{\rho\mathrm{Exp}[\theta m,\ z,\ T]\ \mathrm{Log}\left[\frac{h+I_{in}}{h+ e^{-(k\,M)}\ I_{in}}\right]}{k\ M}$$

(*selection gradient*)

$$\mathrm{SelectionGrad}[\theta m\_,\ z\_,\ M\_,\ B\_,\ T\_] :=$$

$$\alpha_{max}\ \left(m_{\alpha}\ \left(T - T0\alpha\right)\right)\ b\ B - \frac{\rho[\theta m,\ z,\ T]\ \theta m^{-1+2^{z}}\ \left(1 - \theta m^{2^{z}}\right)^{-1}\ \mathrm{Log}\left[\frac{h+I_{in}}{h+e^{-k\,M}\ I_{in}}\right]}{k\ M}$$

$$\mathrm{SelectionGradExp}[\theta m\_,\ z\_,\ M\_,\ B\_,\ T\_] :=$$

$$b\ B\ e^{-\frac{e_{a\alpha}}{(273+T)\ k_b}}\ r0\alpha\ \alpha_{max} - \frac{e^{-\frac{e_{a\rho}}{(273+T)\ k_b}}\ r0\rho\ \theta m^{-1+2^{z}}\ \left(1 - \theta m^{2^{z}}\right)^{-1+2^{-z}}\ \mathrm{Log}\left[\frac{h+i_{in}}{h+e^{-k\,M}\ i_{in}}\right]\ \rho_{max}}{k\ M}$$

In[ ]:=

In[ ]:=

Pairwise invasibility plots (PIP)

In[3968]:=
```
(*Uses a chosen shape parameter z, temperature T,
and color to generate a pairwise invasibility plot using
 the fitness function for an invading mutant mixotroph. Regions
 in which mutant fitness is positive are shaded*)
MakePIP[z_, T_, color_] :=
 RegionPlot[Fitness[θm, z, If[(M /. eqs[θ, z, T]) ≥ 0, M /. eqs[θ, z, T], 0],
    If[(B /. eqs[θ, z, T]) ≥ 0, B /. eqs[θ, z, T], 0], T] ≥  0, {θ, 0, 1}, {θm, 0, 1},
  PlotStyle → color, BoundaryStyle → {Bold, Dashed, Black}, Frame → True,
  FrameLabel → {Style["Resident Heterotrophic investment (θres)", 12, Black],
    Style["Mutant Heterotrophic investment (θmut)", 12, Black]},
  FrameTicksStyle → Directive[Black, 12], ImageSize → Medium]
MakePIPNV[z_, T_, color_, colorm_] :=
 RegionPlot[Fitness[θm, z, M /. eqs[θ, z, T], B /. eqs[θ, z, T], T] ≥  0,
  {θ, 0, 1}, {θm, 0, 1}, PlotStyle → color, Mesh → 20, MeshStyle → colorm,
  BoundaryStyle → {Bold, Dashed, Black}, Frame → True,
  FrameLabel → {Style["Resident Heterotrophic investment (θres)", 12, Black],
    Style["Mutant Heterotrophic investment (θmut)", 12, Black]},
  FrameTicksStyle → Directive[Black, 12], ImageSize → Medium]

MakePIPexp[z_, T_, color_] :=
 RegionPlot[Fitness[θm, z, If[(M /. eqsExp[θ, z, T]) ≥ 0, M /. eqsExp[θ, z, T], 0],
    If[(B /. eqsExp[θ, z, T]) ≥ 0, B /. eqsExp[θ, z, T], 0], T] ≥  0, {θ, 0, 1},
  {θm, 0, 1}, PlotStyle → color, BoundaryStyle → {Bold, Dashed, Black}, Frame → True,
  FrameLabel → {Style["Resident Heterotrophic investment (θres)", 12, Black],
    Style["Mutant Heterotrophic investment (θmut)", 12, Black]},
  FrameTicksStyle → Directive[Black, 12], ImageSize → Medium]
MakePIPNVexp[z_, T_, color_, colorm_] :=
 RegionPlot[Fitness[θm, z, M /. eqsExp[θ, z, T], B /. eqsExp[θ, z, T], T] ≥  0,
  {θ, 0, 1}, {θm, 0, 1}, PlotStyle → color, Mesh → 20, MeshStyle → colorm,
  BoundaryStyle → {Bold, Dashed, Black}, Frame → True,
  FrameLabel → {Style["Resident Heterotrophic investment (θres)", 12, Black],
    Style["Mutant Heterotrophic investment (θmut)", 12, Black]},
  FrameTicksStyle → Directive[Black, 12], ImageSize → Medium]
```

Generating ESS vs Temp plots for generalist and specialist mixotrophs

In[∘]:=
```
(*makeListGen[] and makeListLin[] generate lists containing the
 evolutionarily stable investment strategy θESS as a function of temperature
 for generalist tradeoff and linear tradeoff mixotrophs respectively*)
```

In[3972]:=
```
makeθListGen[] :=
 {
  θESSgen = {};
  Quiet[For[T = 1, T < 41, T++,
    currentθ = θm /. FindRoot[
       SelectionGrad[θm, 1, M /. eqs[θm, 1, T], B /. eqs[θm, 1, T], T] == 0, {θm, .99}];
    If[Re[currentθ] > 1, AppendTo[θESSgen, 1], (*if calculated θESS >
      1 (maximum heterotrophic investment), 1 is added to list*)
     If[Re[currentθ] < 0, AppendTo[θESSgen, 0.], AppendTo[θESSgen, currentθ]]]
     (*if calculate θESS < 0 (minimum heterotrophic investment),
    0 is added to list*)
   ]];
  θESSgen
 }
makeθListLin[] :=
 {
  θESSlin = {};
  Quiet[For[T = 1, T < 41, T++,
    currentθ = θm /. FindRoot[
       SelectionGrad[θm, 0, M /. eqs[θm, 0, T], B /. eqs[θm, 0, T], T] == 0, {θm, .5}];
    (*getting around issue where FindRoot identifies the incorrect,
    evolutionarily unstable root in some cases*)
    If[1/.0001 (SelectionGrad[.0002, 0, M /. eqs[.0002, 0, T], B /. eqs[.0002, 0, T], T] -
         SelectionGrad[.0001, 0, M /. eqs[.0001, 0, T], B /. eqs[.0001, 0, T], T]) > 0,
     AppendTo[θESSlin, 1], If[Re[currentθ] > 1, AppendTo[θESSlin, 1], If[
       Re[currentθ] < 0, AppendTo[θESSlin, 0.000], AppendTo[θESSlin, currentθ]]]]
   ]];
  θESSlin
 }

makeθListGenExp[] :=
 {
  θESSgen = {};
  Quiet[For[T = 1, T < 41, T++,
    currentθ = θm /. FindRoot[SelectionGradExp[θm, 1,
         M /. eqsExp[θm, 1, T], B /. eqsExp[θm, 1, T], T] == 0, {θm, .99}];
    If[Re[currentθ] > 1, AppendTo[θESSgen, 1], (*if calculated θESS >
```

```
      1 (maximum heterotrophic investment), 1 is added to list*)
     If[Re[currentθ] < 0, AppendTo[θESSgen, 0.], AppendTo[θESSgen, currentθ]]]
     (*if calculate θESS < 0 (minimum heterotrophic investment),
    0 is added to list*)
   ]];
  θESSgen
 }
makeθListLinExp[] :=
 {
  θESSlin = {};
  Quiet[For[T = 1, T < 41, T++,
    currentθ = θm /. FindRoot[SelectionGradExp[θm, 0,
         M /. eqsExp[θm, 0, T], B /. eqsExp[θm, 0, T], T] == 0, {θm, .5}];
    (*getting around issue where FindRoot identifies the incorrect,
    evolutionarily unstable root in some cases*)
    If[1/.0001 (SelectionGradExp[.0002, 0, M /. eqsExp[.0002, 0, T],
          B /. eqsExp[.0002, 0, T], T] - SelectionGradExp[.0001, 0,
          M /. eqsExp[.0001, 0, T], B /. eqsExp[.0001, 0, T], T]) > 0,
     AppendTo[θESSlin, 1], If[Re[currentθ] > 1, AppendTo[θESSlin, 1], If[
       Re[currentθ] < 0, AppendTo[θESSlin, 0.000], AppendTo[θESSlin, currentθ]]]]
   ]];
  θESSlin
 }
makeθListGen[] :=
 {
  θESSgen = {};
  Quiet[For[T = 1, T < 41, T++,
    currentθ = θm /. FindRoot[
       SelectionGrad[θm, 1, M /. eqs[θm, 1, T], B /. eqs[θm, 1, T], T] == 0, {θm, .99}];
    If[Re[currentθ] > 1, AppendTo[θESSgen, 1], (*if calculated θESS >
      1 (maximum heterotrophic investment), 1 is added to list*)
     If[Re[currentθ] < 0, AppendTo[θESSgen, 0.], AppendTo[θESSgen, currentθ]]]
     (*if calculate θESS < 0 (minimum heterotrophic investment),
    0 is added to list*)
   ]];
  θESSgen
 }
makeθListLin[] :=
 {
  θESSlin = {};
  Quiet[For[T = 1, T < 41, T++,
    currentθ = θm /. FindRoot[
```

```
      SelectionGrad[θm, 0, M /. eqs[θm, 0, T], B /. eqs[θm, 0, T], T] == 0, {θm, .5}];
    (*getting around issue where FindRoot identifies the incorrect,
    evolutionarily unstable root in some cases*)
    If[1/.0001 (SelectionGrad[.0002, 0, M /. eqs[.0002, 0, T], B /. eqs[.0002, 0, T], T] -
         SelectionGrad[.0001, 0, M /. eqs[.0001, 0, T], B /. eqs[.0001, 0, T], T]) > 0,
     AppendTo[θESSlin, 1], If[Re[currentθ] > 1, AppendTo[θESSlin, 1], If[
       Re[currentθ] < 0, AppendTo[θESSlin, 0.000], AppendTo[θESSlin, currentθ]]]]
   ]];
  θESSlin
 }
makeθListGenExp[] :=
 {
  θESSgen = {};
  Quiet[For[T = 1, T < 41, T++,
    currentθ = θm /. FindRoot[SelectionGradExp[θm, 1,
         M /. eqsExp[θm, 1, T], B /. eqsExp[θm, 1, T], T] == 0, {θm, .99}];
    If[Re[currentθ] > 1, AppendTo[θESSgen, 1], (*if calculated θESS >
      1 (maximum heterotrophic investment), 1 is added to list*)
     If[Re[currentθ] < 0, AppendTo[θESSgen, 0.], AppendTo[θESSgen, currentθ]]]
     (*if calculate θESS < 0 (minimum heterotrophic investment),
    0 is added to list*)
   ]];
  θESSgen
 }
makeθListLinExp[] :=
 {
  θESSlin = {};
  Quiet[For[T = 1, T < 41, T++,
    currentθ = θm /. FindRoot[SelectionGradExp[θm, 0,
         M /. eqsExp[θm, 0, T], B /. eqsExp[θm, 0, T], T] == 0, {θm, .5}];
    (*getting around issue where FindRoot identifies the incorrect,
    evolutionarily unstable root in some cases*)
    If[1/.0001 (SelectionGradExp[.0002, 0, M /. eqsExp[.0002, 0, T],
          B /. eqsExp[.0002, 0, T], T] - SelectionGradExp[.0001, 0,
          M /. eqsExp[.0001, 0, T], B /. eqsExp[.0001, 0, T], T]) > 0,
     AppendTo[θESSlin, 1], If[Re[currentθ] > 1, AppendTo[θESSlin, 1], If[
       Re[currentθ] < 0, AppendTo[θESSlin, 0.000], AppendTo[θESSlin, currentθ]]]]
   ]];
  θESSlin
 }
```

```
makeθListSpec[] :=
 {
  θESSspec1viable = Table[Null, 40];
  θESSspec0 = {};
  Quiet[For[T = 1, T < 41, T++,
    If[eqs[1, -1, T][[1]][[2]] > 0 ,
     If[10^15 > eqs[.95, -1, T][[1]][[2]] > 0, θESSspec1viable[[T ;; T]] = 1]];
    AppendTo[θESSspec0, 0];
   ]];
  θESSspec = Table[θESSspec1viable, θESSspec0];
  θESSspec
 }

makeθListSpecExp[] :=
 {
  θESSspec1viable = Table[Null, 40];
  θESSspec0 = {};
  Quiet[For[T = 1, T < 41, T++,
    If[eqsExp[1, -1, T][[1]][[2]] > 0 ,
     If[10^15 > eqsExp[.95, -1, T][[1]][[2]] > 0, θESSspec1viable[[T ;; T]] = 1]];
    AppendTo[θESSspec0, 0];
   ]];
  θESSspec = Table[θESSspec1viable, θESSspec0];
  θESSspec
 }
```

In[∘]:=

Comparing evolved vs unevolved mixotrophs for carbon cycling

```
In[3991]:= (*generates plots comparing mixotroph and bacteria populations,
and growth rate components derived from photosynthesis,
P(θ,z,I,T,M*), and heterotrophy, G(θ,T,B*),
between evolving mixotrophs whos heterotrophic investment θ varies
 as a function of temperature and genetically static mixotrophs with
 fixed θ. This allows evolutionary and strictly thermal responses
 to be compared. Function inputs consist of the shape parameter, z,
the chosen lower bounds for each set of plots (l1, l2, and l3) and
 the chosen upper bounds for each set of plots (u1, u2, and u3)*)
Ccycling[z_, l1_, u1_, l2_, u2_, l3_, u3_] :=
 {makeθListLin[];
  makeθListGen[];
```

```
θList = List[];
If[z == 0, θList = θESSlin, θList = θESSgen];

Mpopsevo = List[];
Mpopsnoevo = List[];
Quiet[For[t = 1, t < 100, t++, AppendTo[Mpopsevo, M /. eqs[θList[[t]], z, t]]]];
Quiet[For[t = 1, t < 100, t++, AppendTo[Mpopsnoevo, M /. eqs[θList[[T0]], z, t]]]];

bpopsevo = List[];
bpopsnoevo = List[];
Quiet[For[t = 1, t < 100, t++, AppendTo[bpopsevo, B /. eqs[θList[[t]], z, t]]]];
Quiet[For[t = 1, t < 100, t++, AppendTo[bpopsnoevo, B /. eqs[θList[[T0]], z, t]]]];

photgrowthevo = List[];
photgrowthnoevo = List[];
Quiet[For[t = 1, t < 100, t++,
  AppendTo[photgrowthevo, (M /. eqs[θList[[t]], z, t]) * ((ρ[θList[[t]], z, t]
       Log[(h + I_in) / (h + ⅇ^-(k (M/.eqs[θList[[t]],z,t])) I_in)]) / (k (M /. eqs[θList[[t]], z, t])))]]];
Quiet[For[t = 1, t < 100, t++, AppendTo[photgrowthnoevo,
   (M /. eqs[θList[[T0]], z, t]) *
    ((ρ[θList[[T0]], z, t] Log[(h + I_in) / (h + ⅇ^-(k (M/.eqs[θList[[T0]],z,t])) I_in)]) /
      (k (M /. eqs[θList[[T0]], z, t])))]]];


hetgrowthevo = List[];
hetgrowthnoevo = List[];
Quiet[For[t = 1, t < 100, t++, AppendTo[hetgrowthevo,
   (B /. eqs[θList[[t]], z, t]) * (M /. eqs[θList[[t]], z, t]) α[θList[[t]], t] b]]];
Quiet[For[t = 1, t < 100, t++, AppendTo[hetgrowthnoevo,
   (B /. eqs[θList[[T0]], z, t]) *
    (M /. eqs[θList[[T0]], z, t]) α[θList[[T0]], t] b]]];


List[ListPlot[{Mpopsevo, Mpopsnoevo}, Joined → True, PlotRange →
    {{T0, 33}, {l1, u1}}, PlotStyle → {{If[z > 0, RGBColor["#B09771"], Black]},
     {If[z > 0, RGBColor["#B09771"], Black], Dashed}}, Frame → True,
   FrameLabel → {Style["Temperature (°C)", 15, Black], Style["M*", 15, Black]},
   FrameTicksStyle → Directive[Black, 12], ImageSize → Medium],
  ListPlot[{bpopsevo, bpopsnoevo}, Joined → True, PlotRange → {{T0, 33}, {l2, u2}},
```

```
    PlotStyle → {{If[z > 0, RGBColor["#B09771"], Black]},
      {If[z > 0, RGBColor["#B09771"], Black], Dashed}}, Frame → True,
    FrameLabel → {Style["Temperature (°C)", 15, Black], Style["B*", 15, Black]},
    FrameTicksStyle → Directive[Black, 12], ImageSize → Medium],
   ListPlot[{photgrowthevo, photgrowthnoevo, hetgrowthevo, hetgrowthnoevo},
    Joined → True, PlotRange → {{T0, 33}, {l3, u3}},
    PlotStyle → {{Darker[Green]}, {Darker[Green], Dashed}, {Black}, {Black, Dashed}},
    Frame → True, FrameLabel → {Style["Temperature (°C)", Black, 15],
      Style["P(θ,z,I,T,M*)*M, G(θ,T,B*)*M", Black, 12]},
    FrameTicksStyle → Directive[Black, 12], ImageSize → Medium]]
 }
Ccyclingspec[z_, l1_, u1_, l2_, u2_, l3_, u3_] :=
 {makeθListSpec[];
  θList0 = Table[0, 40];
  θList1 = Table[1, 40];

  Mpopsevo0 = List[];
  Mpopsnoevo0 = List[];
  Quiet[For[t = 1, t < 100, t++, AppendTo[Mpopsevo0, M /. eqs[θList0[[t]], z, t]]]];
  Quiet[
   For[t = 1, t < 100, t++, AppendTo[Mpopsnoevo0, M /. eqs[θList0[[T0]], z, t]]]];

  Mpopsevo1 = List[];
  Mpopsnoevo1 = List[];
  Quiet[For[t = 1, t < 100, t++, AppendTo[Mpopsevo1, M /. eqs[θList1[[t]], z, t]]]];
  Quiet[
   For[t = 1, t < 100, t++, AppendTo[Mpopsnoevo1, M /. eqs[θList1[[T0]], z, t]]]];

  bpopsevo0 = List[];
  bpopsnoevo0 = List[];
  Quiet[For[t = 1, t < 100, t++, AppendTo[bpopsevo0, B /. eqs[θList0[[t]], z, t]]]];
  Quiet[
   For[t = 1, t < 100, t++, AppendTo[bpopsnoevo0, B /. eqs[θList0[[T0]], z, t]]]];

  bpopsevo1 = List[];
  bpopsnoevo1 = List[];
  Quiet[For[t = 1, t < 100, t++, AppendTo[bpopsevo1, B /. eqs[θList1[[t]], z, t]]]];
  Quiet[
   For[t = 1, t < 100, t++, AppendTo[bpopsnoevo1, B /. eqs[θList1[[T0]], z, t]]]];

  photgrowthevo0 = List[];
  photgrowthnoevo0 = List[];
  Quiet[
```

```
 For[t = 1, t < 100, t++, AppendTo[photgrowthevo0, (M /. eqs[θList0[[t]], z, t]) *
    ((ρ[θList0[[t]], z, t] Log[(h + Iin) / (h + e^-(k (M/.eqs[θList0[[t]],z,t])) Iin)]) /
      (k (M /. eqs[θList0[[t]], z, t])))]]];
Quiet[For[t = 1, t < 100, t++, AppendTo[photgrowthnoevo0,
   (M /. eqs[θList0[[T0]], z, t]) *
    ((ρ[θList0[[T0]], z, t] Log[(h + Iin) / (h + e^-(k (M/.eqs[θList0[[T0]],z,t])) Iin)]) /
      (k (M /. eqs[θList0[[T0]], z, t])))]]];

photgrowthevo1 = List[];
photgrowthnoevo1 = List[];
Quiet[
 For[t = 1, t < 100, t++, AppendTo[photgrowthevo1, (M /. eqs[θList1[[t]], z, t]) *
    ((ρ[θList1[[t]], z, t] Log[(h + Iin) / (h + e^-(k (M/.eqs[θList1[[t]],z,t])) Iin)]) /
      (k (M /. eqs[θList1[[t]], z, t])))]]];
Quiet[For[t = 1, t < 100, t++, AppendTo[photgrowthnoevo1,
   (M /. eqs[θList1[[T0]], z, t]) *
    ((ρ[θList1[[T0]], z, t] Log[(h + Iin) / (h + e^-(k (M/.eqs[θList1[[T0]],z,t])) Iin)]) /
      (k (M /. eqs[θList1[[T0]], z, t])))]]];


hetgrowthevo0 = List[];
hetgrowthnoevo0 = List[];
Quiet[
 For[t = 1, t < 100, t++, AppendTo[hetgrowthevo0, (B /. eqs[θList0[[t]], z, t]) *
    (M /. eqs[θList0[[t]], z, t]) α[θList0[[t]], t] b]]];
Quiet[For[t = 1, t < 100, t++, AppendTo[hetgrowthnoevo0,
   (B /. eqs[θList0[[T0]], z, t]) *
    (M /. eqs[θList0[0[T0]], z, t]) α[θList0[[T0]], t] b]]];

hetgrowthevo1 = List[];
hetgrowthnoevo1 = List[];
Quiet[
 For[t = 1, t < 100, t++, AppendTo[hetgrowthevo1, (B /. eqs[θList1[[t]], z, t]) *
    (M /. eqs[θList1[[t]], z, t]) α[θList1[[t]], t] b]]];
Quiet[For[t = 1, t < 100, t++, AppendTo[hetgrowthnoevo1,
```

```
    (B /. eqs[θList1[[T0]], z, t]) *
     (M /. eqs[θList1[0[T0]], z, t]) α[θList1[[T0]], t] b]]];


List[ListPlot[{Mpopsevo1, Mpopsevo0, Mpopsnoevo1, Mpopsnoevo0}, Joined → True,
   PlotRange → {{T0, 33}, {l1, u1}}, PlotStyle → {{Lighter[RGBColor["#287DAB"]]},
     {RGBColor["#287DAB"]}, {Lighter[RGBColor["#287DAB"]], Dashed},
     {RGBColor["#287DAB"], Dashed}}, Frame → True,
   FrameLabel → {Style["Temperature (°C)", 15, Black], Style["M*", 15, Black]},
   FrameTicksStyle → Directive[Black, 12], ImageSize → Medium],
  ListPlot[{bpopsevo1, bpopsevo0, bpopsnoevo1, bpopsnoevo0}, Joined → True,
   PlotRange → {{T0, 33}, {l2, u2}}, PlotStyle → {{Lighter[RGBColor["#287DAB"]]},
     {RGBColor["#287DAB"]}, {Lighter[RGBColor["#287DAB"]], Dashed},
     {RGBColor["#287DAB"], Dashed}}, Frame → True,
   FrameLabel → {Style["Temperature (°C)", 15, Black], Style["B*", 15, Black]},
   FrameTicksStyle → Directive[Black, 12], ImageSize → Medium],
  ListPlot[{photgrowthevo0, photgrowthevo1, photgrowthnoevo0, photgrowthnoevo1,
    hetgrowthevo0, hetgrowthevo1, hetgrowthnoevo0, hetgrowthnoevo1},
   Joined → True, PlotRange → {{T0, 33}, {l3, u3}},
   PlotStyle → {{Darker[Green]}, {Darker[Green]}, {Darker[Green], Dashed},
     {Darker[Green], Dashed}, {Black}, {Black}, {Black, Dashed}, {Black, Dashed}},
   Frame → True, FrameLabel → {Style["Temperature (°C)", Black, 15],
     Style["P(θ,z,I,T,M*)*M, G(θ,T,B*)*M", Black, 12]},
   FrameTicksStyle → Directive[Black, 12], ImageSize → Medium]]
}
```

In[•]:=

In[•]:=

θ vs. Temperature plots

I_{in}=100

Linear temperature dependence

In[3982]:=
```
List[KB = 1 × 10^8; Iin = 100;
  ListPlot[{makeθListGen[] // Flatten, makeθListLin[] // Flatten,
    makeθListSpec[][[1]][[1]], makeθListSpec[][[1]][[2]]},
   Joined → True, PlotRange → {{9, 33}, {-.05, 1.05}},
   PlotStyle → {{RGBColor["#B09771"], AbsoluteThickness[3]},
     {Black, AbsoluteThickness[3]}, {RGBColor["#287DAB"], AbsoluteThickness[3]},
     {RGBColor["#287DAB"], AbsoluteThickness[3]}}, Frame → True,
   ImageSize → Medium, FrameLabel → {Style["Temperature (°C)", 12, Black],
     Style["Heterotrophic investment (θESS)", 12, Black]},
   FrameTicksStyle → Directive[Black, 12],
   PlotLabel → Style["KB=1*10^8", 12, Black]], KB = 2 × 10^8;
  Iin = 100;
  ListPlot[{makeθListGen[] // Flatten, makeθListLin[] // Flatten,
    makeθListSpec[][[1]][[1]], makeθListSpec[][[1]][[2]]},
   Joined → True, PlotRange → {{9, 33}, {-.05, 1.05}},
   PlotStyle → {{RGBColor["#B09771"], AbsoluteThickness[3]},
     {Black, AbsoluteThickness[3]}, {RGBColor["#287DAB"], AbsoluteThickness[3]},
     {RGBColor["#287DAB"], AbsoluteThickness[3]}}, Frame → True,
   ImageSize → Medium, FrameLabel → {Style["Temperature (°C)", 12, Black],
     Style["Heterotrophic investment (θESS)", 12, Black]},
   FrameTicksStyle → Directive[Black, 12],
   PlotLabel → Style["KB=2*10^8", 12, Black]], KB = 3 × 10^8;
  Iin = 100;
  ListPlot[{makeθListGen[] // Flatten, makeθListLin[] // Flatten,
    makeθListSpec[][[1]][[1]], makeθListSpec[][[1]][[2]]},
   Joined → True, PlotRange → {{9, 33}, {-.05, 1.05}},
   PlotStyle → {{RGBColor["#B09771"], AbsoluteThickness[3]},
     {Black, AbsoluteThickness[3]}, {RGBColor["#287DAB"], AbsoluteThickness[3]},
     {RGBColor["#287DAB"], AbsoluteThickness[3]}}, Frame → True,
   ImageSize → Medium, FrameLabel → {Style["Temperature (°C)", 12, Black],
     Style["Heterotrophic investment (θESS)", 12, Black]},
   PlotLegends → {"generalist", "linear", "specialist"},
   FrameTicksStyle → Directive[Black, 12], PlotLabel → Style["KB=3*10^8", 12, Black]]]
```

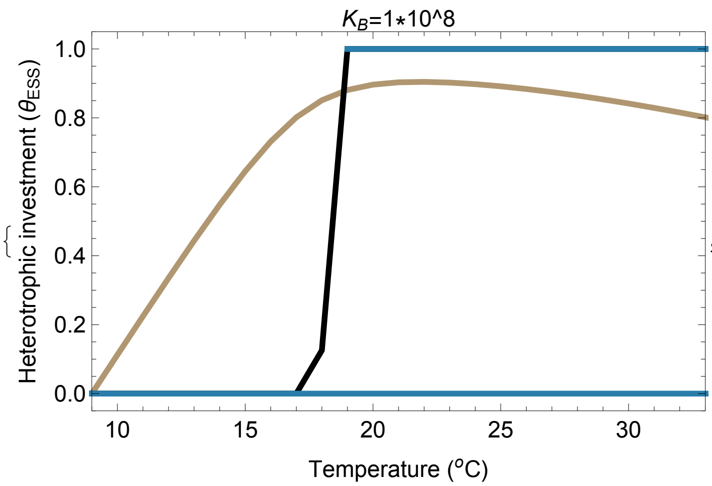
Table: Non−list iterator θESSspec0 at position 2 does not evaluate to a real numeric value.

••• Table: Non−list iterator θESSspec0 at position 2 does not evaluate to a real numeric value.

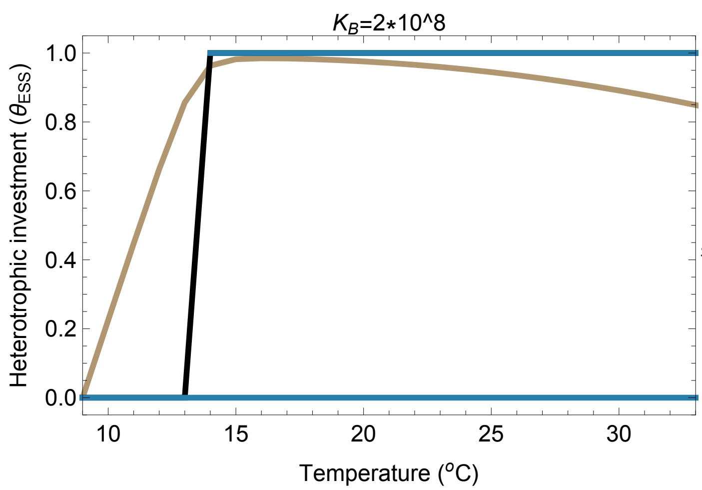
••• Table: Non−list iterator θESSspec0 at position 2 does not evaluate to a real numeric value.

••• General: Further output of Table::nliter will be suppressed during this calculation.

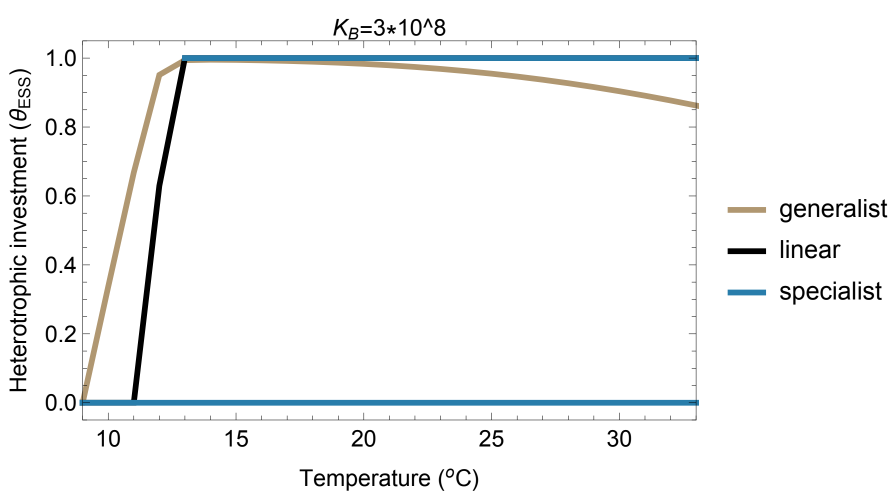
Out[3982]= {



,



,



}

Exponential temperature dependence

In[•]:=
```
List[KB = 1 × 10^8; Iin = 100;
 ListPlot[{makeθListGenExp[] // Flatten, makeθListLinExp[] // Flatten,
   makeθListSpecExp[][[1]][[1]], makeθListSpecExp[][[1]][[2]]},
  Joined → True, PlotRange → {{9, 33}, {-.05, 1.05}},
  PlotStyle → {{RGBColor["#B09771"], AbsoluteThickness[3]},
    {Black, AbsoluteThickness[3]}, {RGBColor["#287DAB"], AbsoluteThickness[3]},
    {RGBColor["#287DAB"], AbsoluteThickness[3]}}, Frame → True,
  ImageSize → Medium, FrameLabel → {Style["Temperature (°C)", 12, Black],
    Style["Heterotrophic investment (θESS)", 12, Black]},
  FrameTicksStyle → Directive[Black, 12],
  PlotLabel → Style["KB=1*10^8", 12, Black]], KB = 2 × 10^8;
 Iin = 100;
 ListPlot[{makeθListGenExp[] // Flatten, makeθListLinExp[] // Flatten,
   makeθListSpecExp[][[1]][[1]], makeθListSpecExp[][[1]][[2]]},
  Joined → True, PlotRange → {{9, 33}, {-.05, 1.05}},
  PlotStyle → {{RGBColor["#B09771"], AbsoluteThickness[3]},
    {Black, AbsoluteThickness[3]}, {RGBColor["#287DAB"], AbsoluteThickness[3]},
    {RGBColor["#287DAB"], AbsoluteThickness[3]}}, Frame → True,
  ImageSize → Medium, FrameLabel → {Style["Temperature (°C)", 12, Black],
    Style["Heterotrophic investment (θESS)", 12, Black]},
  FrameTicksStyle → Directive[Black, 12],
  PlotLabel → Style["KB=2*10^8", 12, Black]], KB = 3 × 10^8;
 Iin = 100;
 ListPlot[{makeθListGenExp[] // Flatten, makeθListLinExp[] // Flatten,
   makeθListSpecExp[][[1]][[1]], makeθListSpecExp[][[1]][[2]]},
  Joined → True, PlotRange → {{9, 33}, {-.05, 1.05}},
  PlotStyle → {{RGBColor["#B09771"], AbsoluteThickness[3]},
    {Black, AbsoluteThickness[3]}, {RGBColor["#287DAB"], AbsoluteThickness[3]},
    {RGBColor["#287DAB"], AbsoluteThickness[3]}}, Frame → True,
  ImageSize → Medium, FrameLabel → {Style["Temperature (°C)", 12, Black],
    Style["Heterotrophic investment (θESS)", 12, Black]},
  PlotLegends → {"generalist", "linear", "specialist"},
  FrameTicksStyle → Directive[Black, 12], PlotLabel → Style["KB=3*10^8", 12, Black]]]
```

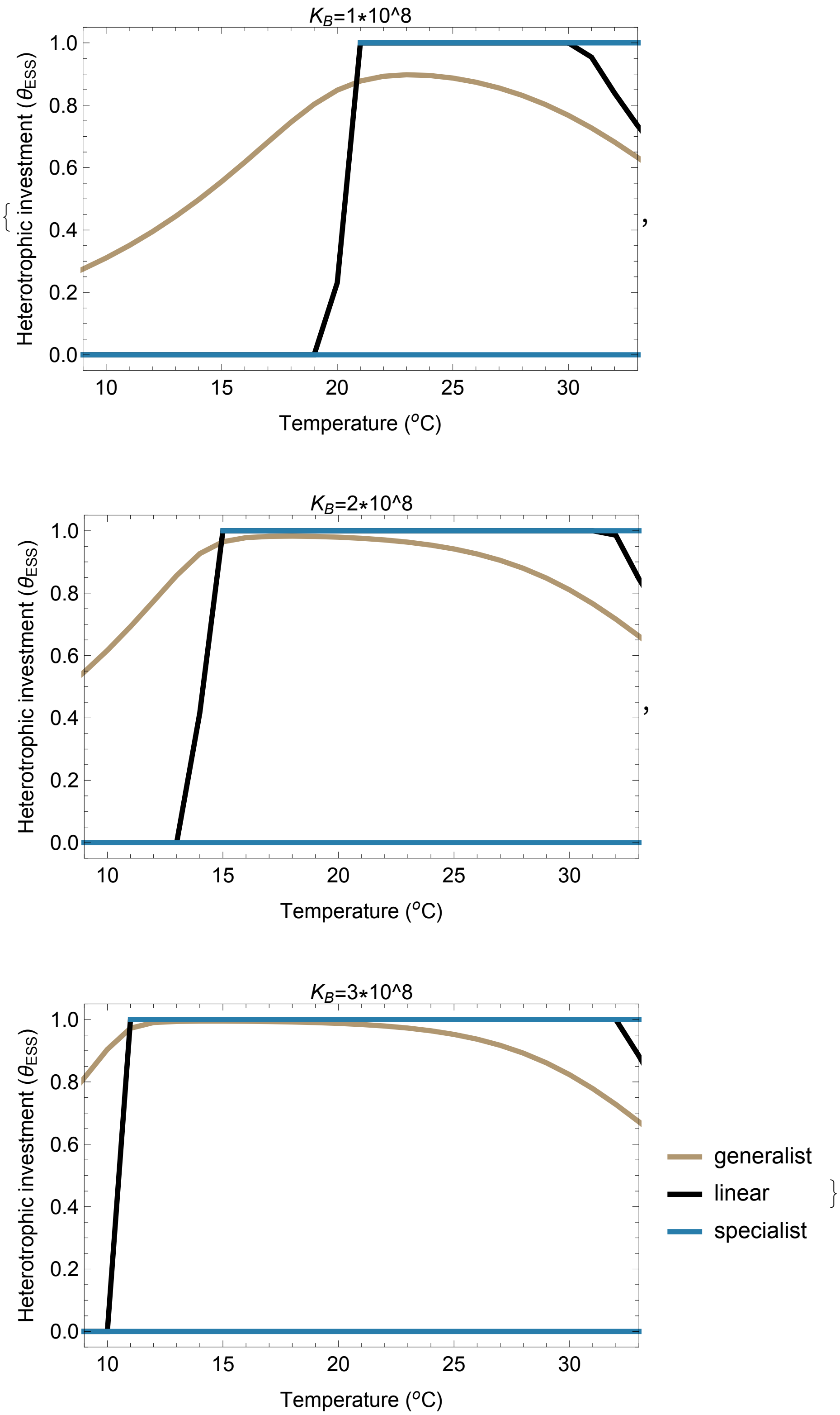
••• Table: Non–list iterator θESSspec0 at position 2 does not evaluate to a real numeric value.

••• Table: Non–list iterator θESSspec0 at position 2 does not evaluate to a real numeric value.

••• Table: Non–list iterator θESSspec0 at position 2 does not evaluate to a real numeric value.

••• General: Further output of Table::nliter will be suppressed during this calculation.

Out[ ]=


K_B=1*10^8
K_B=2*10^8
K_B=3*10^8
Heterotrophic investment (θ_{ESS})
Temperature (^oC)
generalist
linear
specialist

I_{in}=150

In[ ]:=

Linear temperature dependence

In[ ]:=
```
ones = Table[1, 100];
zeros = Table[0, 100];
List[K_B = 1 × 10^8; I_in = 150;
 ListPlot[{makeθListGen[] // Flatten, makeθListLin[] // Flatten,
   makeθListSpec[][[1]][[1]], makeθListSpec[][[1]][[2]]},
  Joined → True, PlotRange → {{9, 33}, {-.05, 1.05}},
  PlotStyle → {{RGBColor["#B09771"], AbsoluteThickness[3]},
    {Black, AbsoluteThickness[3]}, {RGBColor["#287DAB"], AbsoluteThickness[3]},
    {RGBColor["#287DAB"], AbsoluteThickness[3]}}, Frame → True,
  ImageSize → Medium, FrameLabel → {Style["Temperature (°C)", 12, Black],
    Style["Heterotrophic investment (θ_ESS)", 12, Black]},
  FrameTicksStyle → Directive[Black, 12],
  PlotLabel → Style["K_B=1*10^8", 12, Black]], K_B = 2 × 10^8;
 I_in = 150;
 ListPlot[{makeθListGen[] // Flatten, makeθListLin[] // Flatten,
   makeθListSpec[][[1]][[1]], makeθListSpec[][[1]][[2]]},
  Joined → True, PlotRange → {{9, 33}, {-.05, 1.05}},
  PlotStyle → {{RGBColor["#B09771"], AbsoluteThickness[3]},
    {Black, AbsoluteThickness[3]}, {RGBColor["#287DAB"], AbsoluteThickness[3]},
    {RGBColor["#287DAB"], AbsoluteThickness[3]}}, Frame → True,
  ImageSize → Medium, FrameLabel → {Style["Temperature (°C)", 12, Black],
    Style["Heterotrophic investment (θ_ESS)", 12, Black]},
  FrameTicksStyle → Directive[Black, 12],
  PlotLabel → Style["K_B=2*10^8", 12, Black]], K_B = 3 × 10^8;
 I_in = 150;
 ListPlot[{makeθListGen[] // Flatten, makeθListLin[] // Flatten,
   makeθListSpec[][[1]][[1]], makeθListSpec[][[1]][[2]]},
  Joined → True, PlotRange → {{9, 33}, {-.05, 1.05}},
  PlotStyle → {{RGBColor["#B09771"], AbsoluteThickness[3]},
    {Black, AbsoluteThickness[3]}, {RGBColor["#287DAB"], AbsoluteThickness[3]},
    {RGBColor["#287DAB"], AbsoluteThickness[3]}}, Frame → True,
  ImageSize → Medium, FrameLabel → {Style["Temperature (°C)", 12, Black],
    Style["Heterotrophic investment (θ_ESS)", 12, Black]},
  PlotLegends → {"generalist", "linear", "specialist"},
  FrameTicksStyle → Directive[Black, 12], PlotLabel → Style["K_B=3*10^8", 12, Black]]]
```

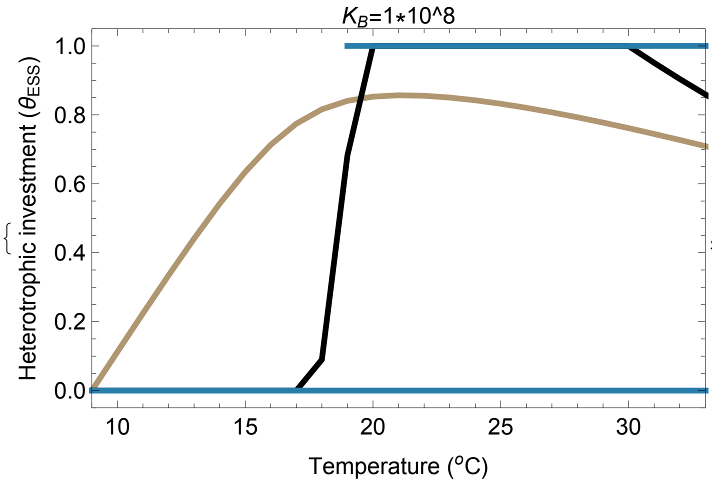
··· Table: Non−list iterator θESSspec0 at position 2 does not evaluate to a real numeric value.

••• Table: Non−list iterator θESSspec0 at position 2 does not evaluate to a real numeric value.

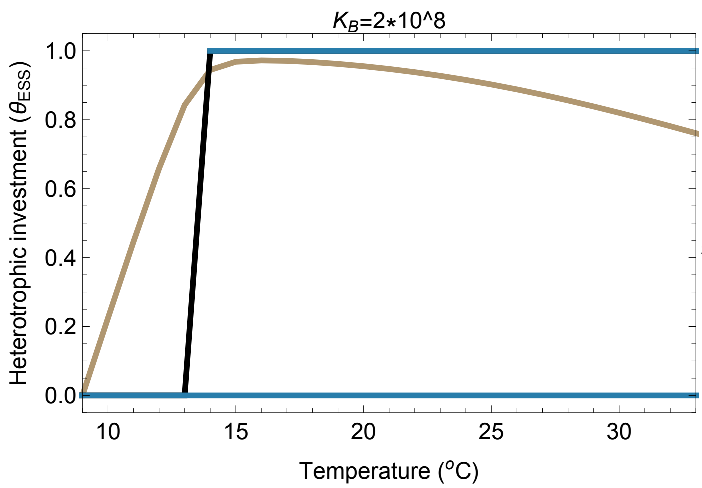
••• Table: Non−list iterator θESSspec0 at position 2 does not evaluate to a real numeric value.

••• General: Further output of Table::nliter will be suppressed during this calculation.

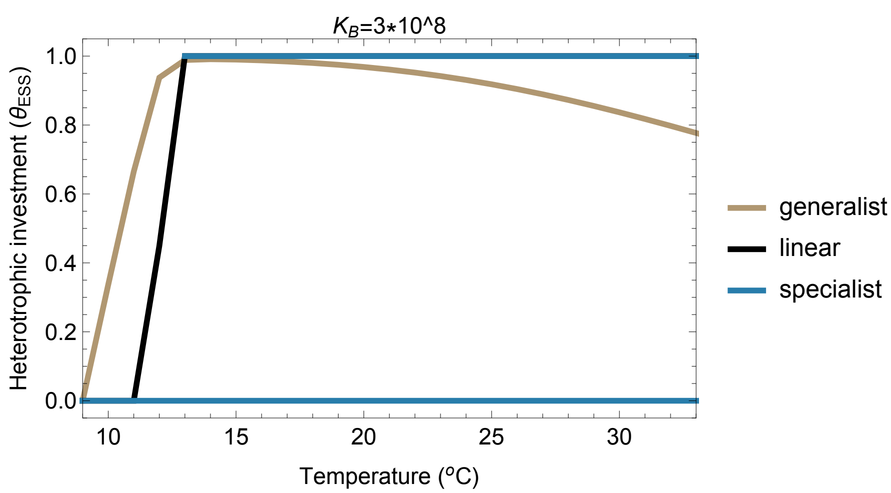
Out[•]= {



,



,



}

Exponential temperature dependence

In[∘]:=
```
List[KB = 1 × 10^8; Iin = 150;
 ListPlot[{makeθListGenExp[] // Flatten, makeθListLinExp[] // Flatten,
   makeθListSpecExp[][[1]][[1]], makeθListSpecExp[][[1]][[2]]},
  Joined → True, PlotRange → {{9, 33}, {-.05, 1.05}},
  PlotStyle → {{RGBColor["#B09771"], AbsoluteThickness[3]},
    {Black, AbsoluteThickness[3]}, {RGBColor["#287DAB"], AbsoluteThickness[3]},
    {RGBColor["#287DAB"], AbsoluteThickness[3]}}, Frame → True,
  ImageSize → Medium, FrameLabel → {Style["Temperature (°C)", 12, Black],
    Style["Heterotrophic investment (θESS)", 12, Black]},
  FrameTicksStyle → Directive[Black, 12],
  PlotLabel → Style["KB=1*10^8", 12, Black]], KB = 2 × 10^8;
 Iin = 150;
 ListPlot[{makeθListGenExp[] // Flatten, makeθListLinExp[] // Flatten,
   makeθListSpecExp[][[1]][[1]], makeθListSpecExp[][[1]][[2]]},
  Joined → True, PlotRange → {{9, 33}, {-.05, 1.05}},
  PlotStyle → {{RGBColor["#B09771"], AbsoluteThickness[3]},
    {Black, AbsoluteThickness[3]}, {RGBColor["#287DAB"], AbsoluteThickness[3]},
    {RGBColor["#287DAB"], AbsoluteThickness[3]}}, Frame → True,
  ImageSize → Medium, FrameLabel → {Style["Temperature (°C)", 12, Black],
    Style["Heterotrophic investment (θESS)", 12, Black]},
  FrameTicksStyle → Directive[Black, 12],
  PlotLabel → Style["KB=2*10^8", 12, Black]], KB = 3 × 10^8;
 Iin = 150;
 ListPlot[{makeθListGenExp[] // Flatten, makeθListLinExp[] // Flatten,
   makeθListSpecExp[][[1]][[1]], makeθListSpecExp[][[1]][[2]]},
  Joined → True, PlotRange → {{9, 33}, {-.05, 1.05}},
  PlotStyle → {{RGBColor["#B09771"], AbsoluteThickness[3]},
    {Black, AbsoluteThickness[3]}, {RGBColor["#287DAB"], AbsoluteThickness[3]},
    {RGBColor["#287DAB"], AbsoluteThickness[3]}}, Frame → True,
  ImageSize → Medium, FrameLabel → {Style["Temperature (°C)", 12, Black],
    Style["Heterotrophic investment (θESS)", 12, Black]},
  PlotLegends → {"generalist", "linear", "specialist"},
  FrameTicksStyle → Directive[Black, 12], PlotLabel → Style["KB=3*10^8", 12, Black]]]
```

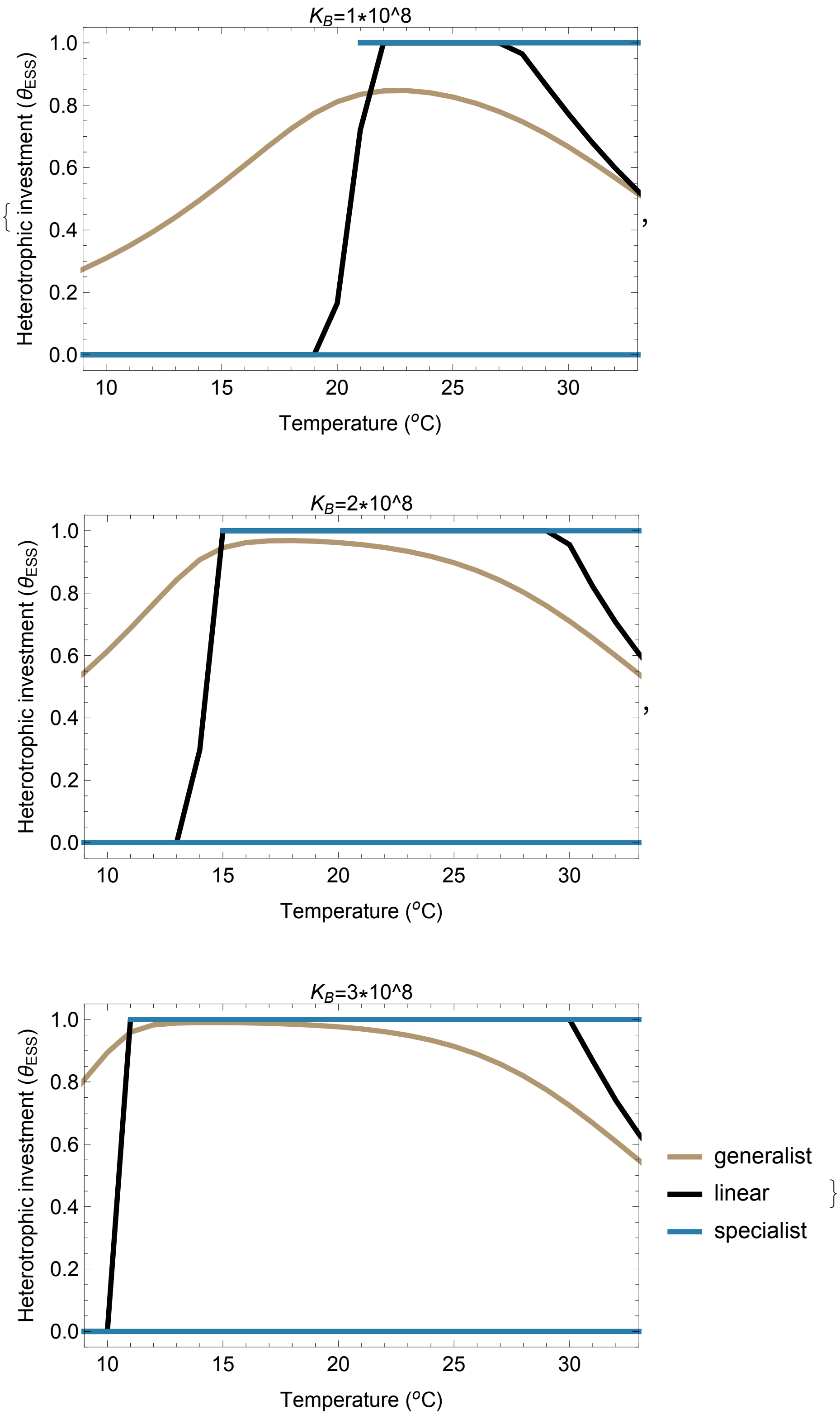
••• Table: Non−list iterator θESSspec0 at position 2 does not evaluate to a real numeric value.

••• Table: Non−list iterator θESSspec0 at position 2 does not evaluate to a real numeric value.

••• Table: Non−list iterator θESSspec0 at position 2 does not evaluate to a real numeric value.

••• General: Further output of Table::nliter will be suppressed during this calculation.

Out[]=



K_B=1*10^8
K_B=2*10^8
K_B=3*10^8
Heterotrophic investment (θ_{ESS})
1.0
0.8
0.6
0.4
0.2
0.0
10
15
20
25
30
Temperature (^{o}C)
generalist
linear
specialist

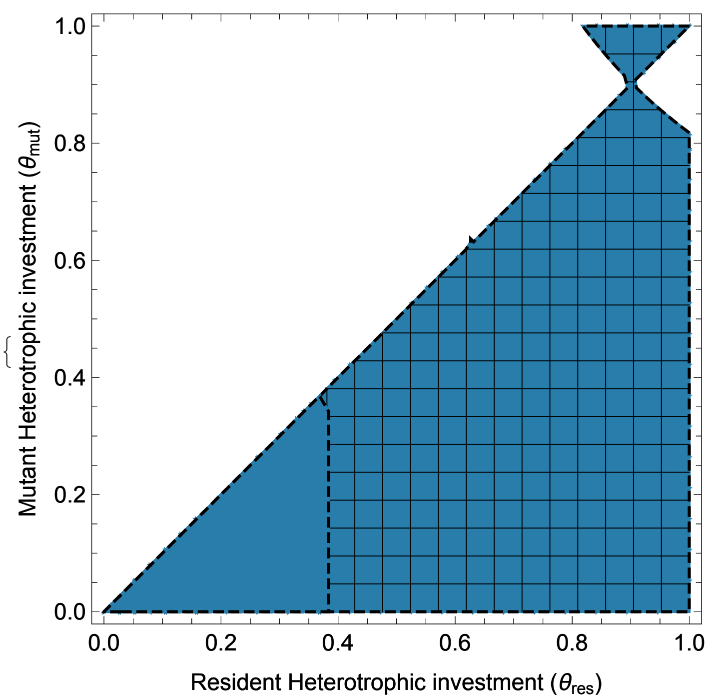
Pairwise invasibility plots

In[∘]:=

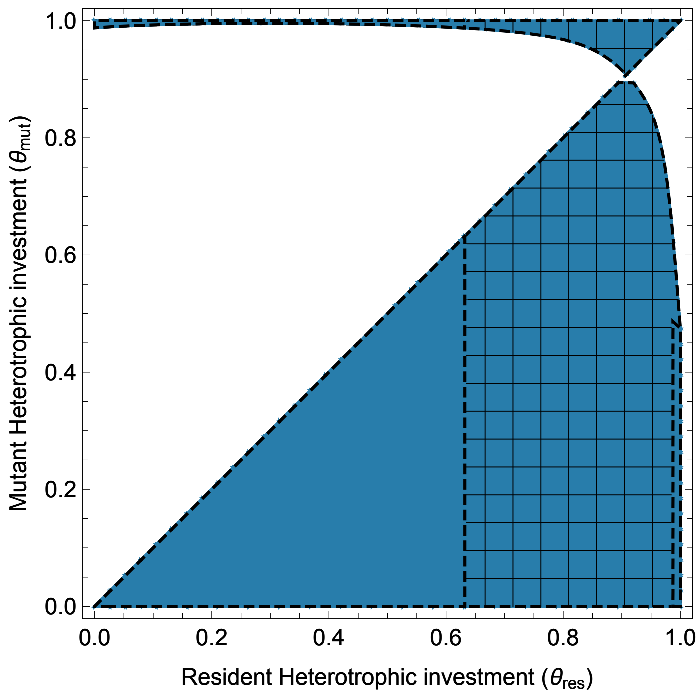
K_B=1*10^8, I_{in}=100

```
In[∘]:= KB = 1 × 10^8; Iin = 100;
Quiet[List[Overlay[{MakePIPNV[-1, T0, RGBColor["#287DAB"], Black],
     MakePIP[-1, T0, RGBColor["#287DAB"]]}],
   Overlay[{MakePIPNV[-1, T0 + 5, RGBColor["#287DAB"], Black],
     MakePIP[-1, T0 + 5, RGBColor["#287DAB"]]}],
   Overlay[{MakePIPNV[-1, T0 + 10, RGBColor["#287DAB"], Black],
     MakePIP[-1, T0 + 10, RGBColor["#287DAB"]]}]]]
Quiet[List[Overlay[{MakePIPNV[0, T0, Black, White], MakePIP[0, T0, Black]}],
   Overlay[{MakePIPNV[0, T0 + 5, Black, White], MakePIP[0, T0 + 5, Black]}], Overlay[
    {MakePIPNV[0, T0 + 10, Black, White], MakePIP[0, T0 + 10, Black]}]]] (*linear*)
Quiet[List[Overlay[{MakePIPNV[1, T0, RGBColor["#B09771"], Black],
     MakePIP[1, T0, RGBColor["#B09771"]]}],
   Overlay[{MakePIPNV[1, T0 + 5, RGBColor["#B09771"], Black],
     MakePIP[1, T0 + 5, RGBColor["#B09771"]]}],
   Overlay[{MakePIPNV[1, T0 + 10, RGBColor["#B09771"], Black],
     MakePIP[1, T0 + 10, RGBColor["#B09771"]]}]]] (*generalist*)
```

Out[∘]= {

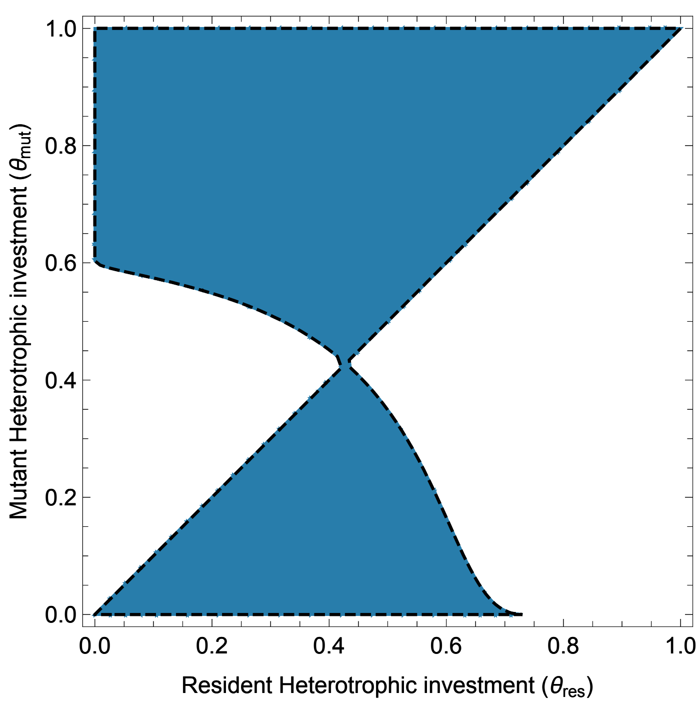


,



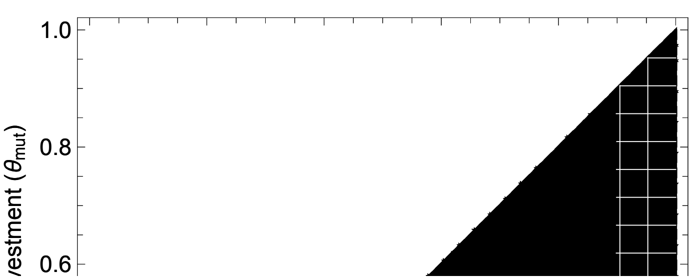
Mutant Heterotrophic investment (θ_{mut})
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

,

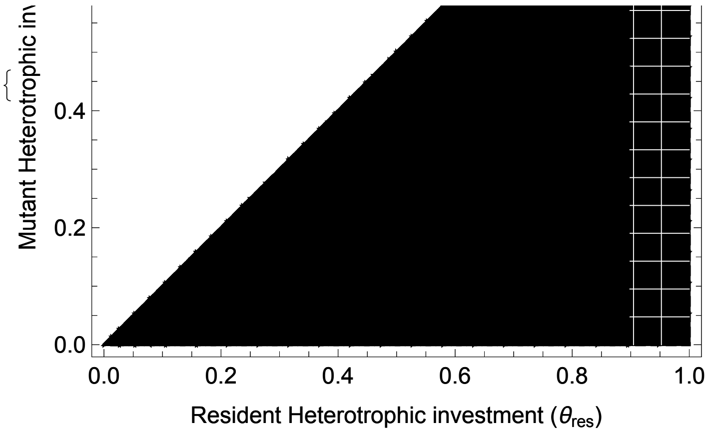
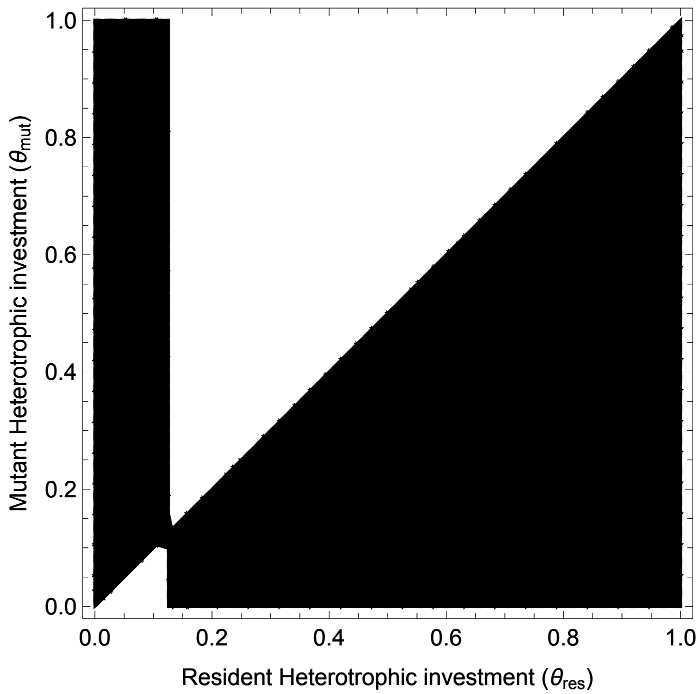
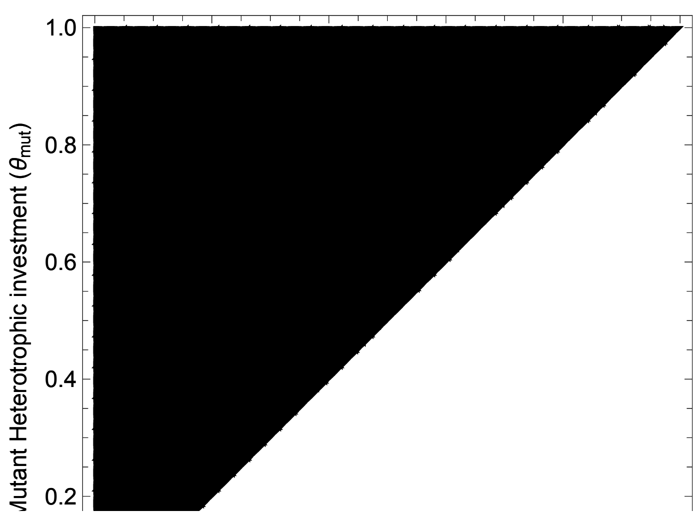


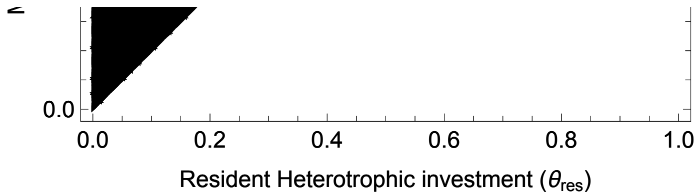
Mutant Heterotrophic investment (θ_{mut})
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

}

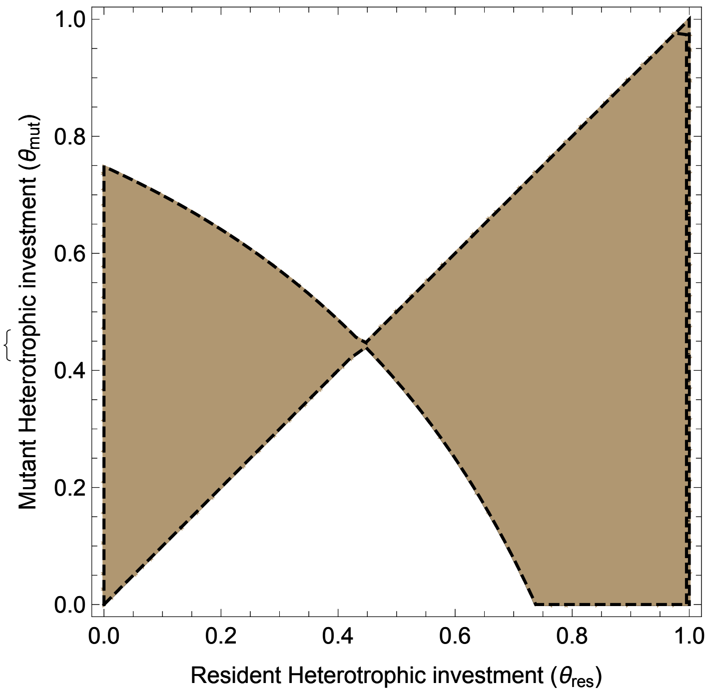


vestment (θ_{mut})
1.0
0.8
0.6

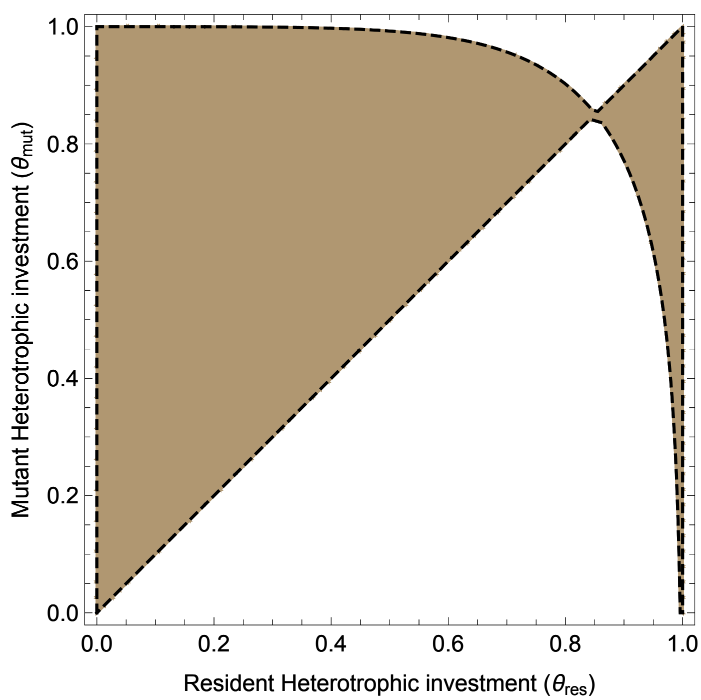
Out[●]= {
Mutant Heterotrophic in
0.0
0.2
0.4
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})
,

Mutant Heterotrophic investment (θ_{mut})
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})
,

Mutant Heterotrophic investment (θ_{mut})
0.2
0.4
0.6
0.8
1.0
}


0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

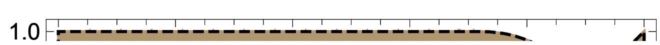
Out[●]= {

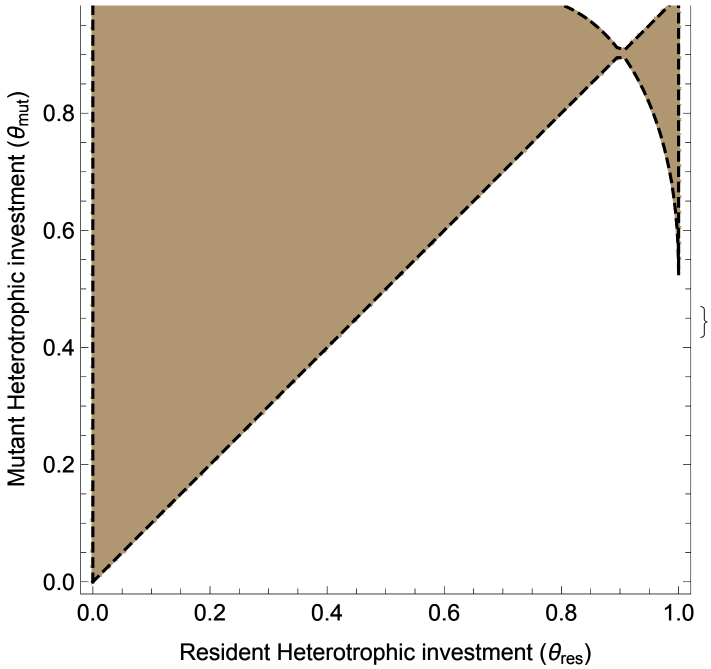

1.0
0.8
0.6
0.4
0.2
0.0
Mutant Heterotrophic investment (θ_{mut})
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

,


1.0
0.8
0.6
0.4
0.2
0.0
Mutant Heterotrophic investment (θ_{mut})
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

,


1.0

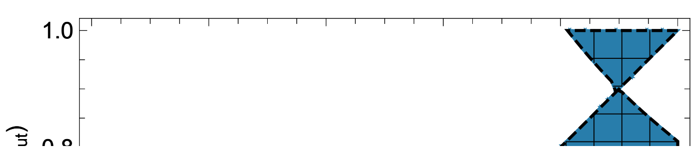


In[•]:=

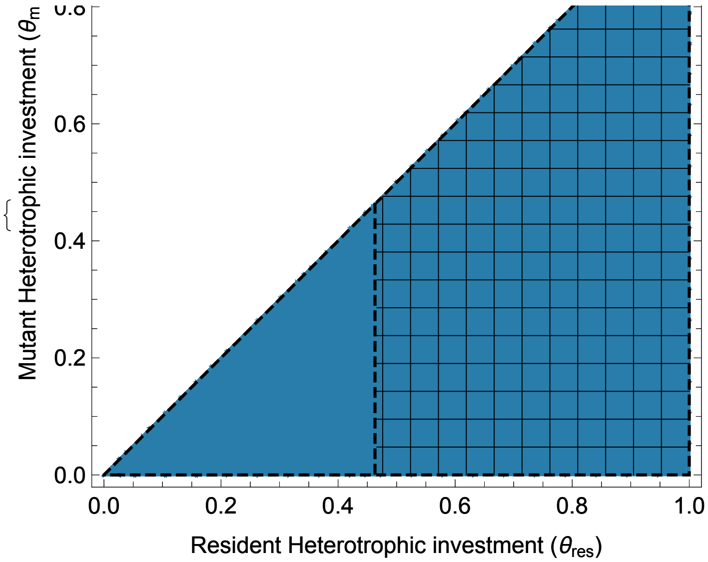
In[•]:=

K_B=2*10^8, I_{in}=100

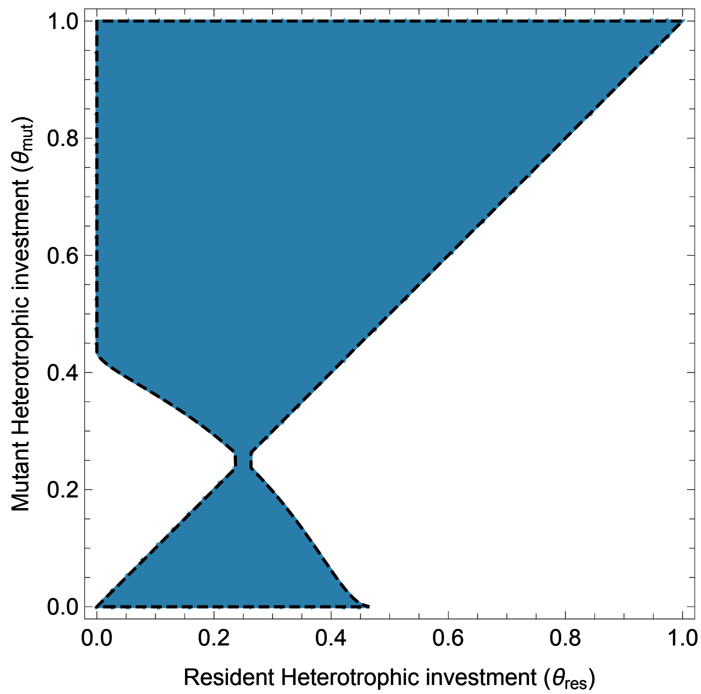
In[•]:=
```
KB = 2 × 10 ^ 8; Iin = 100;
Quiet[List[Overlay[{MakePIPNV[-1, T0, RGBColor["#287DAB"], Black],
    MakePIP[-1, T0, RGBColor["#287DAB"]]}],
  Overlay[{MakePIPNV[-1, T0 + 5, RGBColor["#287DAB"], Black],
    MakePIP[-1, T0 + 5, RGBColor["#287DAB"]]}],
  Overlay[{MakePIPNV[-1, T0 + 10, RGBColor["#287DAB"], Black],
    MakePIP[-1, T0 + 10, RGBColor["#287DAB"]]}]]]
Quiet[List[Overlay[{MakePIPNV[0, T0, Black, White], MakePIP[0, T0, Black]}],
  Overlay[{MakePIPNV[0, T0 + 5, Black, White], MakePIP[0, T0 + 5, Black]}], Overlay[
   {MakePIPNV[0, T0 + 10, Black, White], MakePIP[0, T0 + 10, Black]}]]](*linear*)
Quiet[List[Overlay[{MakePIPNV[1, T0, RGBColor["#B09771"], Black],
    MakePIP[1, T0, RGBColor["#B09771"]]}],
  Overlay[{MakePIPNV[1, T0 + 5, RGBColor["#B09771"], Black],
    MakePIP[1, T0 + 5, RGBColor["#B09771"]]}],
  Overlay[{MakePIPNV[1, T0 + 10, RGBColor["#B09771"], Black],
    MakePIP[1, T0 + 10, RGBColor["#B09771"]]}]]](*generalist*)
```



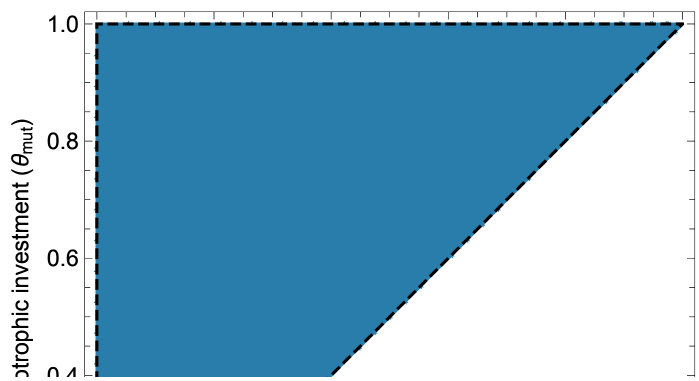
Out[•]= {


Mutant Heterotrophic investment (θ_{m}
0.0
0.2
0.4
0.6
0.8
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

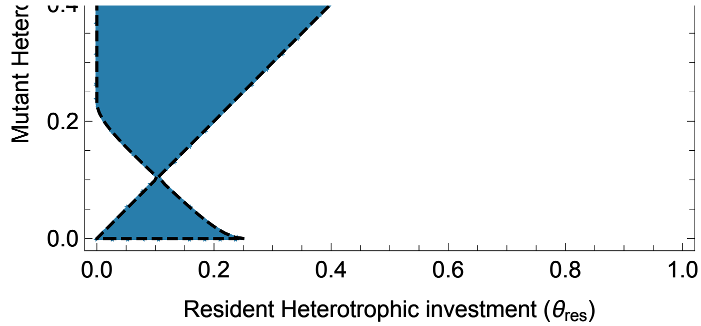
,


Mutant Heterotrophic investment (θ_{mut})
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

,

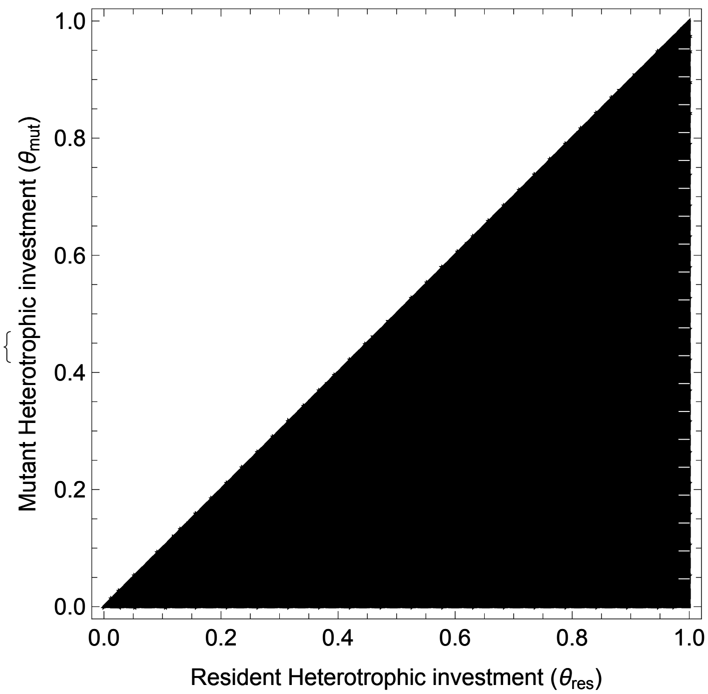

otrophic investment (θ_{mut})
0.4
0.6
0.8
1.0

}



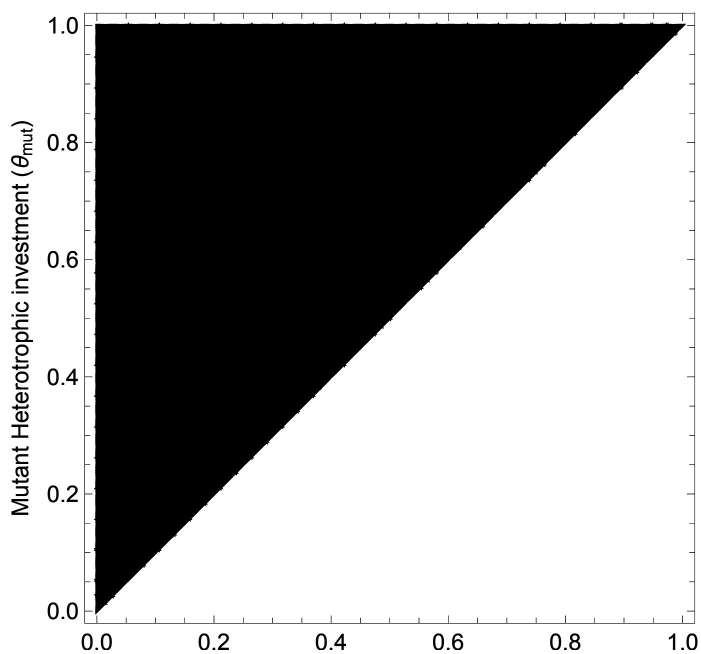
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

Out[•]= {



Mutant Heterotrophic investment (θ_{mut})
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

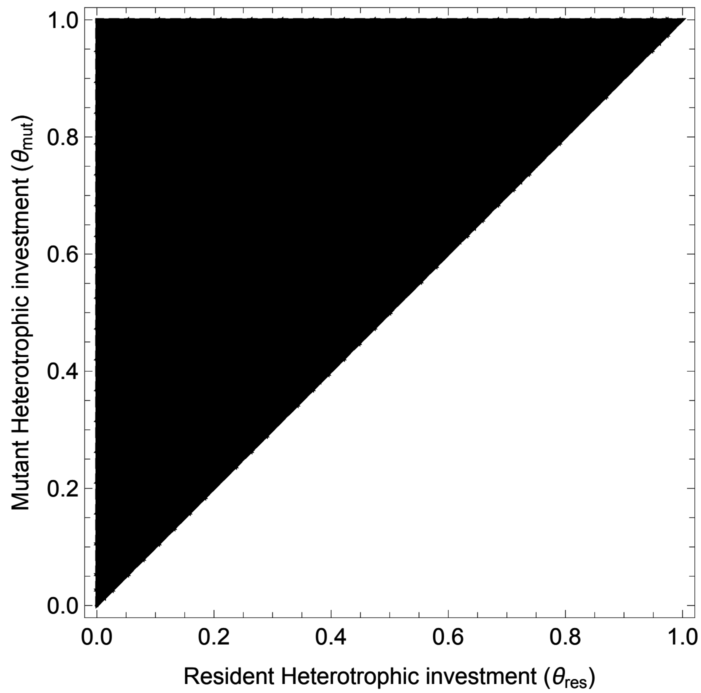
,



Mutant Heterotrophic investment (θ_{mut})
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

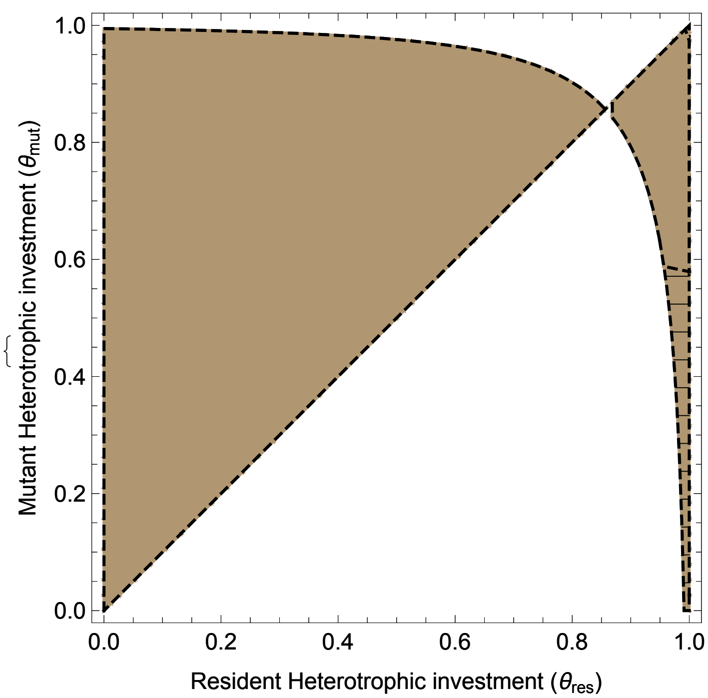
,

Resident Heterotrophic investment (θ_{res})

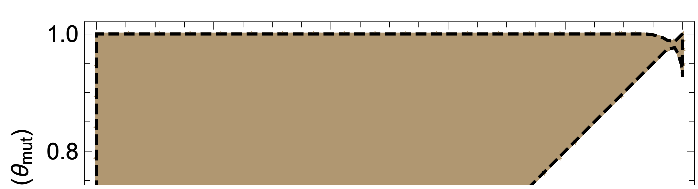


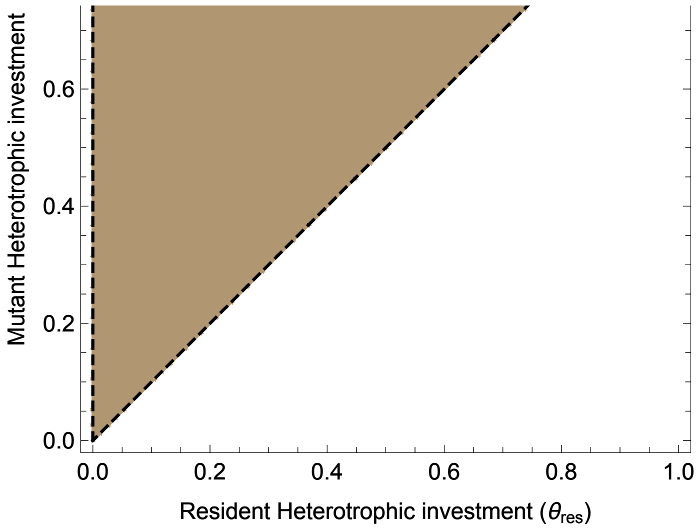
}

Out[•]= {

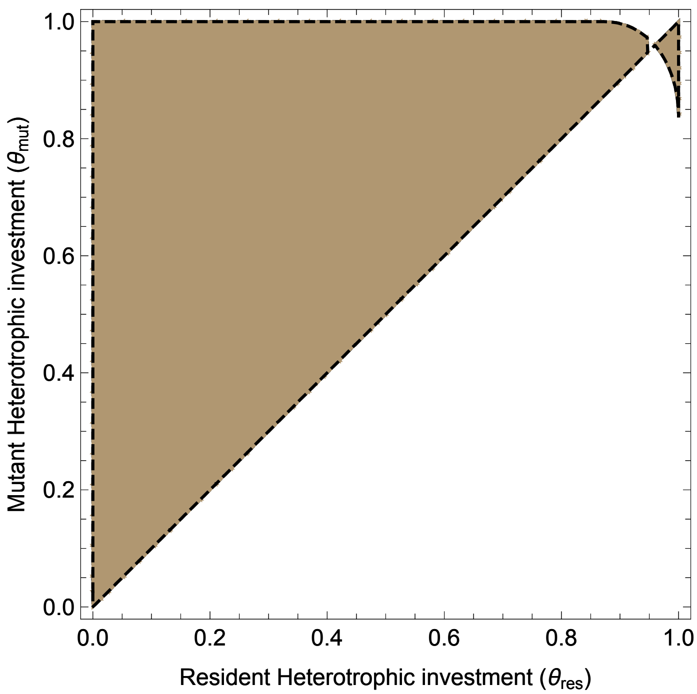


,





,

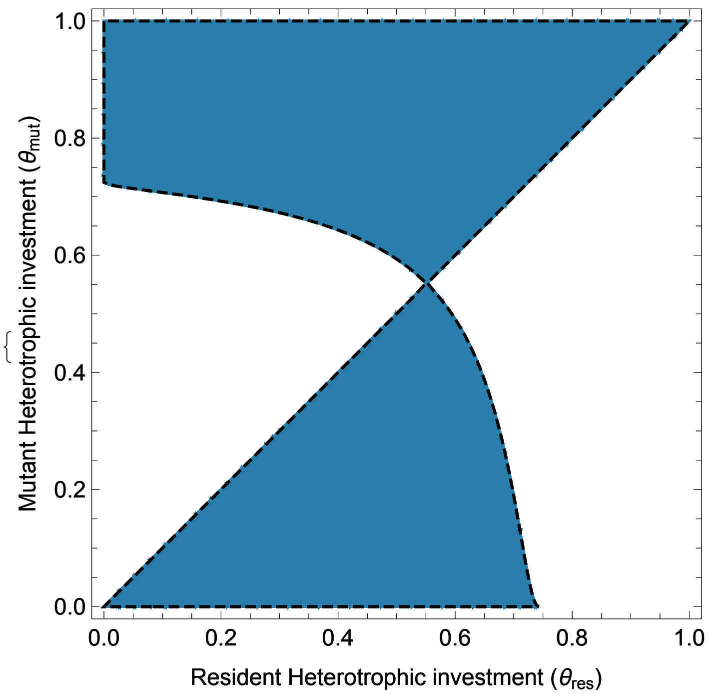


}

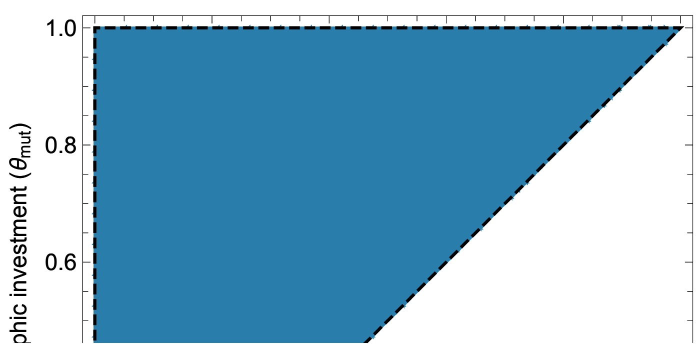
K_B=3*10^8, I_{in}=100

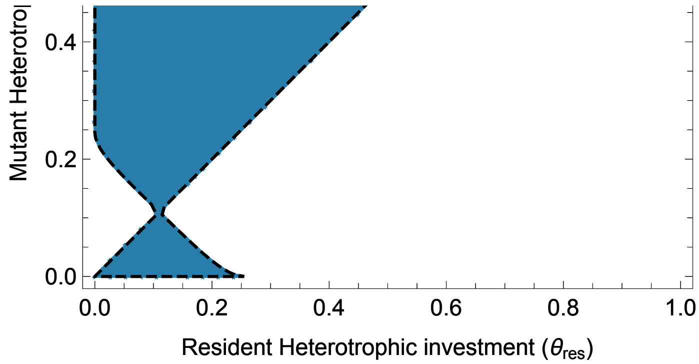
In[∘]:=
```
K_B = 3 × 10^8; I_in = 100;
Quiet[List[Overlay[{MakePIPNV[-1, T0, RGBColor["#287DAB"], Black],
    MakePIP[-1, T0, RGBColor["#287DAB"]]}],
  Overlay[{MakePIPNV[-1, T0 + 5, RGBColor["#287DAB"], Black],
    MakePIP[-1, T0 + 5, RGBColor["#287DAB"]]}],
  Overlay[{MakePIPNV[-1, T0 + 10, RGBColor["#287DAB"], Black],
    MakePIP[-1, T0 + 10, RGBColor["#287DAB"]]}]]]
Quiet[List[Overlay[{MakePIPNV[0, T0, Black, White], MakePIP[0, T0, Black]}],
  Overlay[{MakePIPNV[0, T0 + 5, Black, White], MakePIP[0, T0 + 5, Black]}], Overlay[
   {MakePIPNV[0, T0 + 10, Black, White], MakePIP[0, T0 + 10, Black]}]]] (*linear*)
Quiet[List[Overlay[{MakePIPNV[1, T0, RGBColor["#B09771"], Black],
    MakePIP[1, T0, RGBColor["#B09771"]]}],
  Overlay[{MakePIPNV[1, T0 + 5, RGBColor["#B09771"], Black],
    MakePIP[1, T0 + 5, RGBColor["#B09771"]]}],
  Overlay[{MakePIPNV[1, T0 + 10, RGBColor["#B09771"], Black],
    MakePIP[1, T0 + 10, RGBColor["#B09771"]]}]]] (*generalist*)
```

Out[∘]= {

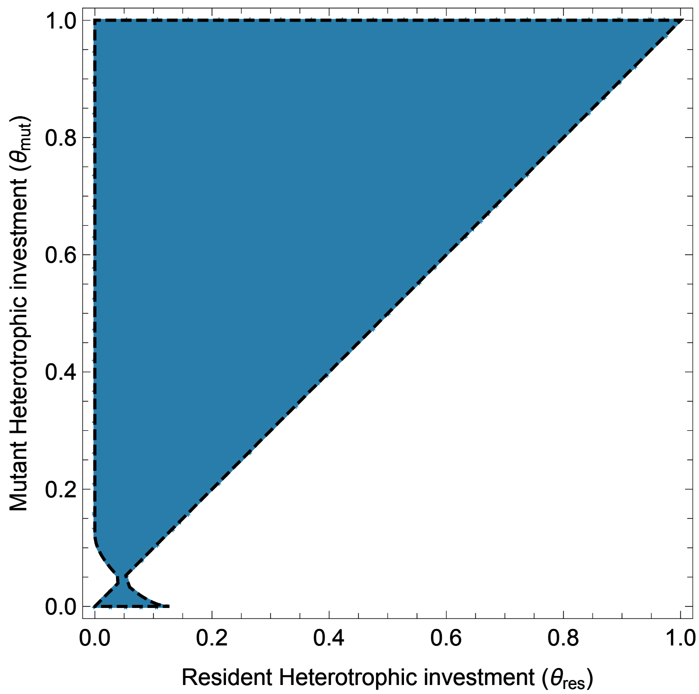


,



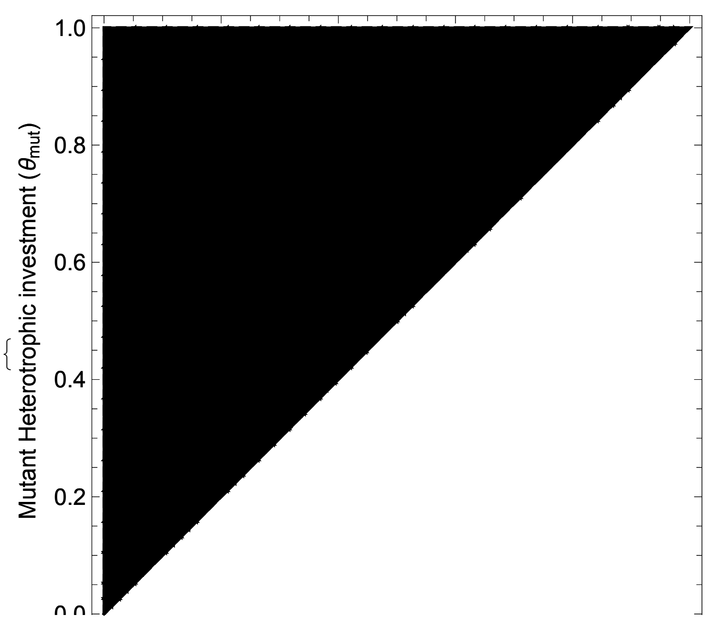

Mutant Heterotrop
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

,

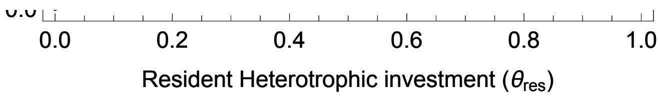

Mutant Heterotrophic investment (θ_{mut})
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

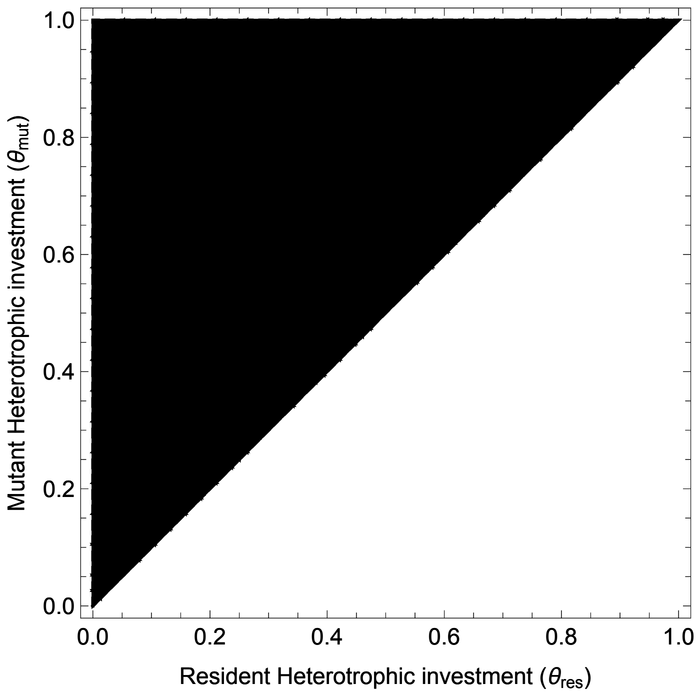
}

Out[ ]= {

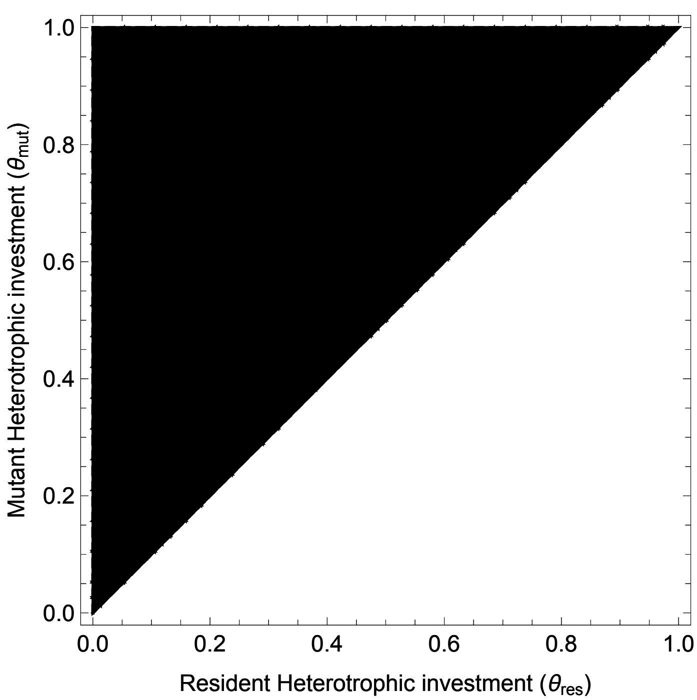

Mutant Heterotrophic investment (θ_{mut})
1.0
0.8
0.6
0.4
0.2

,

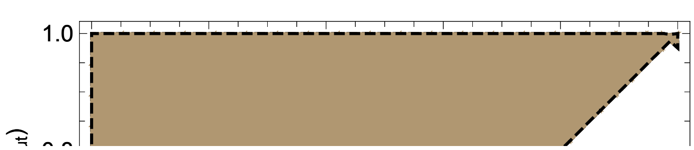

0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

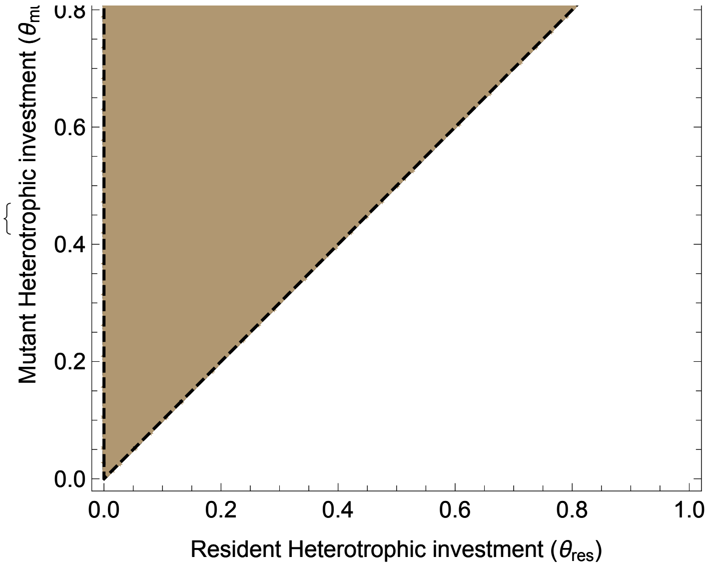

Mutant Heterotrophic investment (θ_{mut})
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

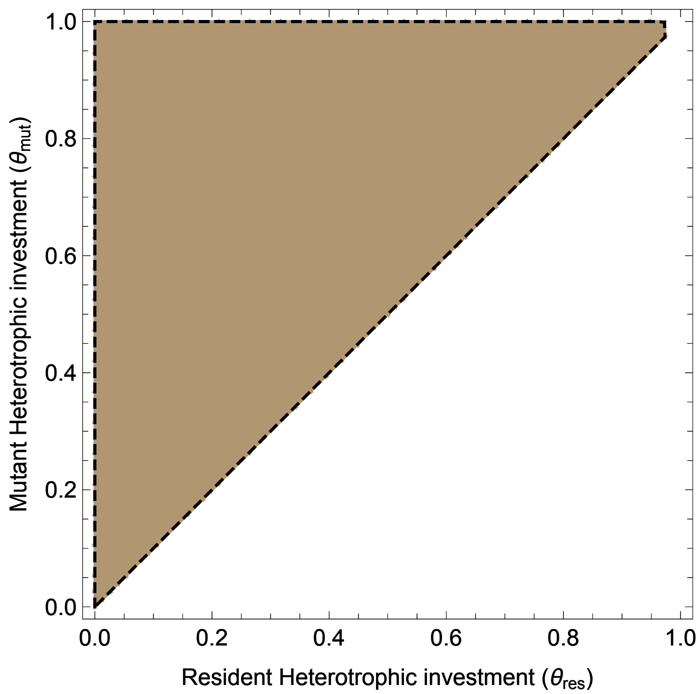
,

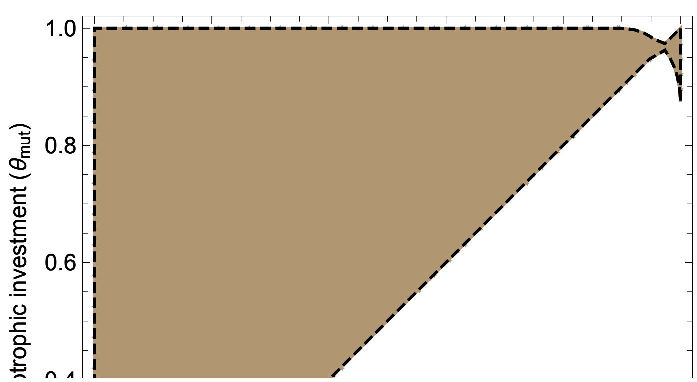

Mutant Heterotrophic investment (θ_{mut})
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

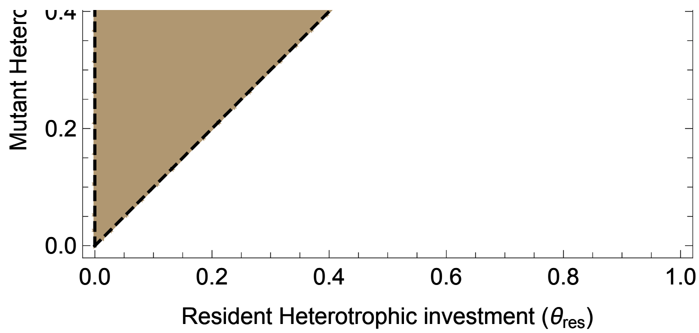
}


1.0

Out[•]= {
0.0
0.2
0.4
0.6
0.8
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})
Mutant Heterotrophic investment (θ_{mu}
,


0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})
Mutant Heterotrophic investment (θ_{mut})
,

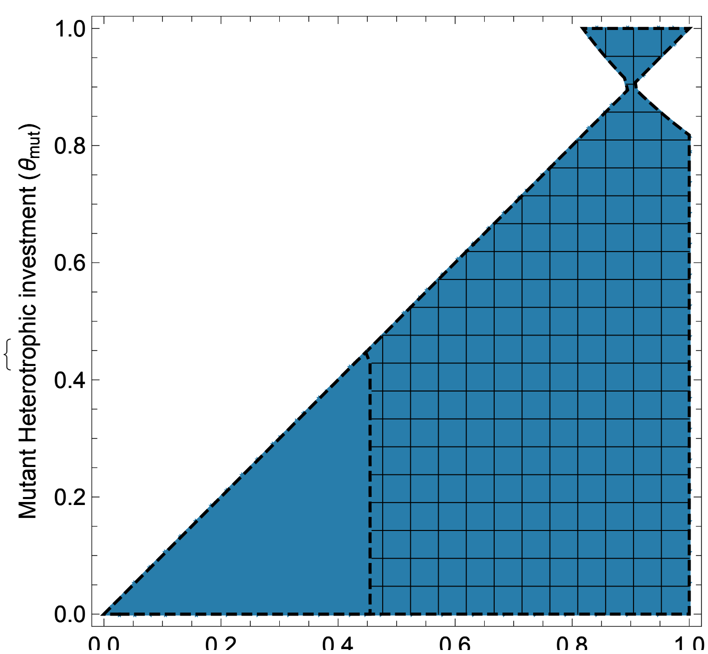

0.6
0.8
1.0
trophic investment (θ_{mut})
}



K_B=1*10^8, I_{in}=150

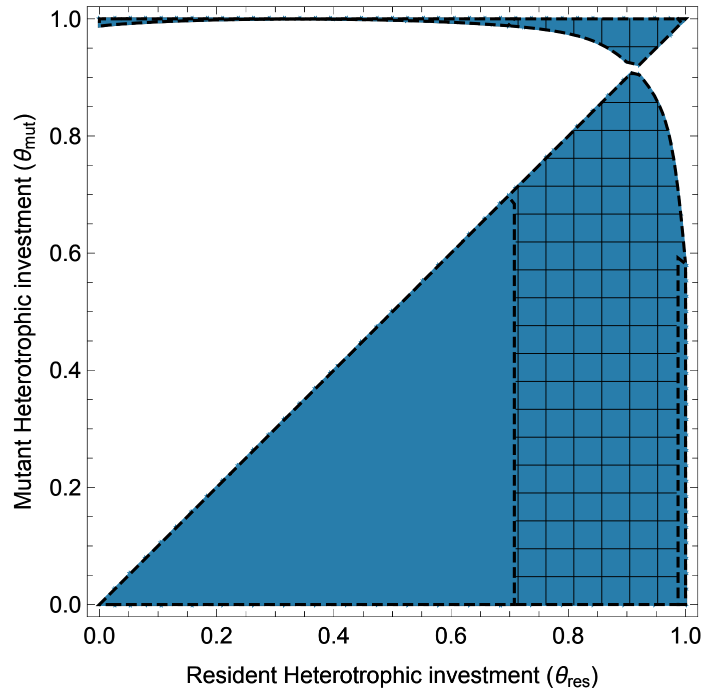
```
In[*]:= K_B = 1 × 10^8; I_in = 150;
Quiet[List[Overlay[{MakePIPNV[-1, T0, RGBColor["#287DAB"], Black],
     MakePIP[-1, T0, RGBColor["#287DAB"]]}],
   Overlay[{MakePIPNV[-1, T0 + 5, RGBColor["#287DAB"], Black],
     MakePIP[-1, T0 + 5, RGBColor["#287DAB"]]}],
   Overlay[{MakePIPNV[-1, T0 + 10, RGBColor["#287DAB"], Black],
     MakePIP[-1, T0 + 10, RGBColor["#287DAB"]]}]]]
Quiet[List[Overlay[{MakePIPNV[0, T0, Black, White], MakePIP[0, T0, Black]}],
   Overlay[{MakePIPNV[0, T0 + 5, Black, White], MakePIP[0, T0 + 5, Black]}], Overlay[
    {MakePIPNV[0, T0 + 10, Black, White], MakePIP[0, T0 + 10, Black]}]]](*linear*)
Quiet[List[Overlay[{MakePIPNV[1, T0, RGBColor["#B09771"], Black],
     MakePIP[1, T0, RGBColor["#B09771"]]}],
   Overlay[{MakePIPNV[1, T0 + 5, RGBColor["#B09771"], Black],
     MakePIP[1, T0 + 5, RGBColor["#B09771"]]}],
   Overlay[{MakePIPNV[1, T0 + 10, RGBColor["#B09771"], Black],
     MakePIP[1, T0 + 10, RGBColor["#B09771"]]}]]](*generalist*)
```

Out[*]= {

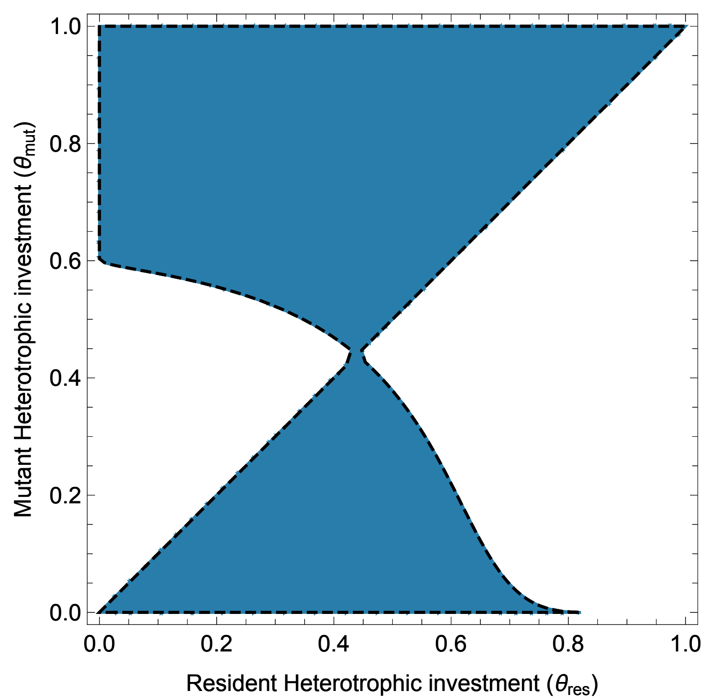


,

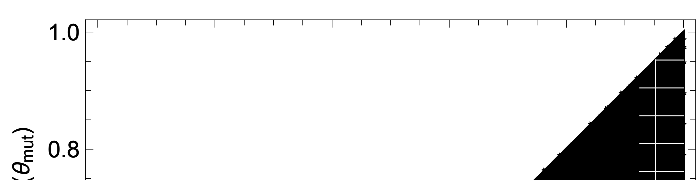

Resident Heterotrophic investment (θ_{res})


1.0
0.8
0.6
0.4
0.2
0.0
Mutant Heterotrophic investment (θ_{mut})
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

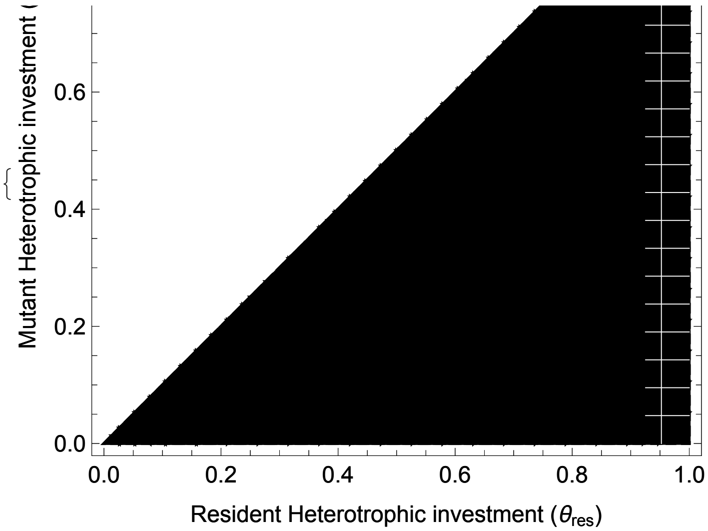
,


1.0
0.8
0.6
0.4
0.2
0.0
Mutant Heterotrophic investment (θ_{mut})
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

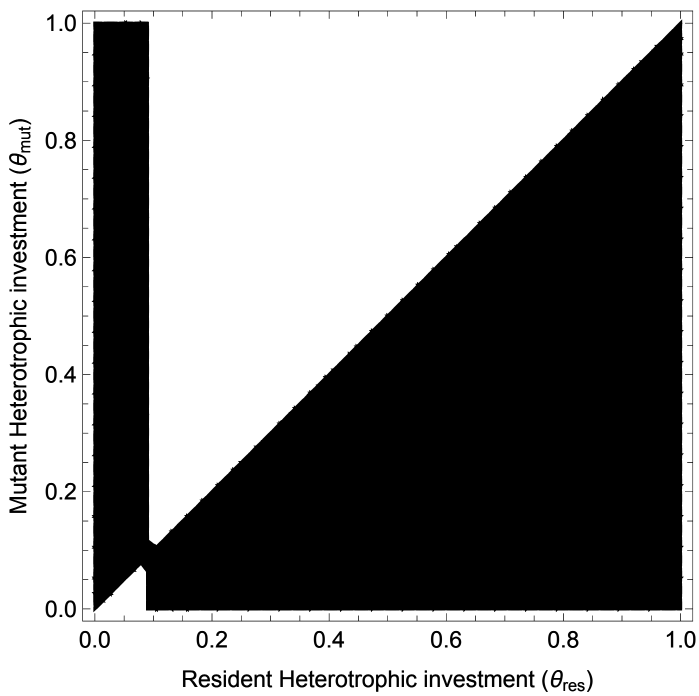
}


1.0
0.8
(θ_{mut})

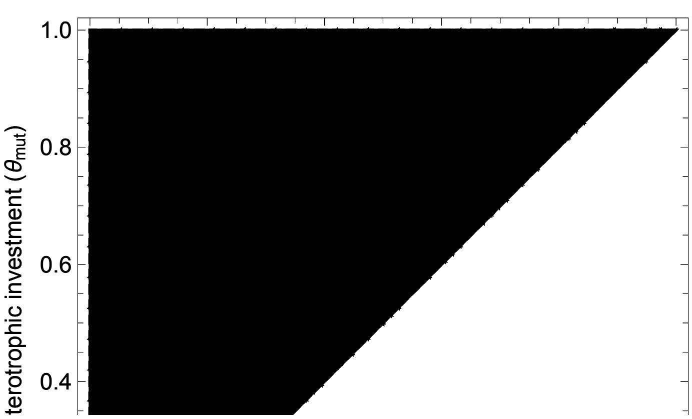
Out[ ]= {


Mutant Heterotrophic investment (
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

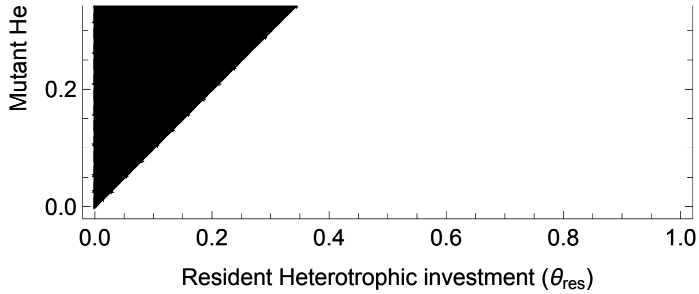
,


1.0
0.8
0.6
0.4
0.2
0.0
Mutant Heterotrophic investment (θ_{mut})
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

,

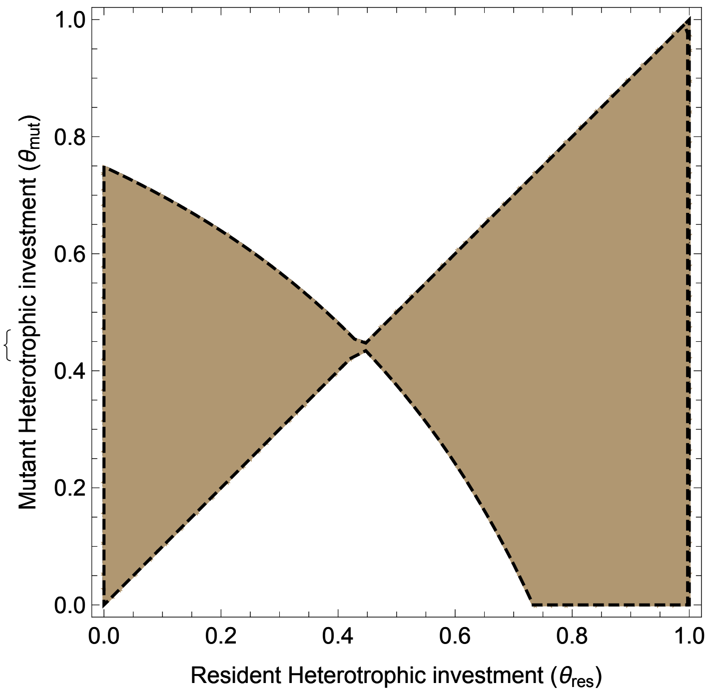

1.0
0.8
0.6
0.4
terotrophic investment (θ_{mut})

}



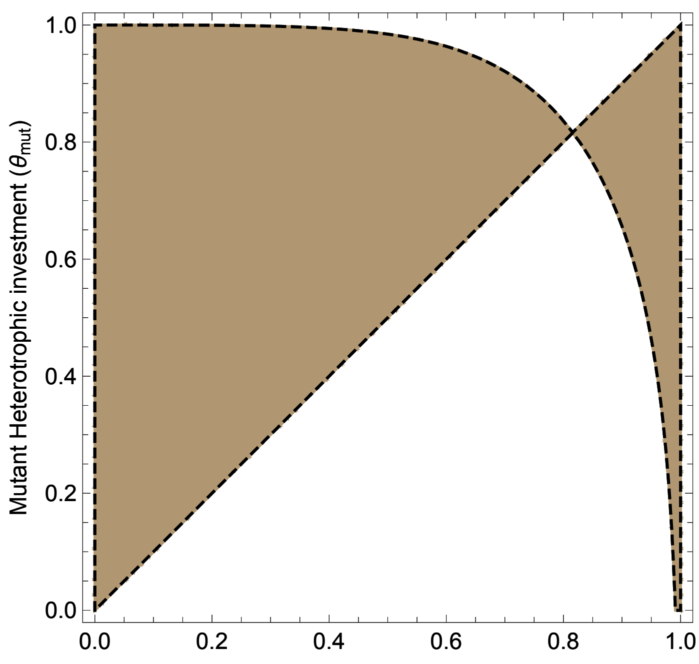
Mutant He
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

Out[ ]= {



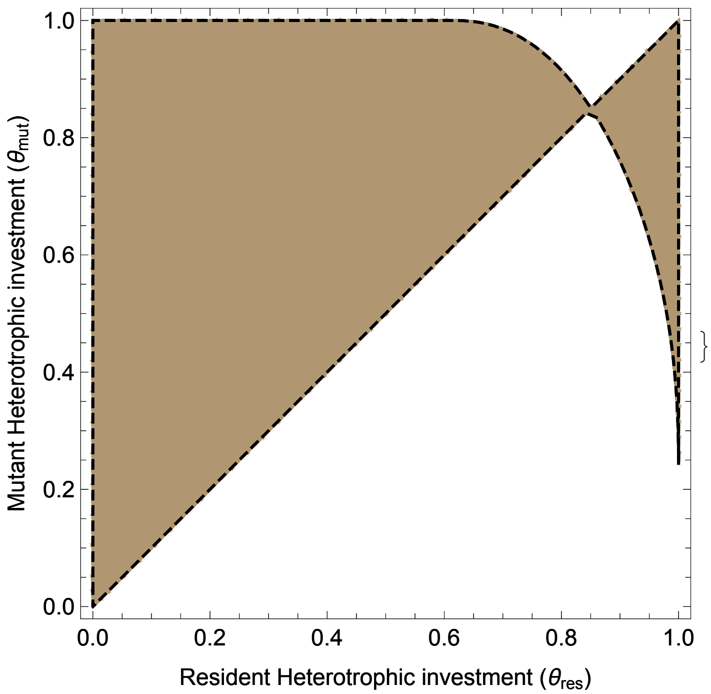
Mutant Heterotrophic investment (θ_{mut})
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

,



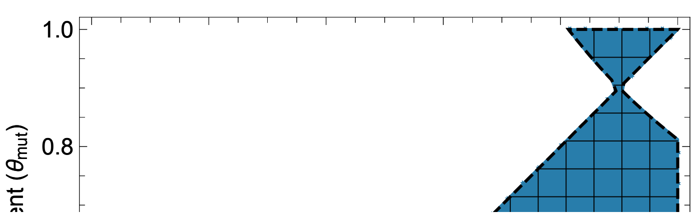
Mutant Heterotrophic investment (θ_{mut})
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

,

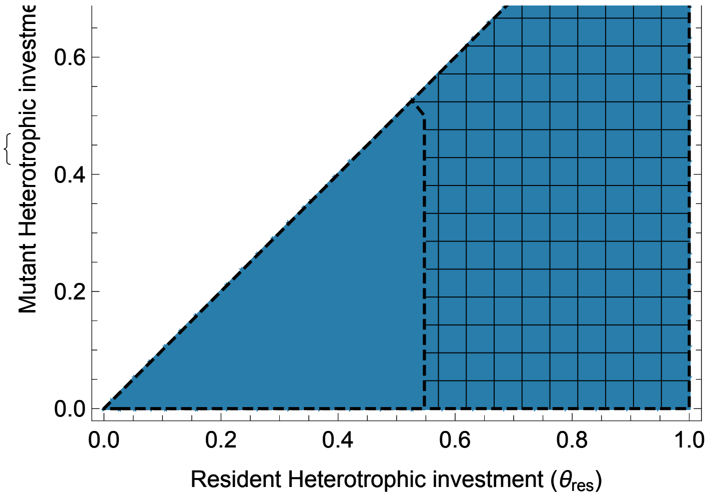


K_B=2*10^8, I_{in}=150

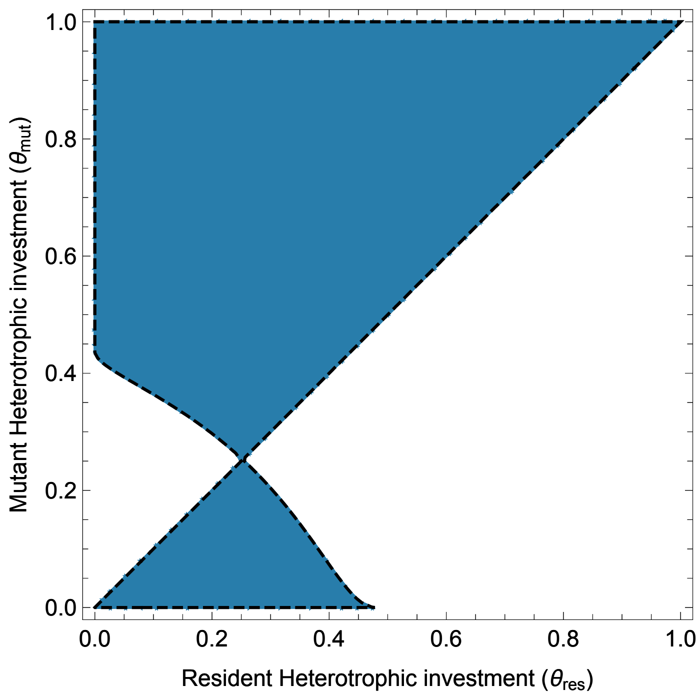
In[∘]:=
```
K_B = 2 × 10^8; I_in = 150;
Quiet[List[Overlay[{MakePIPNV[-1, T0, RGBColor["#287DAB"], Black],
     MakePIP[-1, T0, RGBColor["#287DAB"]]}],
   Overlay[{MakePIPNV[-1, T0 + 5, RGBColor["#287DAB"], Black],
     MakePIP[-1, T0 + 5, RGBColor["#287DAB"]]}],
   Overlay[{MakePIPNV[-1, T0 + 10, RGBColor["#287DAB"], Black],
     MakePIP[-1, T0 + 10, RGBColor["#287DAB"]]}]]]
Quiet[List[Overlay[{MakePIPNV[0, T0, Black, White], MakePIP[0, T0, Black]}],
   Overlay[{MakePIPNV[0, T0 + 5, Black, White], MakePIP[0, T0 + 5, Black]}], Overlay[
    {MakePIPNV[0, T0 + 10, Black, White], MakePIP[0, T0 + 10, Black]}]]] (*linear*)
Quiet[List[Overlay[{MakePIPNV[1, T0, RGBColor["#B09771"], Black],
     MakePIP[1, T0, RGBColor["#B09771"]]}],
   Overlay[{MakePIPNV[1, T0 + 5, RGBColor["#B09771"], Black],
     MakePIP[1, T0 + 5, RGBColor["#B09771"]]}],
   Overlay[{MakePIPNV[1, T0 + 10, RGBColor["#B09771"], Black],
     MakePIP[1, T0 + 10, RGBColor["#B09771"]]}]]] (*generalist*)
```



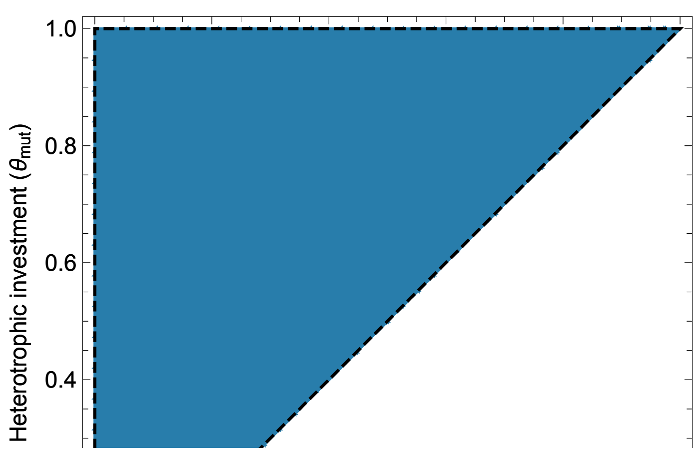
Out[ ]= {


Mutant Heterotrophic investme
0.0
0.2
0.4
0.6
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

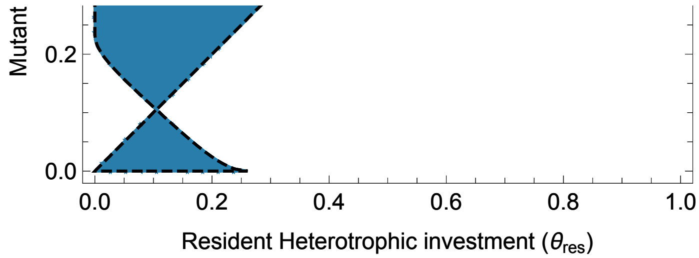
,


Mutant Heterotrophic investment (θ_{mut})
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

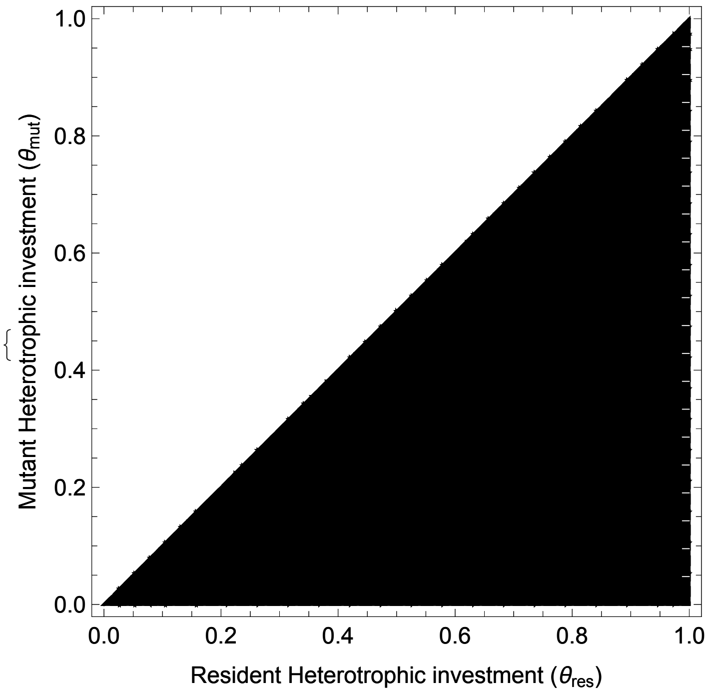
,


Heterotrophic investment (θ_{mut})
0.4
0.6
0.8
1.0

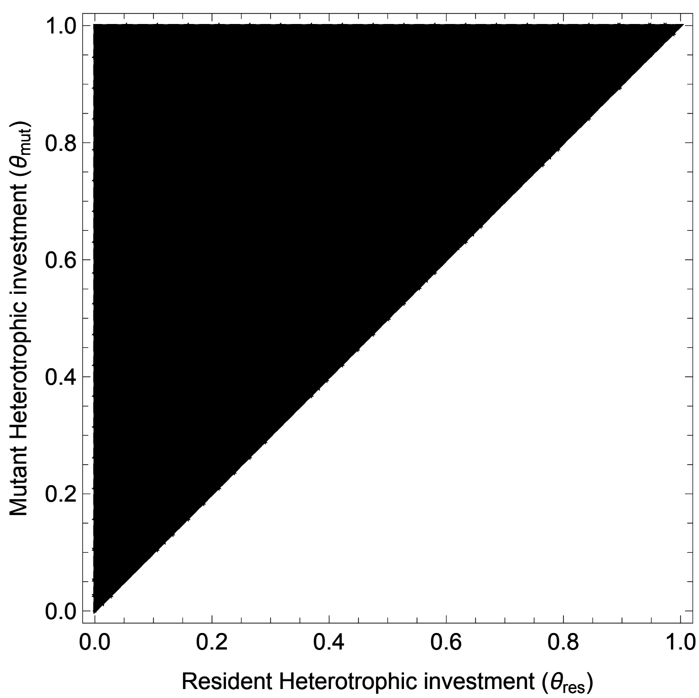
}


Mutant
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

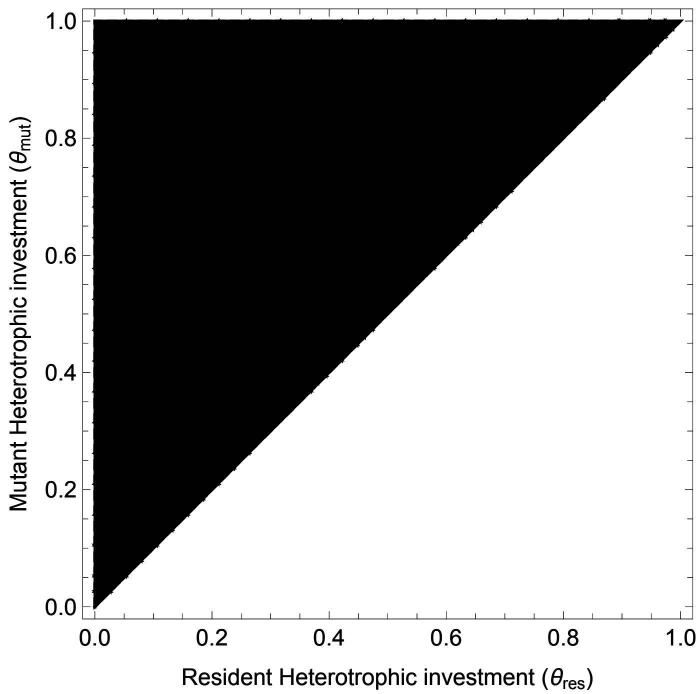
Out[ ]= {


Mutant Heterotrophic investment (θ_{mut})
Resident Heterotrophic investment (θ_{res})
0.0
0.2
0.4
0.6
0.8
1.0

,


Mutant Heterotrophic investment (θ_{mut})
Resident Heterotrophic investment (θ_{res})
0.0
0.2
0.4
0.6
0.8
1.0

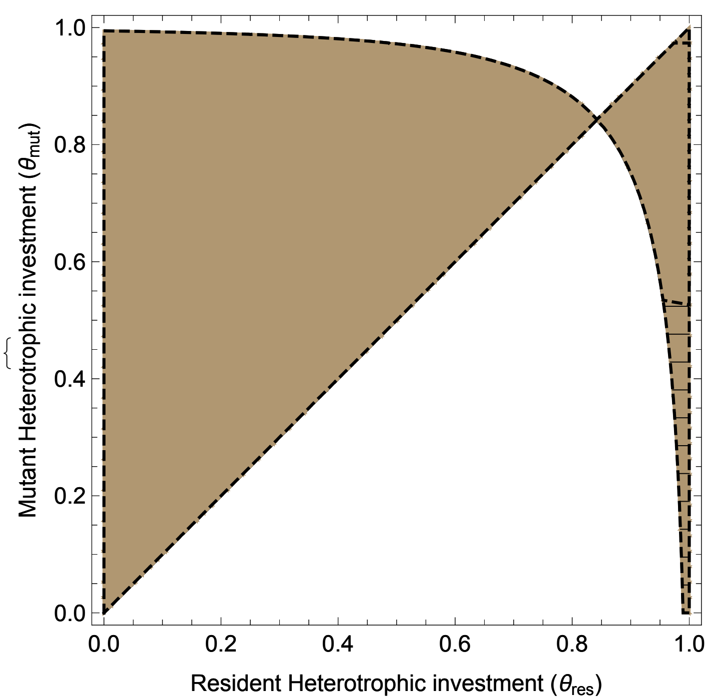
,



1.0
0.8
0.6
0.4
0.2
0.0
Mutant Heterotrophic investment (θ_{mut})
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

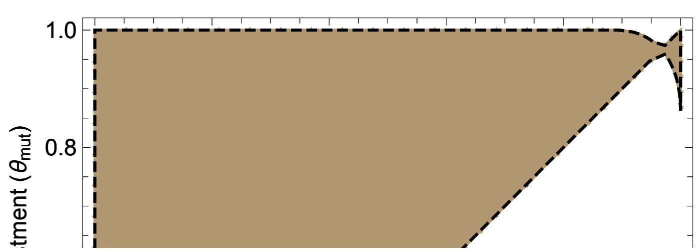
}

Out[∘]= {

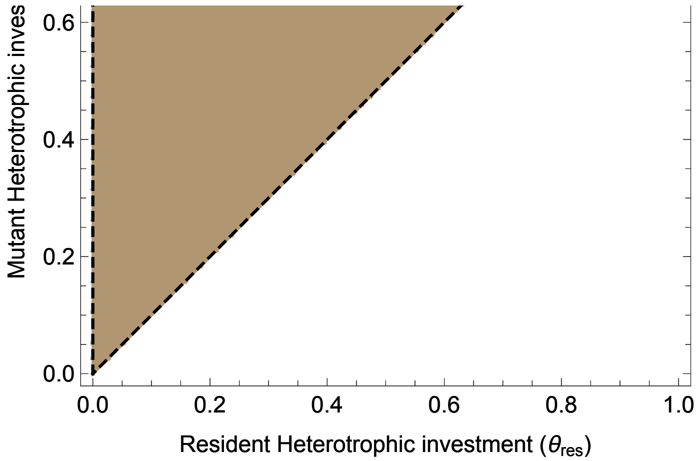


1.0
0.8
0.6
0.4
0.2
0.0
Mutant Heterotrophic investment (θ_{mut})
0.0
0.2
0.4
0.6
0.8
1.0
Resident Heterotrophic investment (θ_{res})

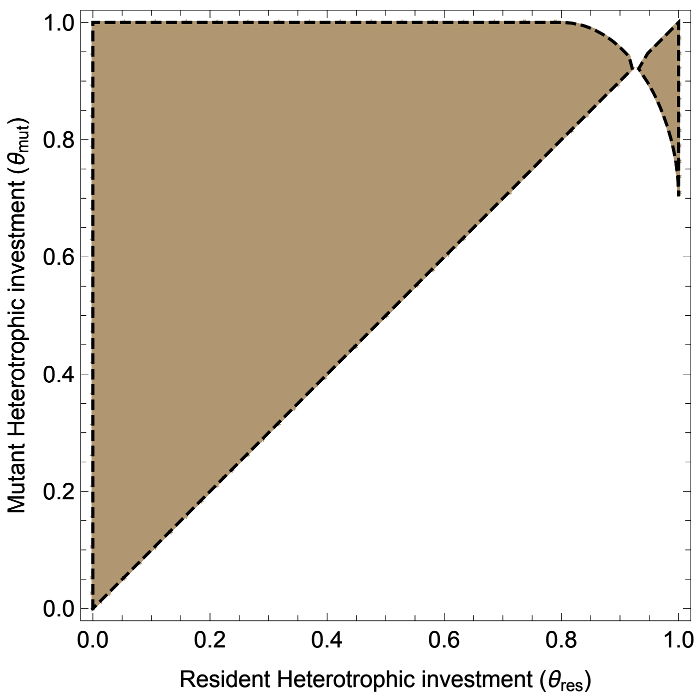
,



1.0
0.8
tment (θ_{mut})



,

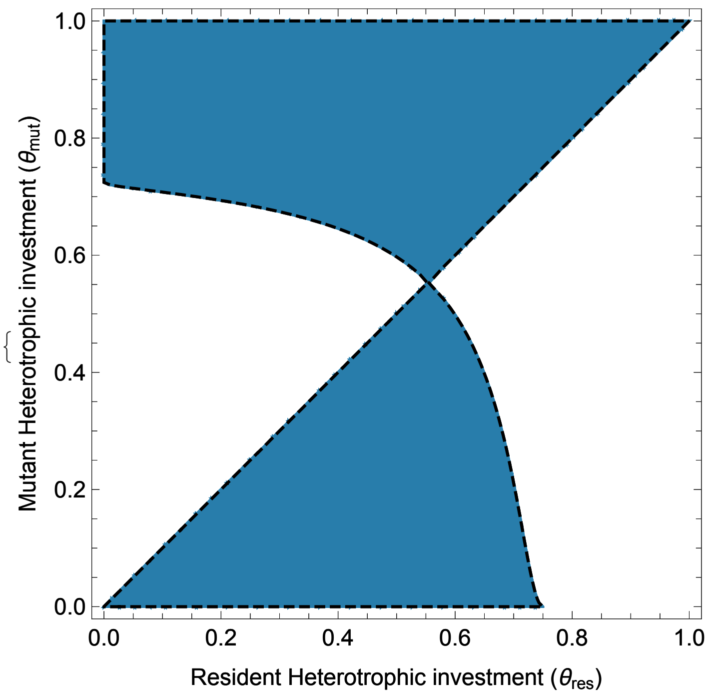


}

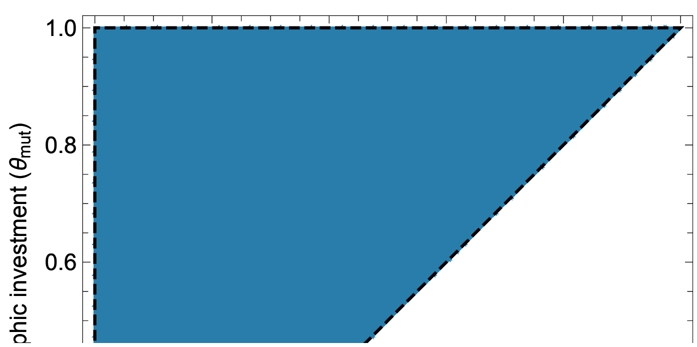
K_B=3*10^8, I_{in}=150

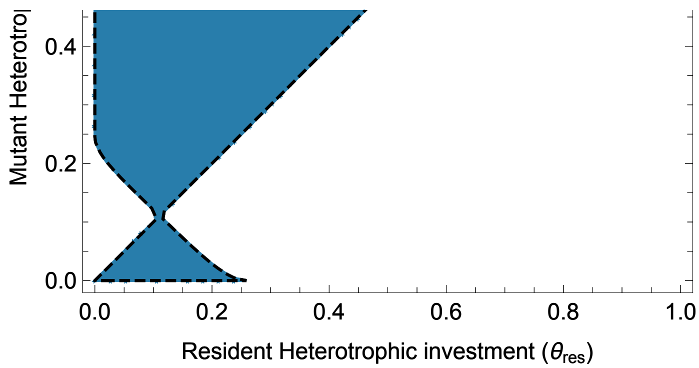
In[ ]:=
```
K_B = 3 × 10^8; I_in = 150;
Quiet[List[Overlay[{MakePIPNV[-1, T0, RGBColor["#287DAB"], Black],
    MakePIP[-1, T0, RGBColor["#287DAB"]]}],
  Overlay[{MakePIPNV[-1, T0 + 5, RGBColor["#287DAB"], Black],
    MakePIP[-1, T0 + 5, RGBColor["#287DAB"]]}],
  Overlay[{MakePIPNV[-1, T0 + 10, RGBColor["#287DAB"], Black],
    MakePIP[-1, T0 + 10, RGBColor["#287DAB"]]}]]]
Quiet[List[Overlay[{MakePIPNV[0, T0, Black, White], MakePIP[0, T0, Black]}],
  Overlay[{MakePIPNV[0, T0 + 5, Black, White], MakePIP[0, T0 + 5, Black]}], Overlay[
   {MakePIPNV[0, T0 + 10, Black, White], MakePIP[0, T0 + 10, Black]}]]] (*linear*)
Quiet[List[Overlay[{MakePIPNV[1, T0, RGBColor["#B09771"], Black],
    MakePIP[1, T0, RGBColor["#B09771"]]}],
  Overlay[{MakePIPNV[1, T0 + 5, RGBColor["#B09771"], Black],
    MakePIP[1, T0 + 5, RGBColor["#B09771"]]}],
  Overlay[{MakePIPNV[1, T0 + 10, RGBColor["#B09771"], Black],
    MakePIP[1, T0 + 10, RGBColor["#B09771"]]}]]] (*generalist*)
```

Out[ ]= {

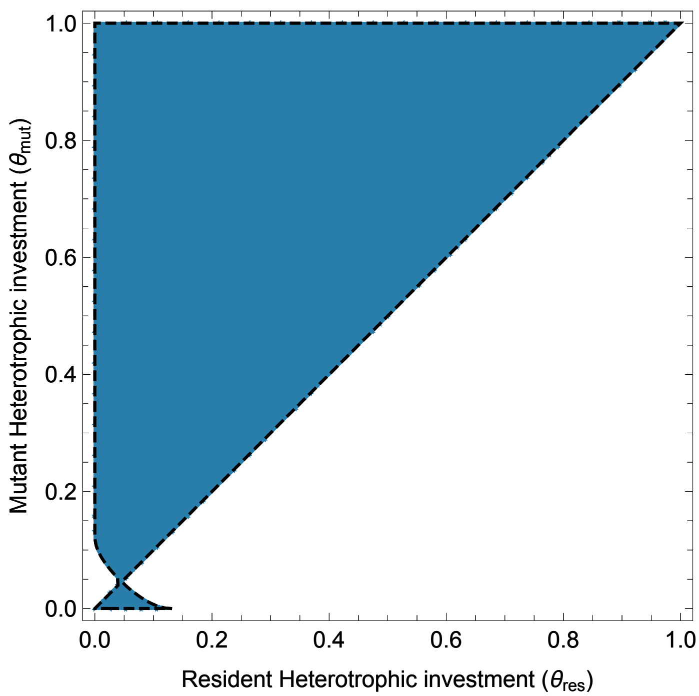


,





,



}

Out[ ]= $Aborted

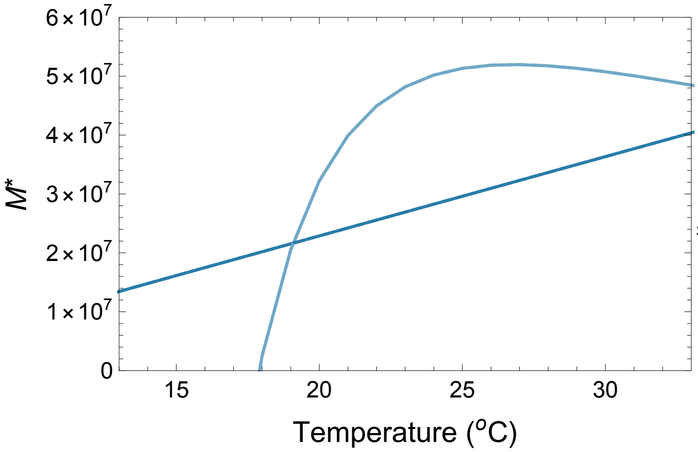
Out[ ]= $Aborted

C-cycling related figures (Dashed - genetically static, Solid - evolving)

K_B=1*10^8, I_{in}=100

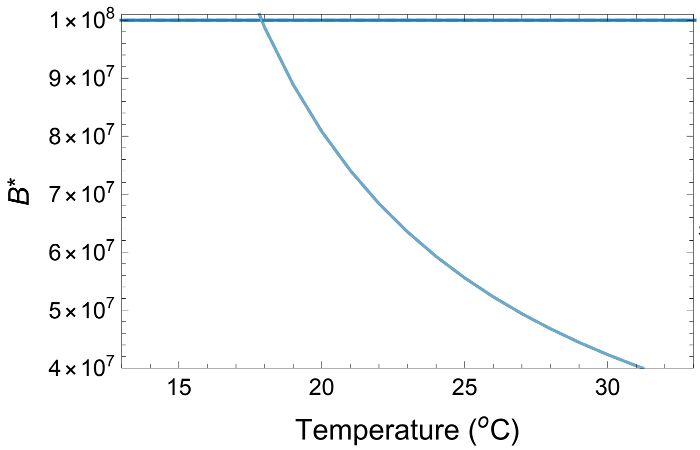
In[∘]:=
```
K_B = 1 × 10^8; I_in = 100;
Quiet[Ccyclingspec[-1, 0, 6 × 10^7 , 4 × 10^7, 1.01 × 10^8, 10^5, 3 × 10^6]]
Quiet[Ccycling[0, 1.3 × 10^7, 6 × 10^7 , 4 × 10^7, 1.01 × 10^8, -20 000, 3 × 10^6]]
Quiet[Ccycling[1, 1.4 × 10^7, 9 × 10^7 , 2 × 10^7, 10 × 10^7, 0, 3 × 10^6]]
```

Out[ ]= {{



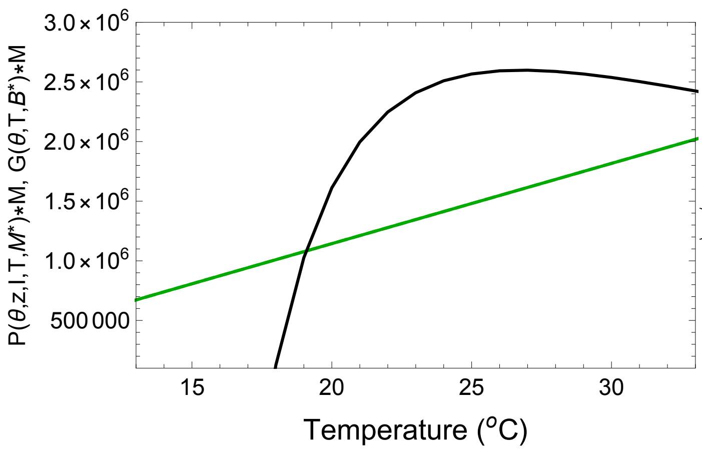
6 × 10^7
5 × 10^7
4 × 10^7
3 × 10^7
2 × 10^7
1 × 10^7
0
M*
15
20
25
30
Temperature (°C)

,



1 × 10^8
9 × 10^7
8 × 10^7
7 × 10^7
6 × 10^7
5 × 10^7
4 × 10^7
B*
15
20
25
30
Temperature (°C)

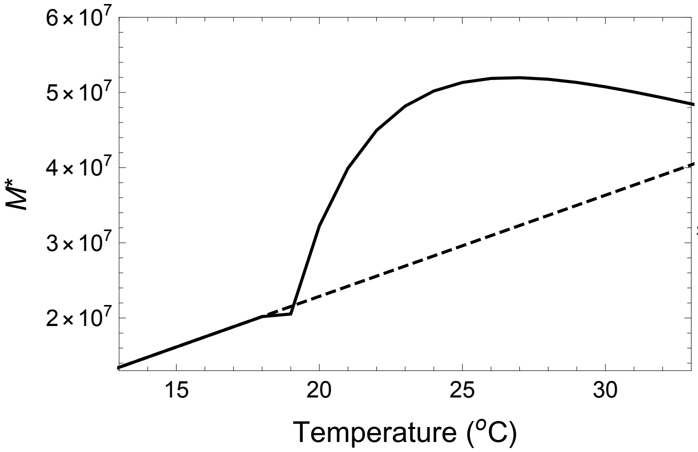
,



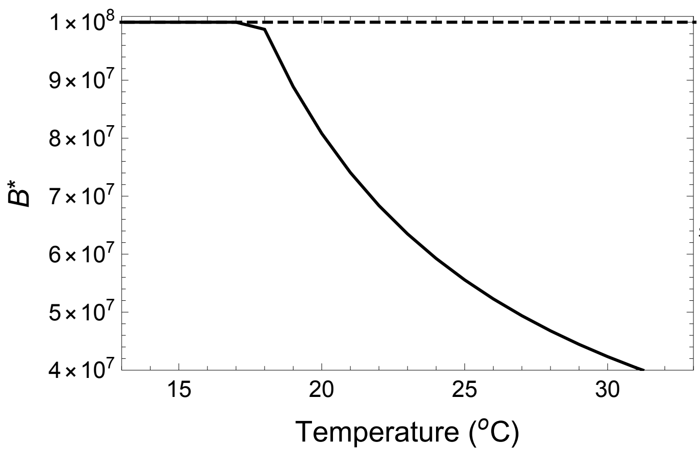
3.0 × 10^6
2.5 × 10^6
2.0 × 10^6
1.5 × 10^6
1.0 × 10^6
500 000
P(θ,z,I,T,M*)*M, G(θ,T,B*)*M
15
20
25
30
Temperature (°C)

}}

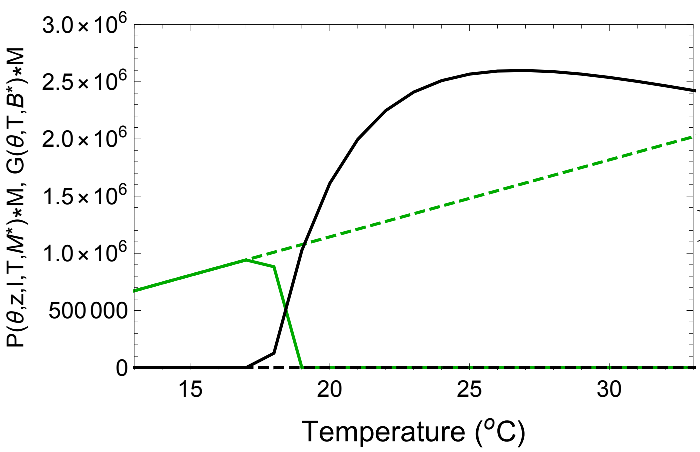
Out[]= {{


M*
6×10^7
5×10^7
4×10^7
3×10^7
2×10^7
15
20
25
30
Temperature (°C)

,

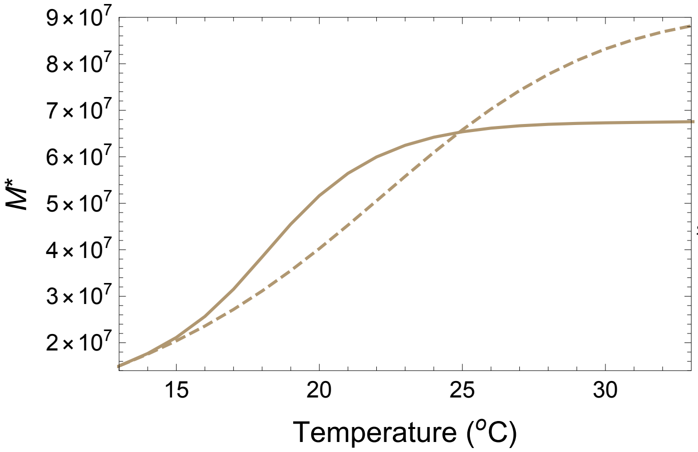

B*
1×10^8
9×10^7
8×10^7
7×10^7
6×10^7
5×10^7
4×10^7
15
20
25
30
Temperature (°C)

,


P(θ,z,I,T,M*)*M, G(θ,T,B*)*M
3.0×10^6
2.5×10^6
2.0×10^6
1.5×10^6
1.0×10^6
500000
0
15
20
25
30
Temperature (°C)

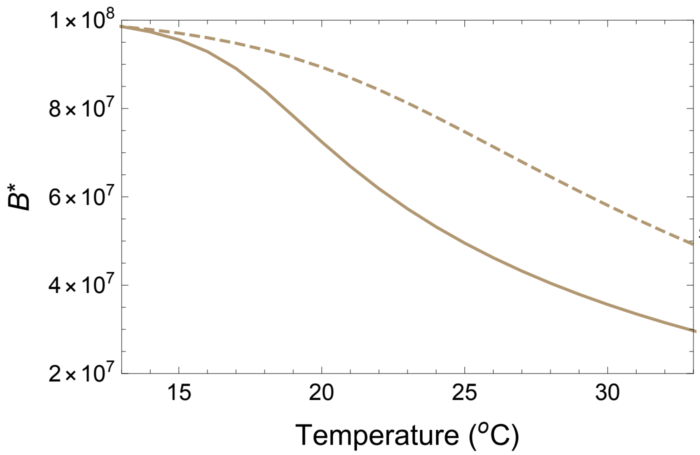
}}

Out[ ]= {{



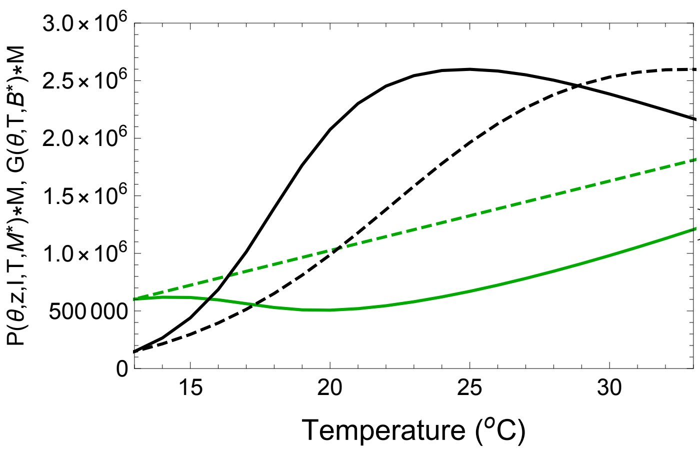
M*
9×10^7
8×10^7
7×10^7
6×10^7
5×10^7
4×10^7
3×10^7
2×10^7
15
20
25
30
Temperature (°C)

,



B*
1×10^8
8×10^7
6×10^7
4×10^7
2×10^7
15
20
25
30
Temperature (°C)

,



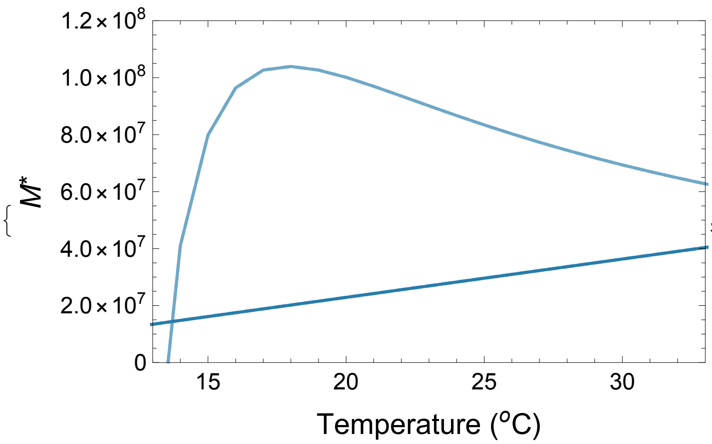
P(θ,z,I,T,M*)*M, G(θ,T,B*)*M
3.0×10^6
2.5×10^6
2.0×10^6
1.5×10^6
1.0×10^6
500000
0
15
20
25
30
Temperature (°C)

}}

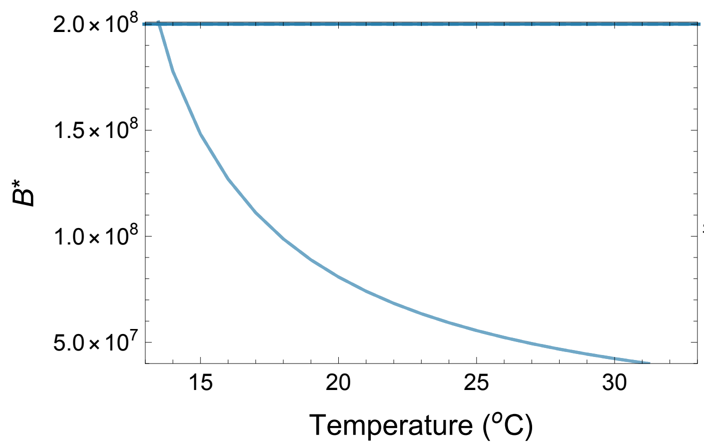
K_B=2*10^8, I_{in}=100

In[4005]:=
```
KB = 2 × 10^8; Iin = 100;
Quiet[Ccyclingspec[-1, 0, 1.2 × 10^8 , 4 × 10^7, 2.01 × 10^8, 10^5, 5.5 × 10^6]]
Quiet[Ccycling[0, 1.3 × 10^7, 1.2 × 10^8 , 4 × 10^7, 2.01 × 10^8, -20 000, 6 × 10^6]]
Quiet[Ccycling[1, 1.4 × 10^7, 1.2 × 10^8 , 2 × 10^7, 2 × 10^8, 0, 6 × 10^6]]
```

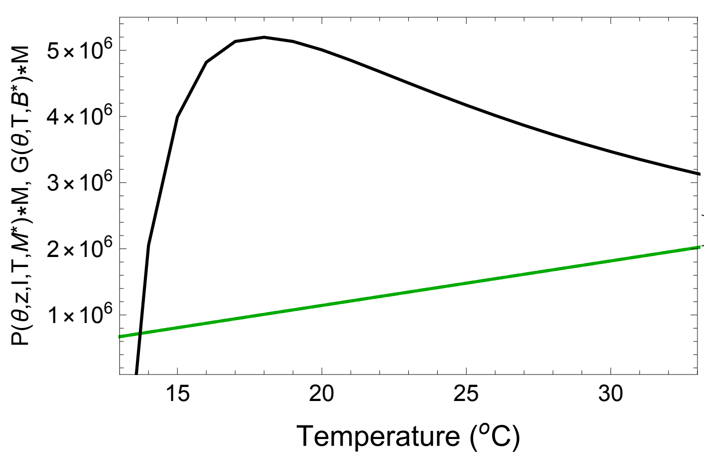
Out[4006]= {{



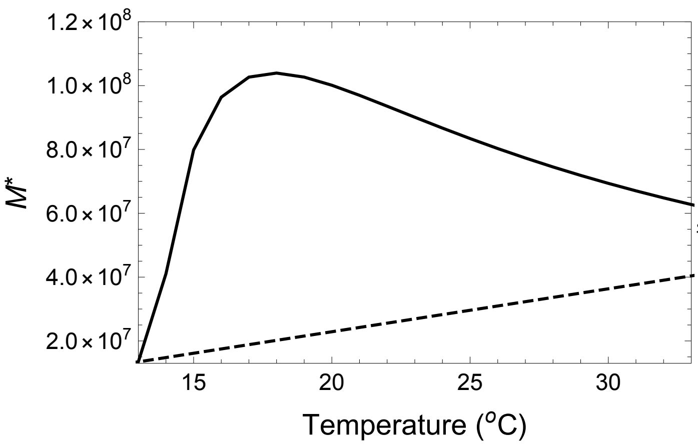
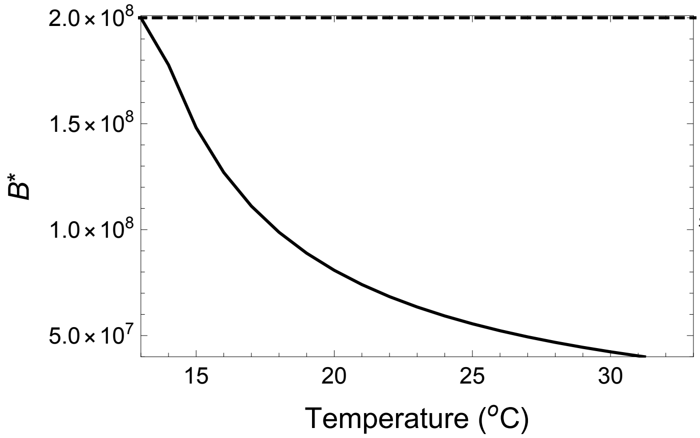
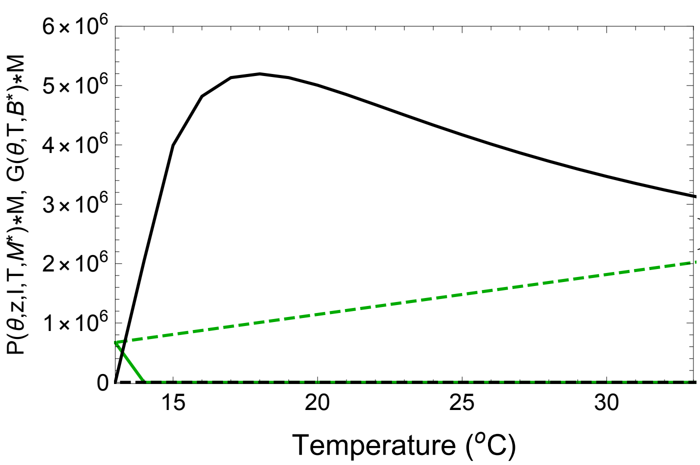
,



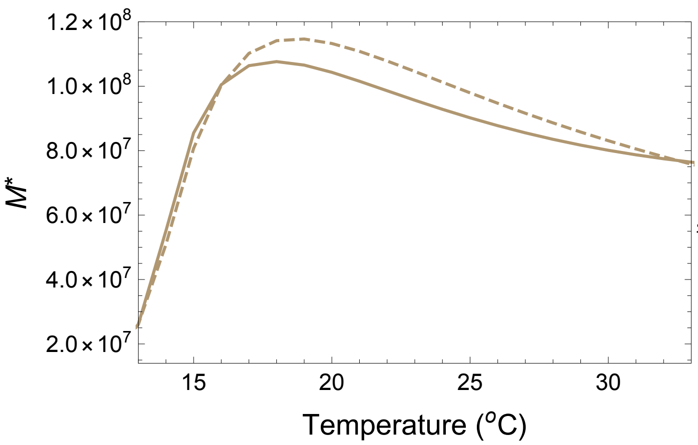
,



}}

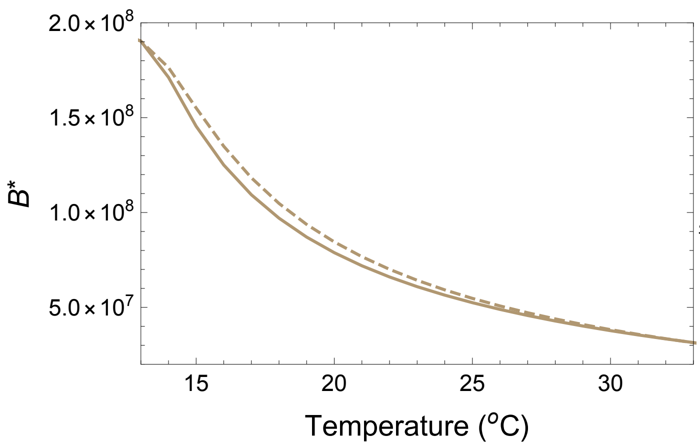
Out[4007]= {{
1.2 × 10^8
1.0 × 10^8
8.0 × 10^7
6.0 × 10^7
4.0 × 10^7
2.0 × 10^7
M*
15
20
25
30
Temperature (°C)
,

2.0 × 10^8
1.5 × 10^8
1.0 × 10^8
5.0 × 10^7
B*
15
20
25
30
Temperature (°C)
,

6 × 10^6
5 × 10^6
4 × 10^6
3 × 10^6
2 × 10^6
1 × 10^6
0
P(θ,z,I,T,M*)*M, G(θ,T,B*)*M
15
20
25
30
Temperature (°C)
}}

Out[4008]= {{



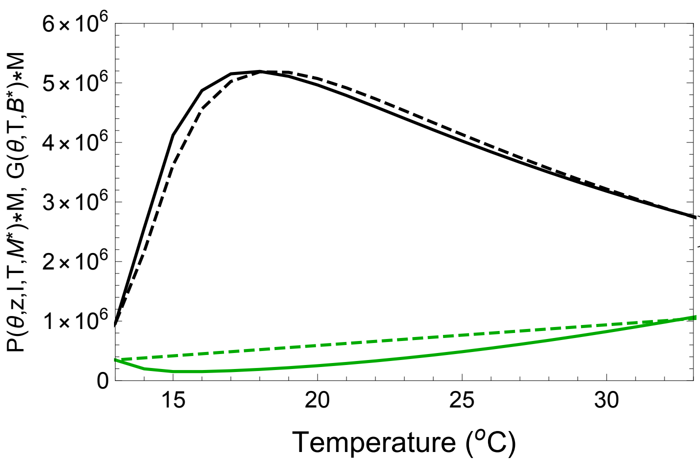
M^*
1.2 × 10^8
1.0 × 10^8
8.0 × 10^7
6.0 × 10^7
4.0 × 10^7
2.0 × 10^7
15
20
25
30
Temperature (°C)

,



B^*
2.0 × 10^8
1.5 × 10^8
1.0 × 10^8
5.0 × 10^7
15
20
25
30
Temperature (°C)

,



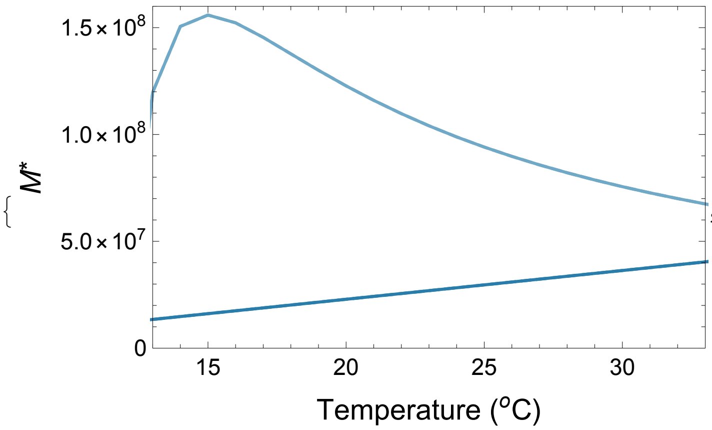
P(θ,z,I,T,M^*)*M, G(θ,T,B^*)*M
6 × 10^6
5 × 10^6
4 × 10^6
3 × 10^6
2 × 10^6
1 × 10^6
0
15
20
25
30
Temperature (°C)

}}

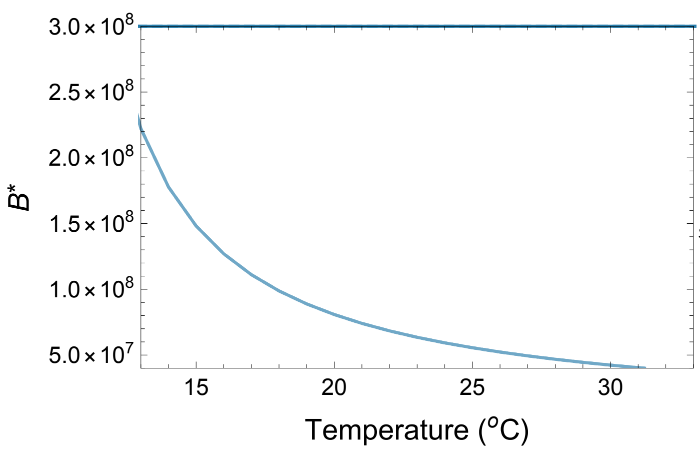
K_B=3*10^8, I_{in}=100

In[4069]:=
```
K_B = 3 × 10^8; I_in = 100;
Quiet[Ccyclingspec[-1, 0, 1.6 × 10^8 , 4 × 10^7, 3 × 10^8, 10^5, 8 × 10^6]]
Quiet[Ccycling[0, 1.3 × 10^7, 2 × 10^8 , 4 × 10^7, 2.5 × 10^8, -20 000, 8 × 10^6]]
Quiet[Ccycling[1, 1.4 × 10^7, 2 × 10^8 , 0, 3 × 10^8, 0, 8 × 10^6]]
```

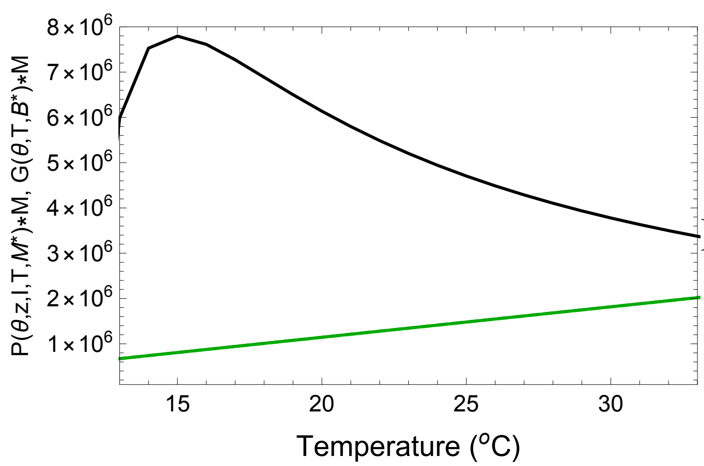
Out[4070]= {{



,

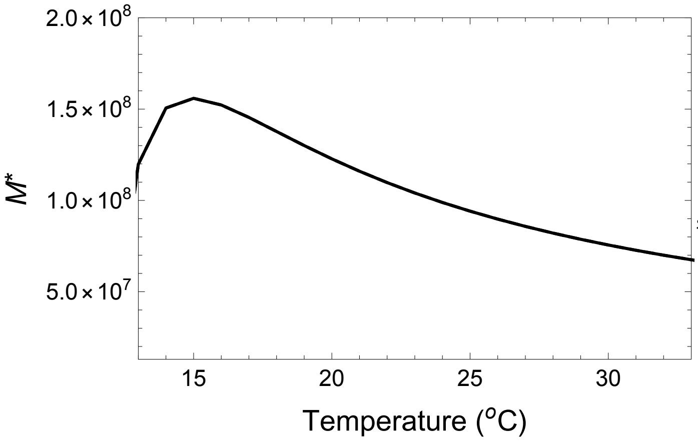


,

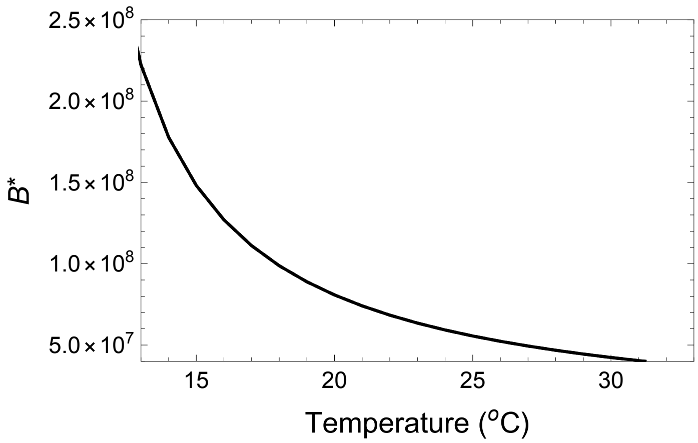


}}

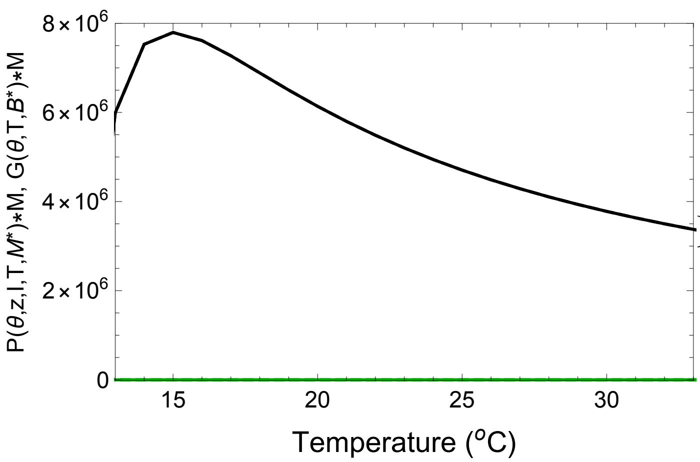
Out[4071]= {{


2.0×10^8
1.5×10^8
1.0×10^8
5.0×10^7
M*
15
20
25
30
Temperature (°C)

,

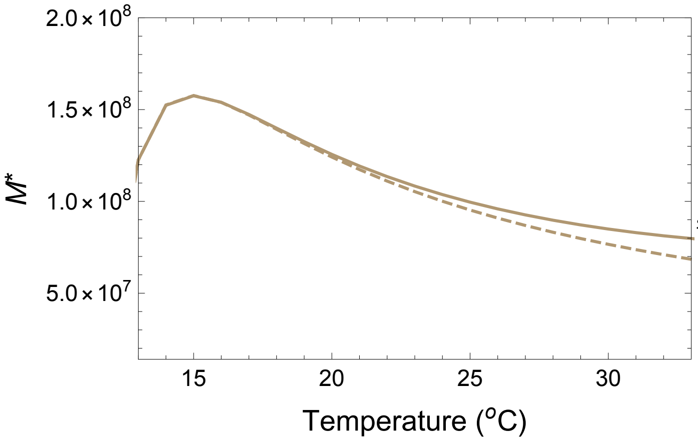

2.5×10^8
2.0×10^8
1.5×10^8
1.0×10^8
5.0×10^7
B*
15
20
25
30
Temperature (°C)

,


8×10^6
6×10^6
4×10^6
2×10^6
0
P(θ,z,I,T,M*)*M, G(θ,T,B*)*M
15
20
25
30
Temperature (°C)

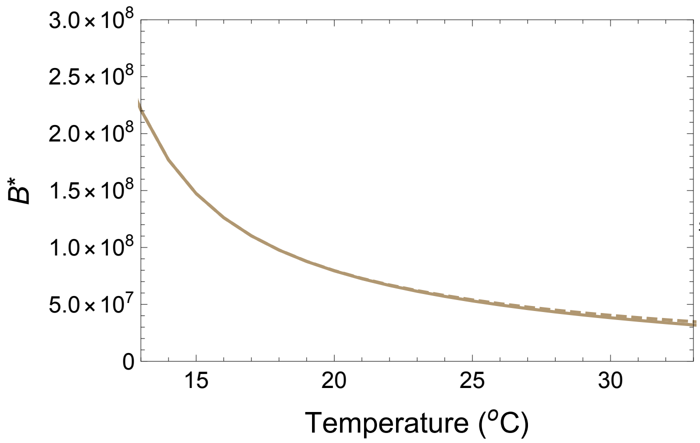
}}

Out[4072]= {{



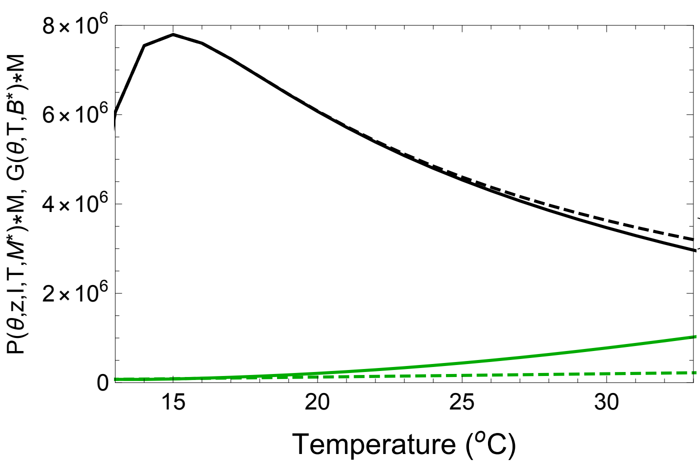
2.0 × 10^8
1.5 × 10^8
1.0 × 10^8
5.0 × 10^7
M*
15
20
25
30
Temperature (°C)

,



3.0 × 10^8
2.5 × 10^8
2.0 × 10^8
1.5 × 10^8
1.0 × 10^8
5.0 × 10^7
0
B*
15
20
25
30
Temperature (°C)

,



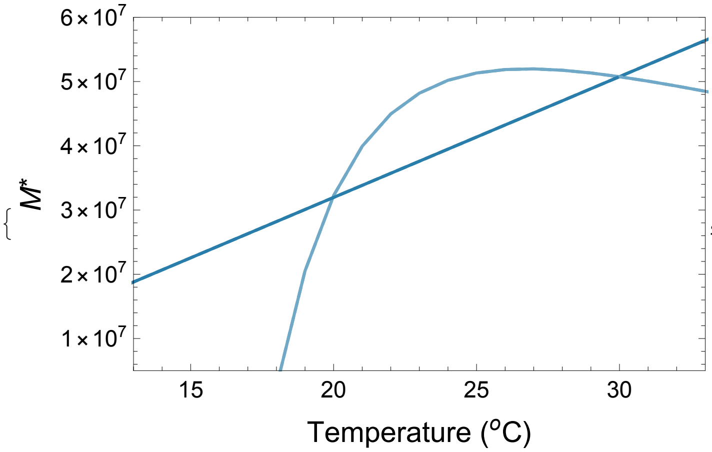
8 × 10^6
6 × 10^6
4 × 10^6
2 × 10^6
0
P(θ,z,I,T,M*)*M, G(θ,T,B*)*M
15
20
25
30
Temperature (°C)

}}

K_B=1*10^8, I_{in}=150

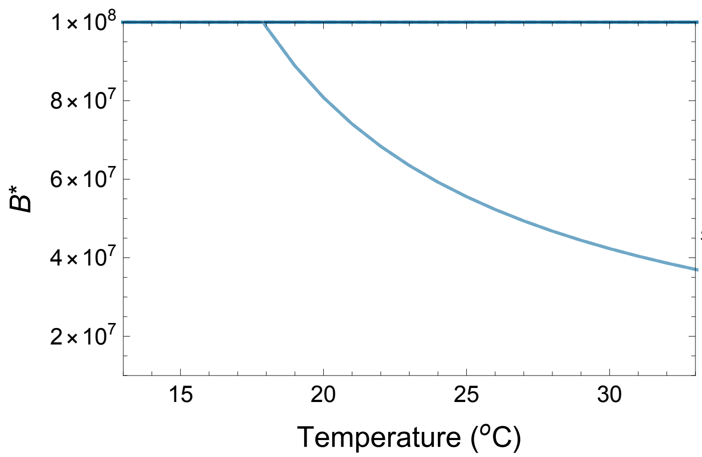
In[4037]:=
```
K_B = 1 × 10^8; I_in = 150;
Quiet[Ccyclingspec[-1, .5 × 10^7, 6.0 × 10^7 , 1 × 10^7, 1 × 10^8, 0, 4 × 10^6]]
Quiet[Ccycling[0, .5 × 10^7, 6.0 × 10^7 , 1 × 10^7, 1 × 10^8, 0, 4 × 10^6]]
Quiet[Ccycling[1, 1.5 × 10^7, 1 × 10^8 , 1 × 10^7, 1 × 10^8, 0, 3 × 10^6]]
```

Out[4038]= {{



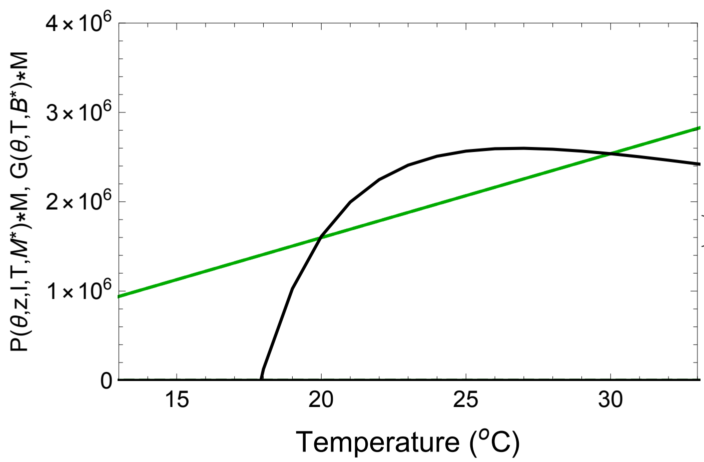
M*
6 × 10^7
5 × 10^7
4 × 10^7
3 × 10^7
2 × 10^7
1 × 10^7
15
20
25
30
Temperature (°C)

,



B*
1 × 10^8
8 × 10^7
6 × 10^7
4 × 10^7
2 × 10^7
15
20
25
30
Temperature (°C)

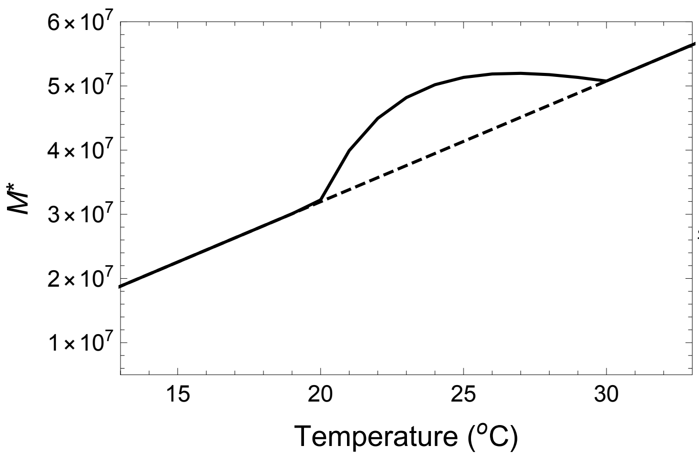
,



P(θ,z,I,T,M*)*M, G(θ,T,B*)*M
4 × 10^6
3 × 10^6
2 × 10^6
1 × 10^6
0
15
20
25
30
Temperature (°C)

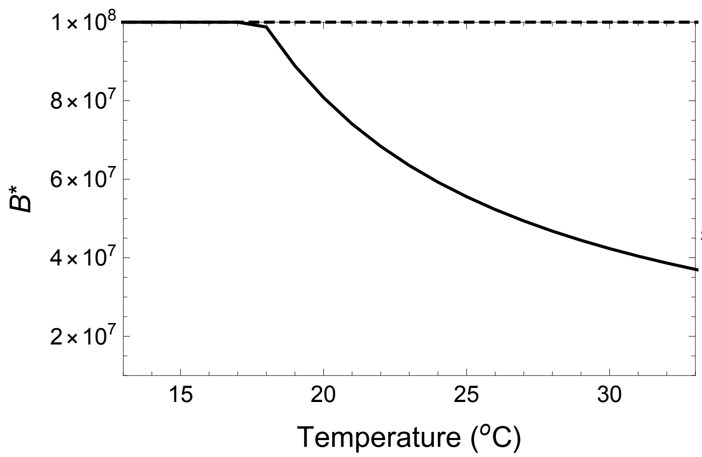
}}

Out[4039]= {{



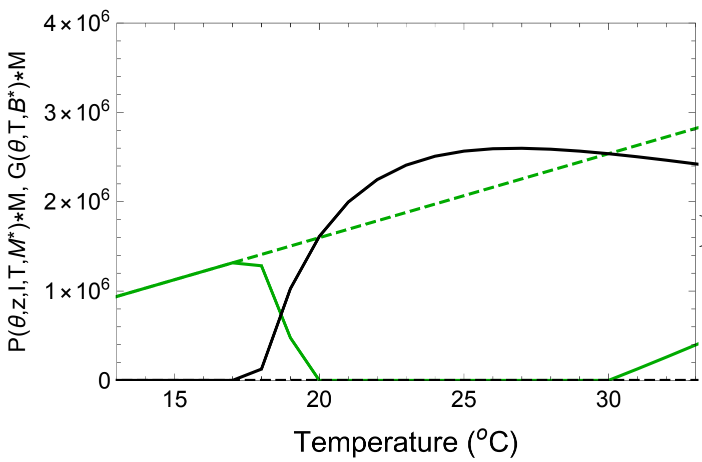
M^*
6×10^7
5×10^7
4×10^7
3×10^7
2×10^7
1×10^7
15
20
25
30
Temperature (^{o}C)

,



B^*
1×10^8
8×10^7
6×10^7
4×10^7
2×10^7
15
20
25
30
Temperature (^{o}C)

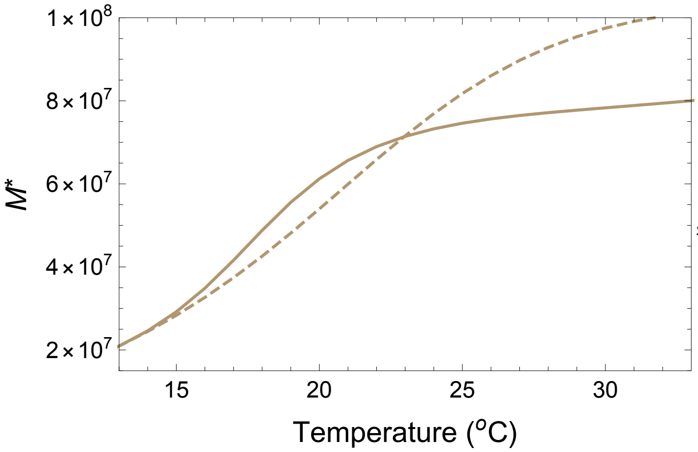
,



$P(\theta,z,I,T,M^*)*M$, $G(\theta,T,B^*)*M$
4×10^6
3×10^6
2×10^6
1×10^6
0
15
20
25
30
Temperature (^{o}C)

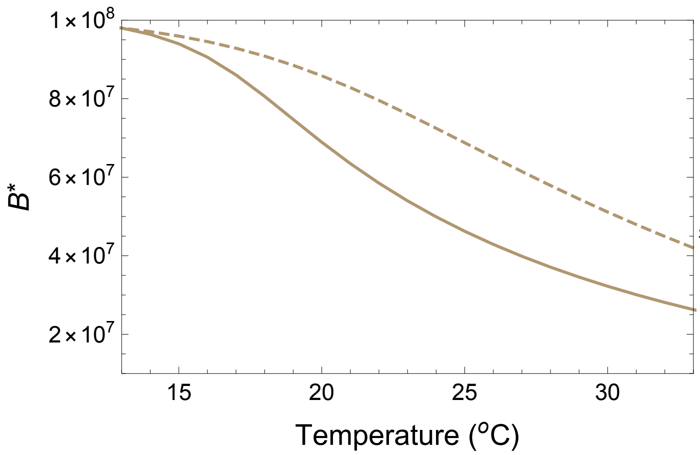
}}

Out[4040]= {{



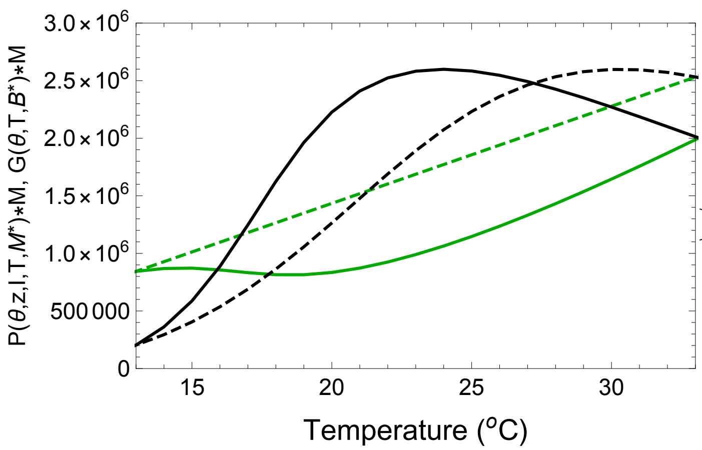
1×10^8
8×10^7
6×10^7
4×10^7
2×10^7
M^*
15
20
25
30
Temperature (^{o}C)

,



1×10^8
8×10^7
6×10^7
4×10^7
2×10^7
B^*
15
20
25
30
Temperature (^{o}C)

,



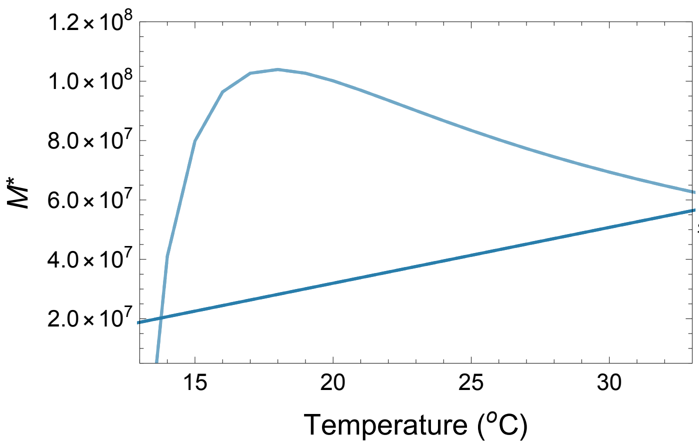
3.0×10^6
2.5×10^6
2.0×10^6
1.5×10^6
1.0×10^6
500 000
0
P(θ,z,I,T,M^*)*M, G(θ,T,B^*)*M
15
20
25
30
Temperature (^{o}C)

}}

K_B=2*10^8, I_{in}=150

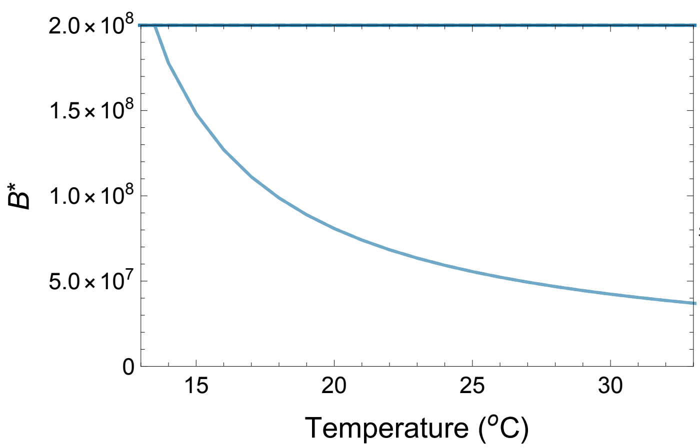
In[4061]:=
```
K_B = 2 × 10^8; I_in = 150;
Quiet[Ccyclingspec[-1, .5 × 10^7, 1.2 × 10^8 , 0, 2 × 10^8, 0, 7 × 10^6]]
Quiet[Ccycling[0, .5 × 10^7, 1.2 × 10^8 , 0, 2 × 10^8, 0, 8 × 10^6]]
Quiet[Ccycling[1, 0 × 10^7, 1.2 × 10^8 , 2 × 10^7, 2 × 10^8, 0, 8 × 10^6]]
```

Out[4062]= {{



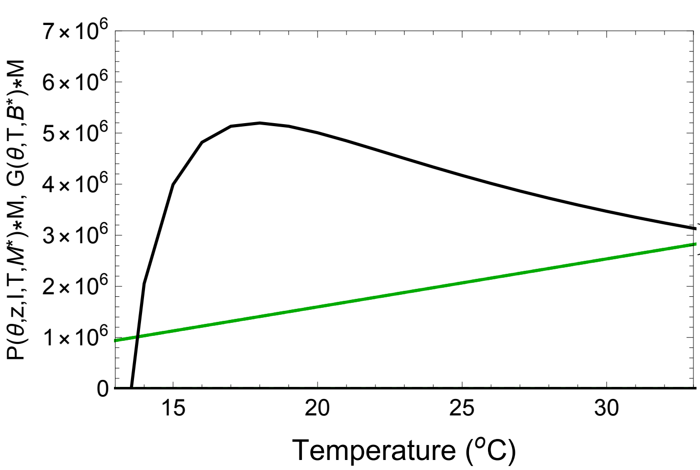
1.2 × 10^8
1.0 × 10^8
8.0 × 10^7
6.0 × 10^7
4.0 × 10^7
2.0 × 10^7
M*
15
20
25
30
Temperature (^oC)

,



2.0 × 10^8
1.5 × 10^8
1.0 × 10^8
5.0 × 10^7
0
B*
15
20
25
30
Temperature (^oC)

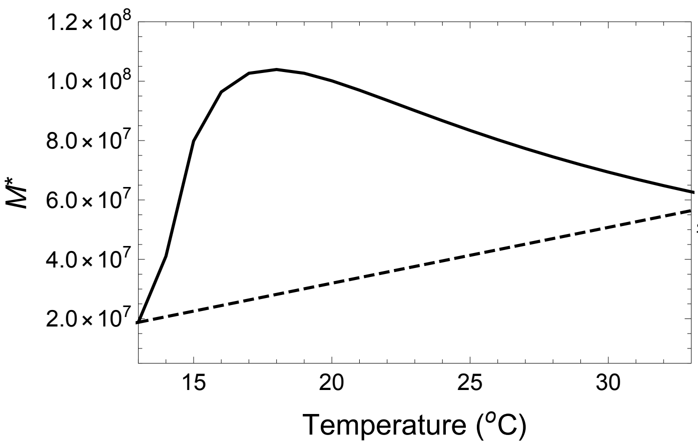
,



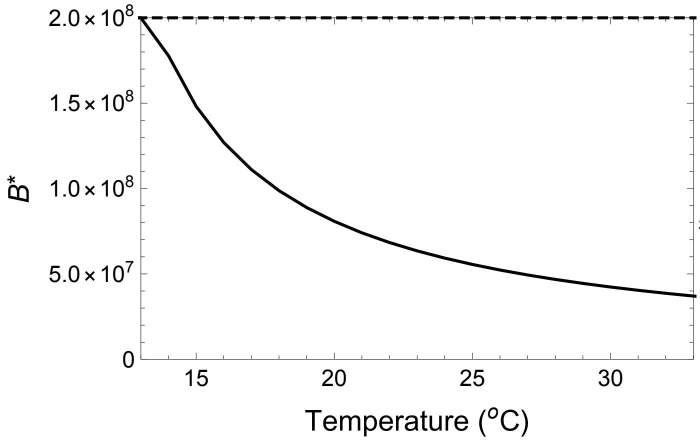
7 × 10^6
6 × 10^6
5 × 10^6
4 × 10^6
3 × 10^6
2 × 10^6
1 × 10^6
0
P(θ,z,I,T,M*)*M, G(θ,T,B*)*M
15
20
25
30
Temperature (^oC)

}}

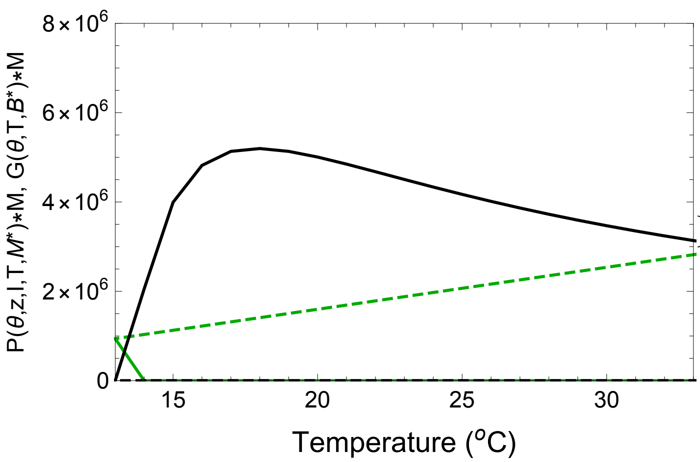
Out[4063]= {{


1.2×10^8
1.0×10^8
8.0×10^7
6.0×10^7
4.0×10^7
2.0×10^7
M^*
15
20
25
30
Temperature (^{o}C)

,

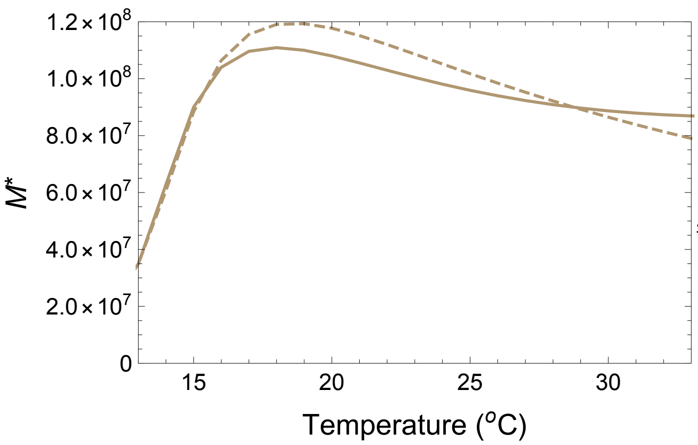

2.0×10^8
1.5×10^8
1.0×10^8
5.0×10^7
0
B^*
15
20
25
30
Temperature (^{o}C)

,


8×10^6
6×10^6
4×10^6
2×10^6
0
P(θ,z,I,T,M^*)*M, G(θ,T,B^*)*M
15
20
25
30
Temperature (^{o}C)

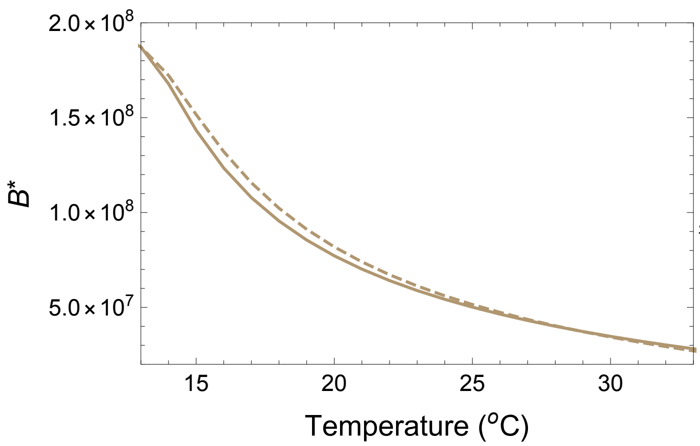
}}

Out[4064]= {{



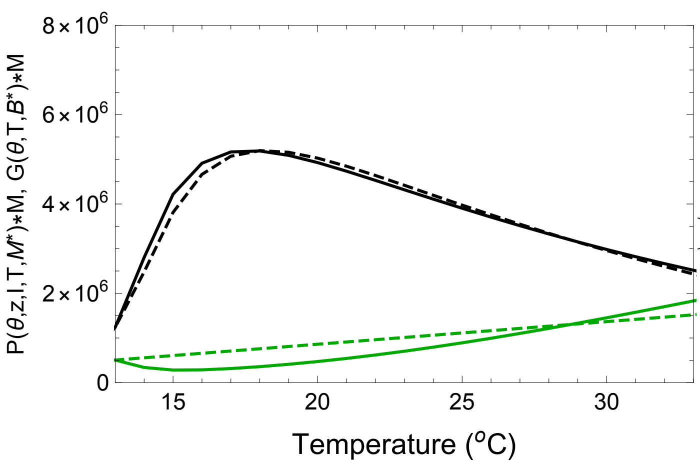
1.2×10^8
1.0×10^8
8.0×10^7
6.0×10^7
4.0×10^7
2.0×10^7
0
M^*
15
20
25
30
Temperature (^{o}C)

,



2.0×10^8
1.5×10^8
1.0×10^8
5.0×10^7
B^*
15
20
25
30
Temperature (^{o}C)

,



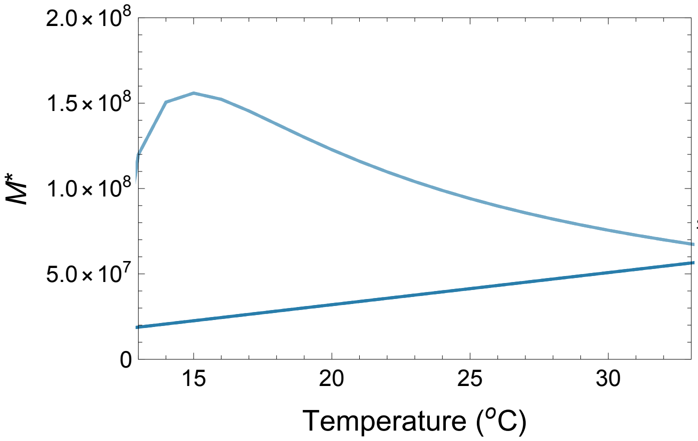
8×10^6
6×10^6
4×10^6
2×10^6
0
P(θ,z,I,T,M^*)*M, G(θ,T,B^*)*M
15
20
25
30
Temperature (^{o}C)

}}

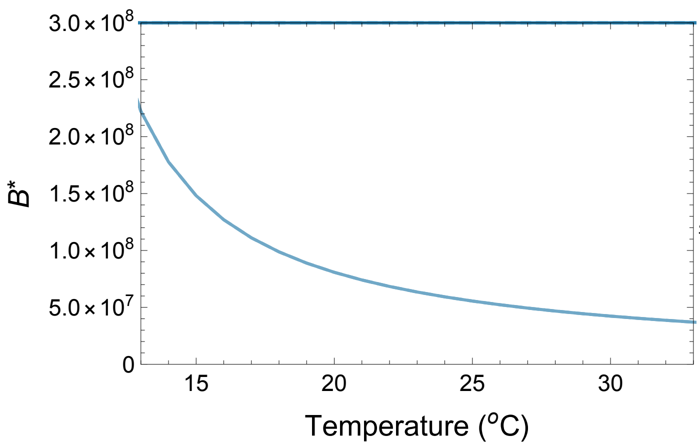
K_B=3*10^8, I_{in}=150

```
K_B = 3 × 10^8; I_in = 150;
Quiet[Ccyclingspec[-1, 0 × 10^7, 2 × 10^8 , 0 × 10^7, 3 × 10^8, 0, 7 × 10^6]]
Quiet[Ccycling[0, 0 × 10^7, 2 × 10^8 , 0 × 10^7, 2.5 × 10^8, 0, 8 × 10^6]]
Quiet[Ccycling[1, 0 × 10^7, 2 × 10^8 , 2 × 10^7, 2.5 × 10^8, 0, 8 × 10^6]]
```

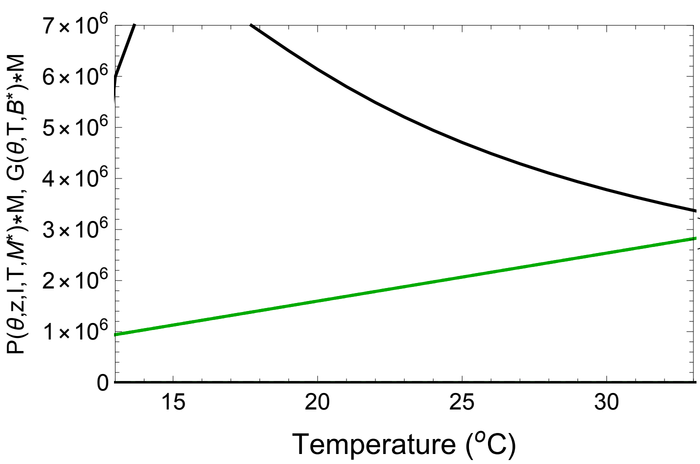
Out[4066]= {{


2.0×10^8
1.5×10^8
1.0×10^8
5.0×10^7
0
M*
15
20
25
30
Temperature (^oC)

,

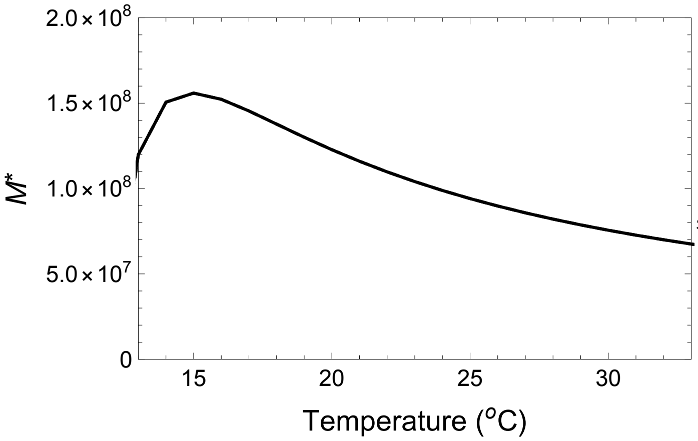

3.0×10^8
2.5×10^8
2.0×10^8
1.5×10^8
1.0×10^8
5.0×10^7
0
B*
15
20
25
30
Temperature (^oC)

,

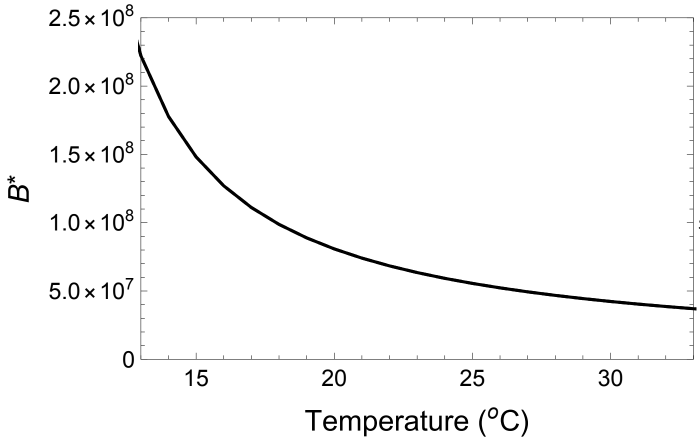

7×10^6
6×10^6
5×10^6
4×10^6
3×10^6
2×10^6
1×10^6
0
P(θ,z,I,T,M*)*M, G(θ,T,B*)*M
15
20
25
30
Temperature (^oC)

}}

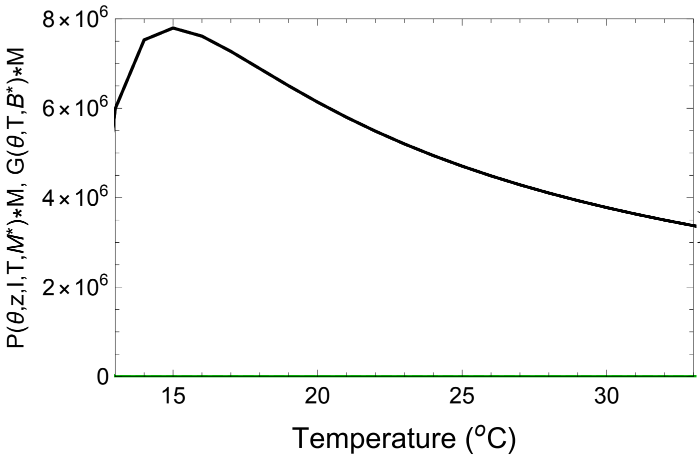
Out[4067]= {{


2.0 × 10^8
1.5 × 10^8
1.0 × 10^8
5.0 × 10^7
0
M*
15
20
25
30
Temperature (°C)

,

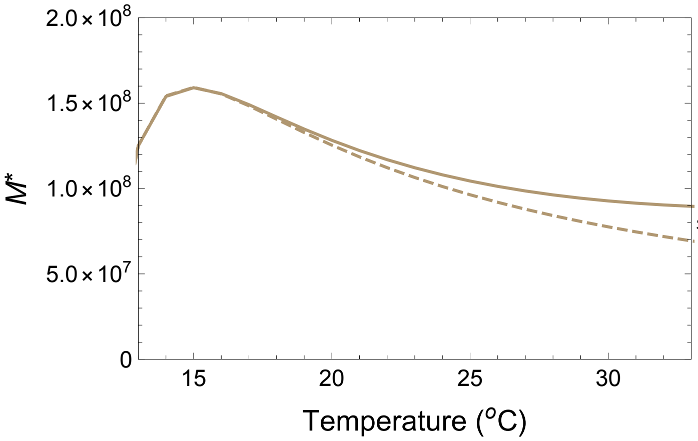

2.5 × 10^8
2.0 × 10^8
1.5 × 10^8
1.0 × 10^8
5.0 × 10^7
0
B*
15
20
25
30
Temperature (°C)

,

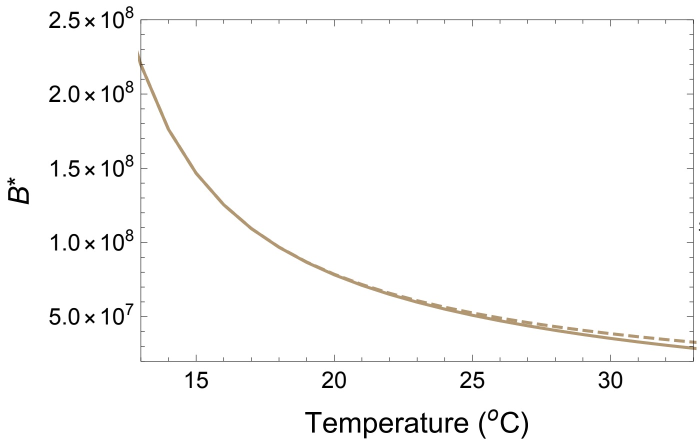

8 × 10^6
6 × 10^6
4 × 10^6
2 × 10^6
0
P(θ,z,I,T,M*)*M, G(θ,T,B*)*M
15
20
25
30
Temperature (°C)

}}

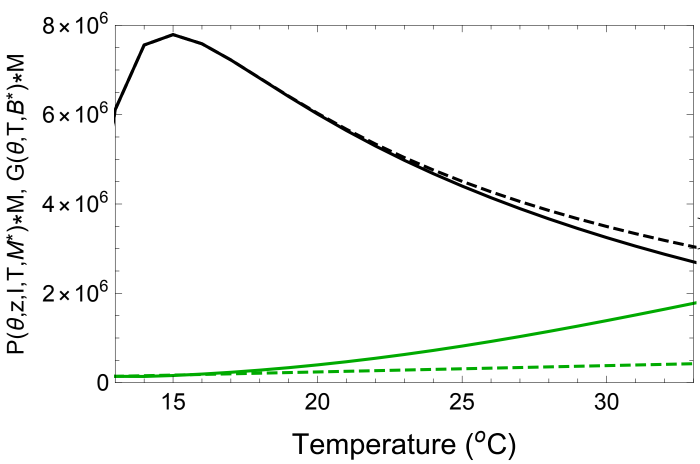
Out[4068]= {{


M*
2.0×10^8
1.5×10^8
1.0×10^8
5.0×10^7
0
15
20
25
30
Temperature (oC)

,


B*
2.5×10^8
2.0×10^8
1.5×10^8
1.0×10^8
5.0×10^7
15
20
25
30
Temperature (oC)

,


P(θ,z,I,T,M*)*M, G(θ,T,B*)*M
8×10^6
6×10^6
4×10^6
2×10^6
0
15
20
25
30
Temperature (oC)

}}