```
In[1]:= xhh1; xhs1; xsh1; xss1;
        xhh2; xhs2; xsh2; xss2;

In[3]:= x1 = xhh1 + xhs1 + xsh1 + xss1;
        x2 = xhh2 + xhs2 + xsh2 + xss2;

In[5]:= G = 5;
        M = 4;
        πS =.;
        πH = 1;

In[9]:= colors = {RGBColor["#B22222"], RGBColor["#DC143C"],
           RGBColor["#FF4501"], RGBColor["#00BFBF"], RGBColor["#6495ED"],
           RGBColor["#F08080"], RGBColor["#02008B"], RGBColor["#008B8B"]};
```

definitions

```
In[10]:= f1[k_, state_] := (k + KroneckerDelta[state, 1])/G (*Round[(k+KroneckerDelta[state,1])/G]*);

In[11]:= Phare1[focal_] := KroneckerDelta[focal, "Hare"];

In[12]:= Pcomposition[k_, state_, i_] :=
```

$$\sum_{l=0}^{k}\sum_{m=0}^{k}\sum_{n=0}^{k}\sum_{o=0}^{k}\sum_{p=0}^{G-1-k}\sum_{q=0}^{G-1-k}\sum_{r=0}^{G-1-k}\sum_{s=0}^{G-1-k}$$

```
         (If[l + m + n + o == k && p + q + r + s == G - 1 - k,
             ((Binomial[i[[1]], l] Binomial[i[[2]], p] Binomial[i[[3]], m]
                 Binomial[i[[4]], q] Binomial[i[[5]], n] Binomial[i[[6]], r] Binomial[
                  i[[7]], o] Binomial[i[[8]], s]) / Binomial[Total[i], G - 1])
              If[state == 1, HeavisideTheta[1 + n + o + r + s - M],
               If[state == 2, HeavisideTheta[1 + m + o + q + s - M], 0]], 0]);
```

Avg Payoffs for finite populations

The total population size is assumed to be N. We choose a group of size G from N.
Given that the population composition is given by i_j individuals of type $j \subset$ $(x_{hh1}, x_{hs1}, x_{sh1}, x_{ss1}, x_{hh2}, x_{hs2}, x_{sh2}, x_{ss2})$, the probability to choose (l,m,n,o,p,q,r,s) individuals of types x_j is given by the multivariate hypergeometric sampling

```
In[13]:= bign =.; (*total pop size*)

In[14]:= (*avgpayoff[focal1_,focal2_,state_,i_]:=Σ_{k=0}^{G-1}((f1[k,state]
           (Phare1[focal1]πH (Binomial[i[[1]]+i[[2]]+i[[3]]+i[[4]],k] Binomial[i[[5]]+i[[6]]+i[[7]]+i[[8]],G-1-k])/Binomial[Total[i],G-1] +
            πS(1-Phare1[focal1]) Pcomposition[k,1,i]) + (1-f1[k,state])
           (Phare1[focal2]πH (Binomial[i[[1]]+i[[2]]+i[[3]]+i[[4]],k] Binomial[i[[5]]+i[[6]]+i[[7]]+i[[8]],G-1-k])/Binomial[Total[i],G-1] +
            πS (1-Phare1[focal2]) Pcomposition[k,2,i])));*)
```

In[15]:=
```
simplifiedavfpayoff[focal1_, focal2_, state_, i_] :=
  Sum[((πH ((Binomial[i[[1]] + i[[3]] + i[[5]] + i[[7]], k]
          Binomial[i[[2]] + i[[4]] + i[[6]] + i[[8]], G - 1 - k]) / Binomial[Total[i], G - 1])
       (f1[k, state] × Phare1[focal1] + (1 - f1[k, state]) Phare1[focal2]) +
     πS (f1[k, state] (1 - Phare1[focal1]) Pcomposition[k, 1, i] +
        (1 - f1[k, state]) × (1 - Phare1[focal2]) Pcomposition[k, 2, i]))), {k, 0, G-1}];
```

In[16]:=
```
finitepayoffstab =
  Flatten[Table[{{foc1, foc2, flags}, simplifiedavfpayoff[foc1, foc2,
        flags, {i1, i2, i3, i4, i5, i6, i7, i8}]}, {foc1, {"Hare", "Stag"}},
      {foc2, {"Hare", "Stag"}}, {flags, {1, 2}}] /.
     HeavisideTheta[0] → 1, 2] // Simplify;
```

In[17]:=
```
onlystrategies = finitepayoffstab[[All, 1]]
```

Out[17]=
```
{{Hare, Hare, 1}, {Hare, Hare, 2}, {Hare, Stag, 1}, {Hare, Stag, 2},
 {Stag, Hare, 1}, {Stag, Hare, 2}, {Stag, Stag, 1}, {Stag, Stag, 2}}
```

In[18]:=
```
payoffs = finitepayoffstab[[All, 2]];
```

Analytics

In[19]:=
```
toreplace = {0, 0, 0, 0, 0, 0, 0, 0};
```

In[20]:=
```
fastpayoffA[p_, q_, ind_, bign_] :=
  Module[{newtoreplace = ReplacePart[toreplace, {p → ind - 1, q → bign - ind}]},
   payoffs[[p]] /. {i1 → newtoreplace[[1]], i2 → newtoreplace[[2]],
     i3 → newtoreplace[[3]], i4 → newtoreplace[[4]], i5 → newtoreplace[[5]],
     i6 → newtoreplace[[6]], i7 → newtoreplace[[7]], i8 → newtoreplace[[8]]}];
fastpayoffB[p_, q_, ind_, bign_] := Module[
  {newtoreplace = ReplacePart[toreplace, {p → ind, q → bign - ind - 1}]},
  payoffs[[q]] /. {i1 → newtoreplace[[1]], i2 → newtoreplace[[2]],
    i3 → newtoreplace[[3]], i4 → newtoreplace[[4]], i5 → newtoreplace[[5]],
    i6 → newtoreplace[[6]], i7 → newtoreplace[[7]], i8 → newtoreplace[[8]]}]
```

In[22]:=
```
γ[ind_, p_, q_, ω_, bign_] := Simplify[(1 + ω fastpayoffB[p, q, ind, bign]) / (1 + ω fastpayoffA[p, q, ind, bign])];
ϕ[ω_, p_, q_, bign_] := 1 / (1 + Sum[Product[γ[j, p, q, ω, bign], {j, 1, k}], {k, 1, bign-1}]);
```

$$\gamma[ind\_, p\_, q\_, \omega\_, bign\_] := \text{Simplify}\left[\frac{1 + \omega\, \text{fastpayoffB}[p, q, ind, bign]}{1 + \omega\, \text{fastpayoffA}[p, q, ind, bign]}\right];$$

$$\phi[\omega\_, p\_, q\_, bign\_] := \frac{1}{1 + \sum_{k=1}^{bign-1} \prod_{j=1}^{k} \gamma[j, p, q, \omega, bign]};$$

In[24]:=
```
bign = 16;
fixprobs = Table[Table[Table[If[p == q, nan, ϕ[ω, p, q, bign] // N], {q, 1, 8, 1}],
    {p, 1, 8, 1}], {ω, 0.0, 1.0, 0.1}];
```

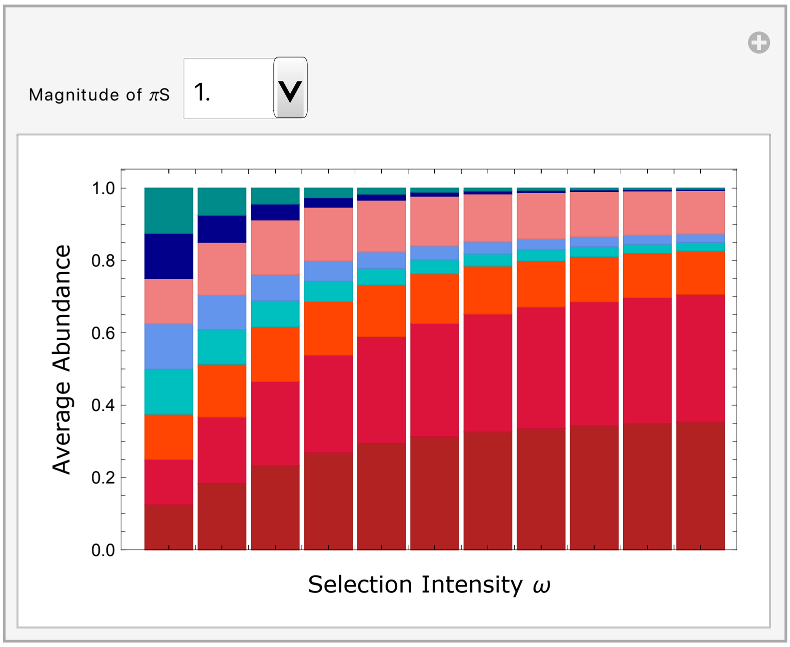
Plot against different values of πS

In[26]:=
```
fixprobsvsmagnitude = Table[fixprobs /. {πS → i}, {i, 1, 4, 0.5}];
```

In[27]:=
```
pis = Table[i, {i, 1, 4, 0.5}]
```
Out[27]=
```
{1., 1.5, 2., 2.5, 3., 3.5, 4.}
```

```
Manipulate[totest = fixprobsvsmagnitude[[Position[pis, magnitude][[1, 1]]]];
 columnstochastic = Table[
   Table[ReplacePart[totest[[j]][[i]], i → 1 - ((totest[[j]][[All, i]] // Total) - nan)],
    {i, 1, 8, 1}], {j, 1, Length[totest], 1}];
 abundances = Table[If[j ≤ Length[columnstochastic],
     Re[Eigensystem[columnstochastic[[j]], 1][[2]] // Flatten] / Total[Re[Eigensystem[columnstochastic[[j]], 1][[2]] // Flatten]],
     Re[Eigensystem[columnstochastic[[j]]][[2, 2]] // Flatten] / Total[Re[Eigensystem[columnstochastic[[j]]][[2, 2]] // Flatten]]],
   {j, 1, Length[columnstochastic], 1}];
BarChart[abundances, ChartLayout → "Stacked", ChartStyle → colors, Frame → True,
  FrameLabel → {Style["Selection Intensity ω", "Label", 12, Black],
    Style["Average Abundance", "Label", 12, Black]},
  ImageSize → Medium], {{magnitude, 1., "Magnitude of πS"}, pis}]
```

Out[28]=



In[29]:=
```
normalisedfixprobsmatrix = SetPrecision[Table[totest = fixprobsvsmagnitude[[i]];
   Table[Table[ReplacePart[totest[[j]][[i]],
      i → 1 - ((totest[[j]][[All, i]] // Total) - nan)], {i, 1, 8, 1}],
    {j, 1, Length[totest], 1}], {i, 1, Length[fixprobsvsmagnitude], 1}], 3];
```

In[30]:=
```
Manipulate[MatrixPlot[
  normalisedfixprobsmatrix[[Position[pis, magnitude][[1, 1]], k]]
  ─────────────────────────────────────────────────────────────── , Mesh → True,
                              1 / 16
  Frame → True, FrameTicks → {{{1, "H,H,1"}, {2, "H,H,2"}, {3, "H,S,1"},
     {4, "H,S,2"}, {5, "S,H,1"}, {6, "S,H,2"}, {7, "S,S,1"}, {8, "S,S,2"}}},
  ColorFunction → (ColorData["TemperatureMap"]@Rescale[#1, {0, 14}, {0, 1}] &),
  ColorFunctionScaling → False,
  Epilog → {Black, MapIndexed[Text[#1, #2 - 1 / 2] &, Transpose[
       normalisedfixprobsmatrix[[Position[pis, magnitude][[1, 1]], k]]
       ─────────────────────────────────────────────────────────────── // Reverse],
                                   1 / 16
     {2}]}, PlotLabel → Style["Fixation probability (divided by ρN)\n
      of Strategy Row into Strategy Column", "Author", 14, Black]],
 {{magnitude, 1., "Magnitude of πS"}, pis}, {{k, 1, "Selection Intensity"},
  1, Dimensions[fixprobsvsmagnitude][[2]], 1}]
```

Out[30]=

In[31]:=
```
BarLegend[{"TemperatureMap", {0, 14}}]
```

Out[31]=