EquilibriumLimitOrders

Empirical Market Microstructure
Companion Mathematica notebooks

Joel Hasbrouck
January, 2007

This notebook covers the Parlour (1998) model, discussed in section 12.2.

Initializations

```
<< BarCharts`
```

```
<< Notation`
```

The following commands define symbolizations that are convenient for labeling things.

```
Symbolize[Anything_Rule]; Symbolize[Anything_Rules];
Symbolize[Anything_Solution]; Symbolize[Anything_Solutions];
```

Notations specific to this notebook

These conform to Parlour's usage as much as possible.

```
Symbolize[n^B];
Symbolize[n^A];
Symbolize[n^B_t];
Symbolize[n^A_t];
Symbolize[n^B_t+1];
Symbolize[n^A_t+1];
Symbolize[n^B_t-1];
Symbolize[n^A_t-1];
Symbolize[N^b];
Symbolize[N^s];
```

Probability that between t and T there will be at least Nb market buy orders (or Ns market sell orders):

```
Notation[r^b[t_, nB_, nA_, Nb_] ⟺ rb[t_, nB_, nA_, Nb_]];
Notation[r^s[t_, nB_, nA_, Ns_] ⟺ rs[t_, nB_, nA_, Ns_]];
```

Probability that a limit buy order added at time t will execute.

```
Notation[p^b[t_, nB_, nA_] ⟺ pb[t_, nB_, nA_]];
Notation[p^s[t_, nB_, nA_] ⟺ ps[t_, nB_, nA_]];
```

Other constants and functions

```
Symbolize[β̄]; Symbolize[β̲];
```

```
Notation[F_β[β_] ⟺ Fβ[β_]];
Symbolize[F_β Rule];
```

β Distributions

This section defines the β distributions. Parlour uses only the uniform; the presentation also uses triangular and inverse triangular ("wing") distributions

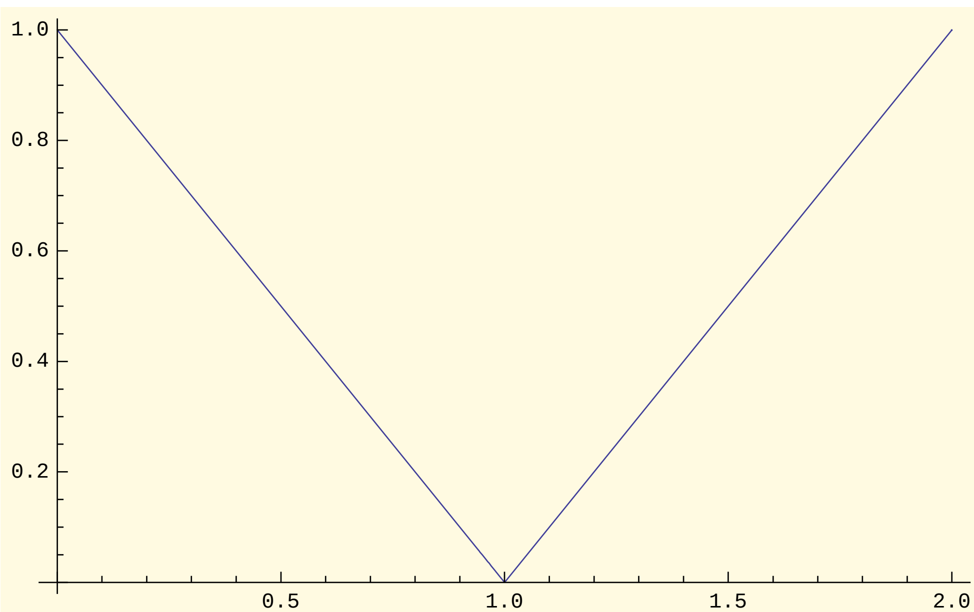
Uniform

```
d = 1; β̲ = 1 - d; β̄ = 1 + d;
F_β Uniform = F_β[β_] :> CDF[UniformDistribution[{β̲, β̄}], β];
```

Inverse Triangular ("Wing")

```
β̲ = 0; β̄ = 2;
```

```
Plot[Abs[x - 1], {x, 0, 2}]
```

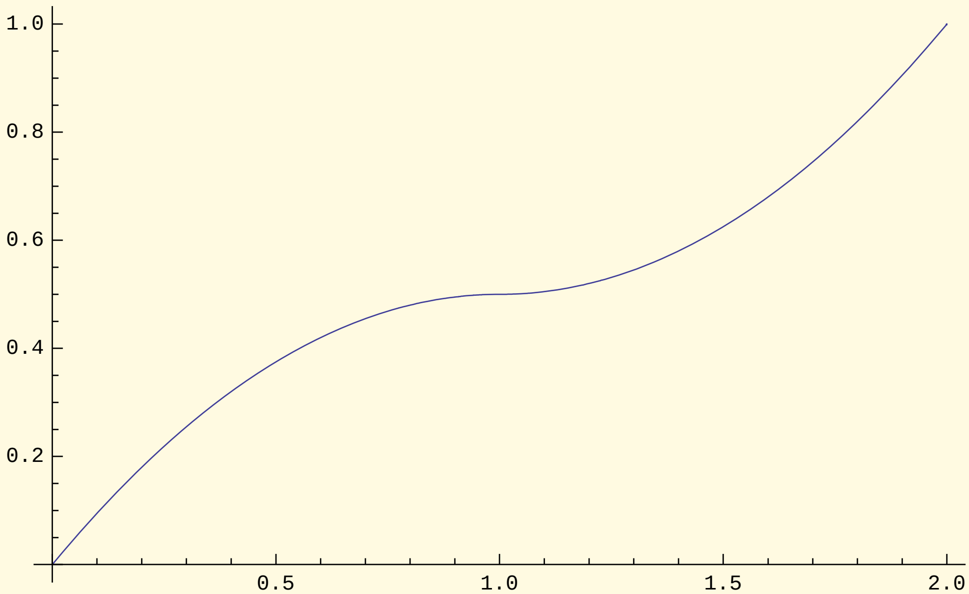


```
Integrate[Abs[x - 1], {x, 0, X}, Assumptions → {X > 0, X < 2, X ∈ Reals}]
```

$$\begin{cases} \frac{1}{2}\left(2X - X^2\right) & 0 < X \le 1 \\ \frac{1}{2}\left(2 - 2X + X^2\right) & \text{True} \end{cases}$$

```
Fβ Wing = Fβ[β_] :→ Which[β ≤ 0, 0, 0 < β ≤ 1, β - β^2/2, 1 < β < 2, 1 - β + β^2/2, β ≥ 2, 1];
```

```
Plot[Evaluate[Fβ[x] /. Fβ Wing], {x, 0, 2}, PlotRange → All]
```



Triangular

Procedures to compute and analyze equilibria

Compute equilibrium

```
pSetup := Module[{},
  (*Print["Clearing and resetting definitions for p^s and p^b..."];*)
  ClearAll[pb, ps];
  p^s[t_, n^B : _, n^A : _] := p^s[t, n^B, n^A] = r^b[t + 1, n^B, n^A + 1, n^A + 1];
  p^b[t_, n^B : _, n^A : _] := p^b[t, n^B, n^A] = r^s[t + 1, n^B + 1, n^A, n^B + 1];
 ]
```

```
βSetup := Module[{},
   (*Print["Clearing and resetting definitions for β's..."];*)
   ClearAll[βLmtSellLow, βLmtBuyHigh, βLmtSellHigh, βLmtBuyLow];
   βLmtSellLow[t_, n^B : _, n^A : _] :=
    βLmtSellLow[t, n^B, n^A] = Max[1/V (A - 1/(1 - p^s[t, n^B, n^A])), β̲];
   βLmtBuyHigh[t_, n^B : _, n^A : _] := βLmtBuyHigh[t, n^B, n^A] =
     Min[1/V (B + 1/(1 - p^b[t, n^B, n^A])), β̄];
   βLmtSellHigh[t_, n^B : _, n^A : _] := βLmtSellHigh[t, n^B, n^A] =
     1/V If[p^s[t, n^B, n^A] == 0, B, A];
   βLmtBuyLow[t_, n^B : _, n^A : _] := βLmtBuyLow[t, n^B, n^A] =
     1/V If[p^b[t, n^B, n^A] == 0, A, B];
  ];
```

```
PSetup := Module[{},
   (*Print["Clearing and resetting definitions for P..."];*)
   ClearAll[P];
   P[t_, n^B : _, n^A : _, SellMkt] := P[t, n^B, n^A, SellMkt] =
     (1 / 2) Evaluate[F_β[βLmtSellLow[t, n^B, n^A]] /. F_βRule];
   P[t_, n^B : _, n^A : _, SellLmt] := P[t, n^B, n^A, SellLmt] =
     (1 / 2)
      Evaluate[F_β[βLmtSellHigh[t, n^B, n^A]] - F_β[βLmtSellLow[t, n^B, n^A]] /. F_βRule];
   P[t_, n^B : _, n^A : _, SellOut] := P[t, n^B, n^A, SellOut] =
     (1 / 2) Evaluate[1 - F_β[βLmtSellHigh[t, n^B, n^A]] /. F_βRule];
   P[t_, n^B : _, n^A : _, BuyOut] := P[t, n^B, n^A, BuyOut] =
     (1 / 2) Evaluate[F_β[βLmtBuyLow[t, n^B, n^A]] /. F_βRule];
   P[t_, n^B : _, n^A : _, BuyLmt] := P[t, n^B, n^A, BuyLmt] =
     (1 / 2)
      Evaluate[F_β[βLmtBuyHigh[t, n^B, n^A]] - F_β[βLmtBuyLow[t, n^B, n^A]] /. F_βRule];
   P[t_, n^B : _, n^A : _, BuyMkt] := P[t, n^B, n^A, BuyMkt] =
     (1 / 2) Evaluate[1 - F_β[βLmtBuyHigh[t, n^B, n^A]] /. F_βRule];
  ];
```

```
rSetup := Module[{},
   (*Print["Clearing and resetting definitions for rs and rb..."];*)
   ClearAll[rb, rs];
   rb[t_, nB : _, nA : _, Nb : _] := rb[t, nB, nA, Nb] =
     Which[Nb ≤ 0, 1,
       (t == T) && (Nb == 1), (1 - Fβ[A / V]) / 2 /. FβRule,
       Nb > T - t + 1, 0,
       Nb ≤ 0, 1,
       True, P[t, nB, nA, SellMkt] rb[t + 1, Max[nB - 1, 0], nA, Nb] +
        P[t, nB, nA, SellLmt] rb[t + 1, nB, nA + 1, Nb] +
        P[t, nB, nA, SellOut] rb[t + 1, nB,  nA, Nb] + P[t, nB, nA, BuyOut]
          rb[t + 1, nB, nA, Nb] + P[t, nB, nA, BuyLmt]  rb[t + 1, nB + 1, nA, Nb] +
        P[t, nB, nA, BuyMkt]  rb[t + 1, nB, Max[nA - 1, 0], Nb - 1]] // N;
   rs[t_, nB : _, nA : _, Ns : _] := rs[t, nB, nA, Ns] =
     Which[Ns ≤ 0, 1,
       (t == T) && (Ns == 1), Fβ[B / V] / 2 /. FβRule,
       Ns > T - t + 1, 0,
       True, P[t, nB, nA, SellMkt] rs[t + 1, Max[nB - 1, 0], nA, Ns - 1] +
        P[t, nB, nA, SellLmt] rs[t + 1, nB, nA + 1, Ns] +
        P[t, nB, nA, SellOut] rs[t + 1, nB,  nA, Ns] + P[t, nB, nA, BuyOut]
          rs[t + 1, nB,  nA, Ns] + P[t, nB, nA, BuyLmt]  rs[t + 1, nB + 1, nA, Ns] +
        P[t, nB, nA, BuyMkt]  rs[t + 1, nB, Max[nA - 1, 0],  Ns]] // N;
  ];
```

Transition probabilities:

```
PrSetup := Module[{},
  (*Print["Clearing and resetting definitions for Pr..."];*)
  ClearAll[Pr];
  Pr[t_, {n^B_t : _, n^A_t : _}, {n^B_t+1 : _, n^A_t+1 : _}] := Pr[t, {n^B_t, n^A_t}, {n^B_t+1, n^A_t+1}] =
    Which[
     (*(n^B_t>MaxDepth) ||
      (n^A_t>MaxDepth) || (n^B_t+1>MaxDepth) ||( n^A_t+1>MaxDepth) ,0,*)
     n^B_t == 0 && n^B_t+1 == 0 && n^A_t == 0 && n^A_t+1 == 0, P[t, n^B_t, n^A_t, BuyOut] +
      P[t, n^B_t, n^A_t, SellOut] + P[t, n^B_t, n^A_t, SellMkt] + P[t, n^B_t, n^A_t, BuyMkt],
     n^B_t == 0 && n^B_t+1 == 0 && n^A_t == n^A_t+1, P[t, n^B_t, n^A_t, BuyOut] +
      P[t, n^B_t, n^A_t, SellOut] + P[t, n^B_t, n^A_t, SellMkt],
     n^B_t == n^B_t+1 && n^A_t == 0 && n^A_t+1 == 0, P[t, n^B_t, n^A_t, BuyOut] +
      P[t, n^B_t, n^A_t, SellOut] + P[t, n^B_t, n^A_t, BuyMkt],
     n^A_t == n^A_t+1 && n^B_t == n^B_t+1, P[t, n^B_t, n^A_t, BuyOut] + P[t, n^B_t, n^A_t, SellOut],
     n^B_t+1 == n^B_t + 1 && n^A_t == n^A_t+1, P[t, n^B_t, n^A_t, BuyLmt],
     n^B_t == n^B_t+1 && n^A_t+1 == n^A_t + 1, P[t, n^B_t, n^A_t, SellLmt],
     n^B_t+1 == n^B_t - 1 && n^A_t == n^A_t+1, P[t, n^B_t, n^A_t, SellMkt],
     n^B_t+1 == n^B_t && n^A_t+1 == n^A_t - 1, P[t, n^B_t, n^A_t, BuyMkt],
     True, 0];
 ];
```

```
PrStateSetup := Module[{},
  (*Print["Clearing and resetting definitions for PrState..."];*)
  ClearAll[PrState];
  PrState[t_, n^B : _, n^A : _] := PrState[t, n^A, n^B] = Which[
     t == 1 && n^A == 0 && n^B == 0, 1,
     1 < t && (n^A > t - 1 || n^B > t - 1), 0,
     n^B > MaxDepth || n^A > MaxDepth , 0,
     True,
     Sum[PrState[t - 1, n^B_t-1, n^A_t-1] Pr[t - 1, {n^B_t-1, n^A_t-1}, {n^B, n^A}],
      {n^B_t-1, Max[0, n^B - 1], Min[n^B + 1, t - 1]},
      {n^A_t-1, Max[0, n^A - 1], Min[n^A + 1, t - 1]}]
    ];
 ];
```

```
Off[General::"spell1", General::"spell"]
```

```
Off[Optional::"opdef"]
```

```
ClearAll[InitializeAll];
InitializeAll[T0_: 3, FβRule_: Fβ Uniform,
   βLow_: 0, βHigh_: 2, V0_: 11 / 2, B0_: 5, A0_: 6] := Module[{},
   Print["Initializing all probabilities and rules."];
   T = T0; V = V0 // N; B = B0 // N; A = A0 // N; Fβ Rule = FβRule; β̲ = βLow; β̄ = βHigh;
   Print["T=" , T, " Fβ Rule=", Fβ Rule,
    " β̲=", β̲, " β̄=", β̄, " V=", V, " B=", B, " A=", A];
   pSetup;
   βSetup;
   PSetup;
   rSetup;
   PrSetup;
   PrStateSetup;
  ];
```

```
On[Optional::"opdef"]
```

```
InitializeAll[]
```

Initializing all probabilities and rules.

T=3 $F_{\beta\,\mathrm{Rule}}$=$F_\beta[\beta\_]$:→ CDF[UniformDistribution[{β̲, β̄}], β] β̲=0 β̄=2 V=5.5 B=5. A=6.

```
Calculate[MaxDepth0_: 4] := Module[{},
   MaxDepth = MaxDepth0;
   (*Do[
    {P[t,n^B,n^A,SellMkt],P[t,n^B,n^A,SellLmt],P[t,n^B,n^A,SellOut],P[t,n^B,n^A,BuyOut],
     P[t,n^B,n^A,BuyLmt],P[t,n^B,n^A,BuyMkt]},
    {t,T,1,-1},
    {n^B,0,Min[MaxDepth,t-1]},
    {n^A,0,Min[MaxDepth,t-1]}];
   Do[
    Pr[t,{n_t^B,n_t^A},{n_{t+1}^B,n_{t+1}^A}],
    {t,1,T-1},
    {n_t^B,0,Min[MaxDepth,t-1]},
    {n_t^A,0,Min[MaxDepth,t-1]},
    {n_{t+1}^B,Max[n_t^B-1,0],n_t^B+1},
    {n_{t+1}^A,Max[n_t^A-1,0],n_t^A+1}];*)
   Do[PrState[t, n^B, n^A],
    {t, 1, T},
    {n^B, 0, Min[MaxDepth, t - 1]},
    {n^A, 0, Min[MaxDepth, t - 1]}];
   d = Flatten[Table[
      Sum[Pr[t, {n_t^B, n_t^A}, {n_{t+1}^B, n_{t+1}^A}],
       {n_{t+1}^B, 0, t},
       {n_{t+1}^A, 0, t}],
      {t, 1, T - 1},
      {n_t^B, 0, Min[MaxDepth, t - 1]},
      {n_t^A, 0, Min[MaxDepth, t - 1]}
     ]];
   Print["Max discrepancy in transition probabilities=", Max[Abs[d - 1]]];
   d = Flatten[Table[
      Sum[PrState[t, n^B, n^A],
       {n^B, 0, Min[t - 1, MaxDepth]},
       {n^A, 0, Min[t - 1, MaxDepth]}],
      {t, 1, T}]];
   Print["Max discrepancy in state probabilities=", Max[Abs[d - 1]]];
  ];
```

```
Timing[InitializeAll[6]; Calculate[3];]
```

Initializing all probabilities and rules.

T=6 $F_{\beta\,Rule}=F_{\beta}[\beta\_] \mapsto$ CDF[UniformDistribution[{$\underline{\beta}$, $\bar{\beta}$}], β] $\underline{\beta}$=0 $\bar{\beta}$=2 V=5.5 B=5. A=6.

Max discrepancy in transition probabilities=1.11022×10^{-16}

Max discrepancy in state probabilities=4.44089×10^{-16}

```
{0.265, Null}
```

Analyze and summarize properties of equilibrium

```
Timing[InitializeAll[3]; Calculate[];]
```

Initializing all probabilities and rules.

T=3 $F_{\beta\,Rule}=F_{\beta}[\beta\_] \mapsto$ CDF[UniformDistribution[{$\underline{\beta}$, $\bar{\beta}$}], β] $\underline{\beta}$=0 $\bar{\beta}$=2 V=5.5 B=5. A=6.

Max discrepancy in transition probabilities=0.

Max discrepancy in state probabilities=1.11022×10^{-16}

```
{0.016, Null}
```

```
DepthDistribution := Table[Sum[PrState[t, n^B, n^A], {n^B, 0, Min[MaxDepth, t]}],
    {n^A, 0, MaxDepth}, {t, 1, T}] // Chop;
```

```
DepthDistribution // TableForm
```

1	0.92517	0.886755
0	0.0748299	0.113245
0	0	0
0	0	0
0	0	0

```
PrintDepthDistribution :=
 NumberForm[TableForm[Prepend[DepthDistribution, Range[T]],
   TableSpacing → {1, 1},
   TableHeadings → {Prepend[Range[0, MaxDepth], "d\\t:"]}], {5, 3}]
```

```
PrintDepthDistribution
```

d\t:	1.000	2.000	3.000
0.000	1.000	0.925	0.887
1.000	0.000	0.075	0.113
2.000	0.000	0.000	0.000
3.000	0.000	0.000	0.000
4.000	0.000	0.000	0.000

```
ExpectedDepth := Transpose[DepthDistribution].Range[0, MaxDepth]
```

```
PrintExpectedDepth := NumberForm[TableForm[Prepend[{ExpectedDepth}, Range[T]],
   TableSpacing → {1, 1}, TableHeadings → {{"t", "E[Depth]"}}], {5, 3}]
```

```
PrintExpectedDepth
```

t	1.000	2.000	3.000
E[Depth]	0.000	0.075	0.113

```
EventProb[event_] :=
 Table[Sum[P[t, n^B, n^A, event] PrState[t, n^B, n^A], {n^B, 0, Min[MaxDepth, t]},
    {n^A, 0, Min[MaxDepth, t]}], {t, 1, T}] // N
```

```
EventProb[SellMkt]
```

```
{0.197897, 0.214904, 0.227273}
```

```
events = {SellMkt, SellLmt, SellOut, BuyOut, BuyLmt, BuyMkt};
```

```
Clear[SellMkt, SellLmt, SellOut, BuyOut, BuyLmt, BuyMkt];
```

```
PrintEventProbs := NumberForm[TableForm[
    EventProb[#] & /@ events, TableHeadings → {events, Range[T]}], {5, 3}];
```

```
PrintEventProbs
```

	1.000	2.000	3.000
SellMkt	0.198	0.215	0.227
SellLmt	0.075	0.054	0.000
SellOut	0.227	0.231	0.273
BuyOut	0.227	0.231	0.273
BuyLmt	0.075	0.054	0.000
BuyMkt	0.198	0.215	0.227

Analyzes and graphs used in book

Time profile with uniform distribution

Order types

◻ *Effects of different distributions*

β Distributions

◻ *Set base styles for screen or tiff (file) output.*

```
nominalFigureWidth = 4;
tiffResolution = 400;
nominalFontSize = 10;
imageSize = nominalFigureWidth * tiffResolution * {1, 0.625};
bStyleOut =
 {FontFamily → "Times", FontSize → Floor[nominalFontSize * tiffResolution / 72]}
```

```
{FontFamily → Times, FontSize → 55}
```

```
bStyleOut = {FontFamily → "Times", FontSize → 40};
```

```
bStyleScreen = {FontFamily → "Times", FontSize → 12, LineSpacing → {1, -3}};
```

```
bStyle = bStyleOut;
```

◻ *Uniform*

```
InitializeAll[7]; Calculate[];
```

Initializing all probabilities and rules.

T=7 $F_{\beta\,\text{Rule}}$=F_β[β_] :→ CDF[UniformDistribution[{$\underline{\beta}$, $\bar{\beta}$}], β] $\underline{\beta}$=0 $\bar{\beta}$=2 V=5.5 B=5. A=6.

Max discrepancy in transition probabilities=1.11022×10^{-16}

Max discrepancy in state probabilities=3.33067×10^{-16}

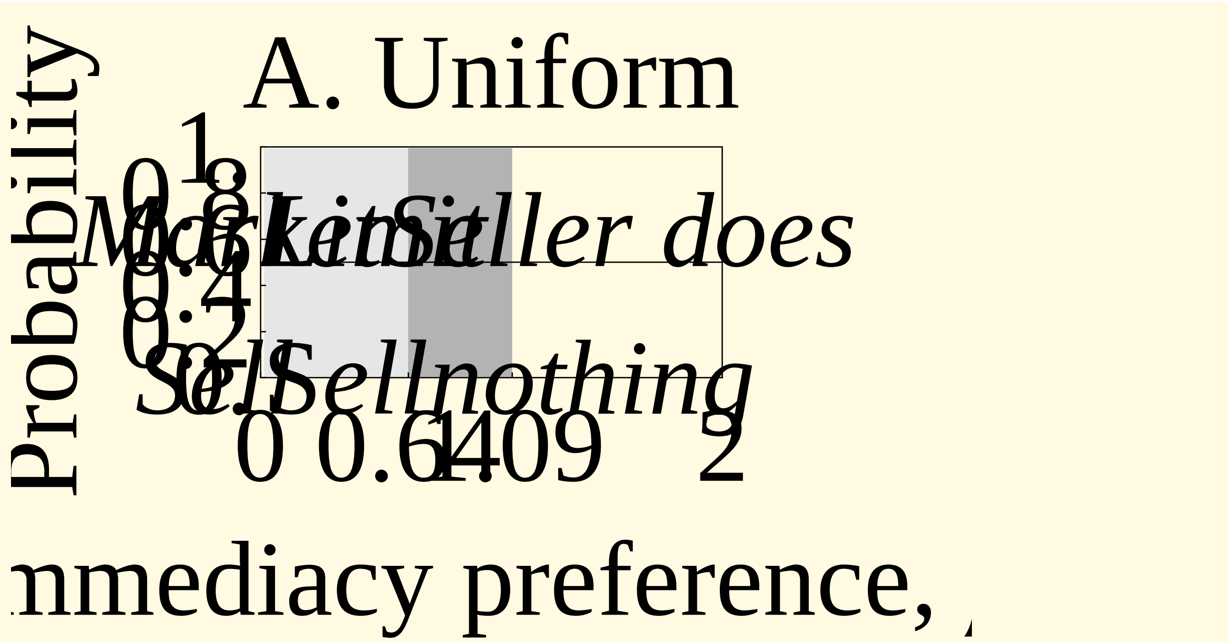
```
PrintExpectedDepth
```

t	1.000	2.000	3.000	4.000	5.000	6.000	7.000
E[Depth]	0.000	0.157	0.265	0.325	0.350	0.350	0.323

```
βC = Round[(Apply[#, {3, 0, 0}] &) /@
   {βLmtSellLow, βLmtSellHigh, βLmtBuyLow, βLmtBuyHigh}, 0.01]
```

```
{0.64, 1.09, 0.91, 1.36}
```

```
d = 0.005`;
p1 = Graphics[{GrayLevel[0.9`], Rectangle[{3 d, d}, {βC⟦1⟧, 1 - d}],
    GrayLevel[0], Text["Market\nSell ", {βC⟦1⟧ - 0.1`, 0.9` - d}, {1, 1}],
    GrayLevel[0.7`], Rectangle[{βC⟦1⟧, d}, {βC⟦2⟧, 1 - d}],
    GrayLevel[0], Text[" Limit\nSell ", {βC⟦2⟧ - 0.1`, 0.9` - d}, {1, 1}],
    Text[" Seller does\nnothing ", {1.5`, 0.9` - d}, {0, 1}],
    GrayLevel[0.`], Line[{{0, 0.5`}, {2, 0.5`}}]},
  FrameTicks → {{0, {βC⟦1⟧, βC⟦"  1.00"⟧}, {βC⟦2⟧, βC⟦"  2.00"⟧}, 2},
    Range[0, 1, 0.2`], None, None},
  FrameLabel → {"Immediacy preference, β", "Probability", "A. Uniform", None},
  Frame → True, PlotRange → {{0, 2}, {0, 1}}, BaseStyle → bStyle]
```



▱ *Triangular*

```
InitializeAll[7, F_β Triangular]; Calculate[];
```

Initializing all probabilities and rules.

T=7 $F_{\beta\,\text{Rule}}$=F_β[β_] :→ Which[$\beta \le 0$, 0, $0 < \beta \le 1$, $\frac{\beta^2}{2}$, $1 < \beta \le 2$, $-1 + 2\beta - \frac{\beta^2}{2}$, $\beta > 2$, 1]

$\underline{\beta}$=0 $\bar{\beta}$=2 V=5.5 B=5. A=6.

Max discrepancy in transition probabilities=2.22045×10^{-16}

Max discrepancy in state probabilities=2.22045×10^{-16}

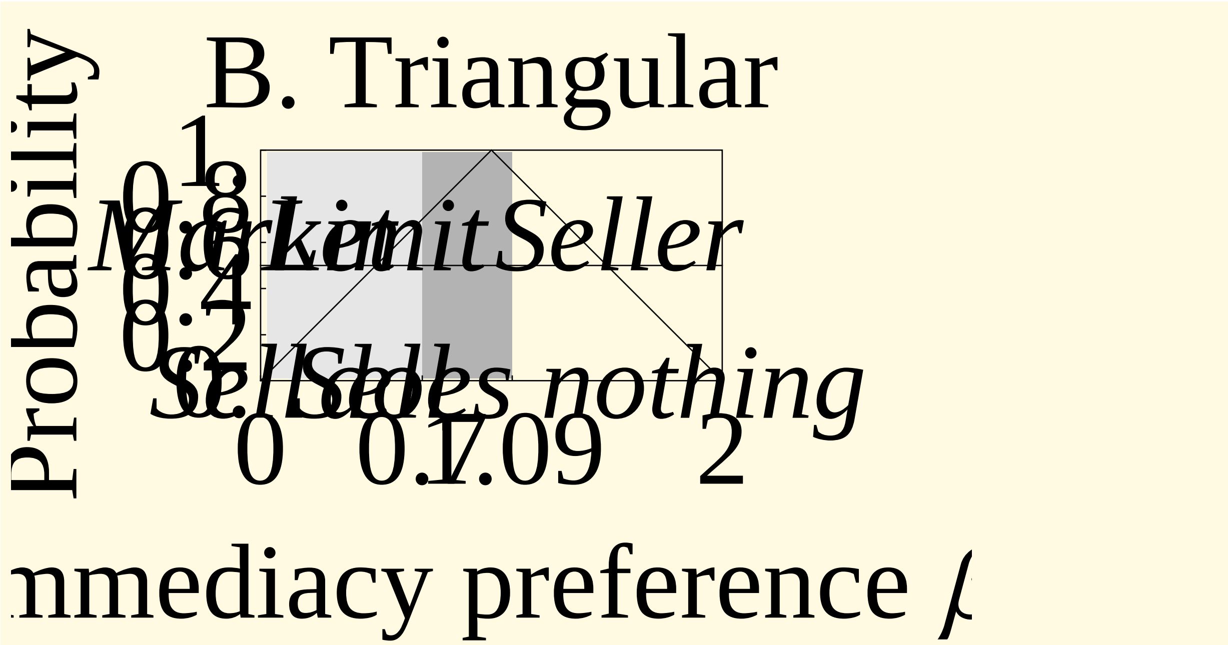
```
PrintExpectedDepth
```

```
t         1.000 2.000 3.000 4.000 5.000 6.000 7.000
E[Depth]  0.000 0.208 0.370 0.477 0.540 0.565 0.534
```

```
d = 0.01;
h = 0.25;
βC = Round[(Apply[#, {3, 0, 0}] &) /@
   {βLmtSellLow, βLmtSellHigh, βLmtBuyLow, βLmtBuyHigh}, 0.01]
```

```
{0.7, 1.09, 0.91, 1.3}
```

```
p2 = Graphics[{GrayLevel[0.9`], Rectangle[{3 d, d}, {βC[[1]], 1 - d}],
    GrayLevel[0], Text["Market\nSell ", {βC[[1]] - 0.1`, 0.9` - d}, {1, 1}],
    GrayLevel[0.7`], Rectangle[{βC[[1]], d}, {βC[[2]], 1 - d}],
    GrayLevel[0], Text["Limit\nSell", {βC[[2]] - 0.1`, 0.9` - d}, {1, 1}],
    Text[" Seller\ndoes nothing ", {1.5`, 0.9` - d}, {0, 1}], GrayLevel[0.`],
    Line[{{0, 0.5`}, {2, 0.5`}}], GrayLevel[0.`], Line[{{0, 0}, {1, 1}, {2, 0}}]},
  FrameTicks → {{0, {βC[[1]], βC[["  1.00"]]}, {βC[[2]], βC[["  2.00"]]}, 2},
    Range[0, 1, 0.2`], None, None},
  FrameLabel → {"Immediacy preference β ", "Probability", "B. Triangular", None},
  Frame → True, PlotRange → {{0, 2}, {0, 1}}, BaseStyle → bStyle]
```



◻ *Inverted triangular*

```
InitializeAll[7, Fβ Wing]; Calculate[];
```

Initializing all probabilities and rules.

T=7 $F_{\beta\,Rule}$=F_β[β_] :→ Which[$\beta \leq 0$, 0, $0 < \beta \leq 1$, $\beta - \frac{\beta^2}{2}$, $1 < \beta < 2$, $1 - \beta + \frac{\beta^2}{2}$, $\beta \geq 2$, 1]

$\underline{\beta}$=0 $\bar{\beta}$=2 V=5.5 B=5. A=6.

Max discrepancy in transition probabilities=1.11022×10^{-16}

Max discrepancy in state probabilities=4.44089×10^{-16}

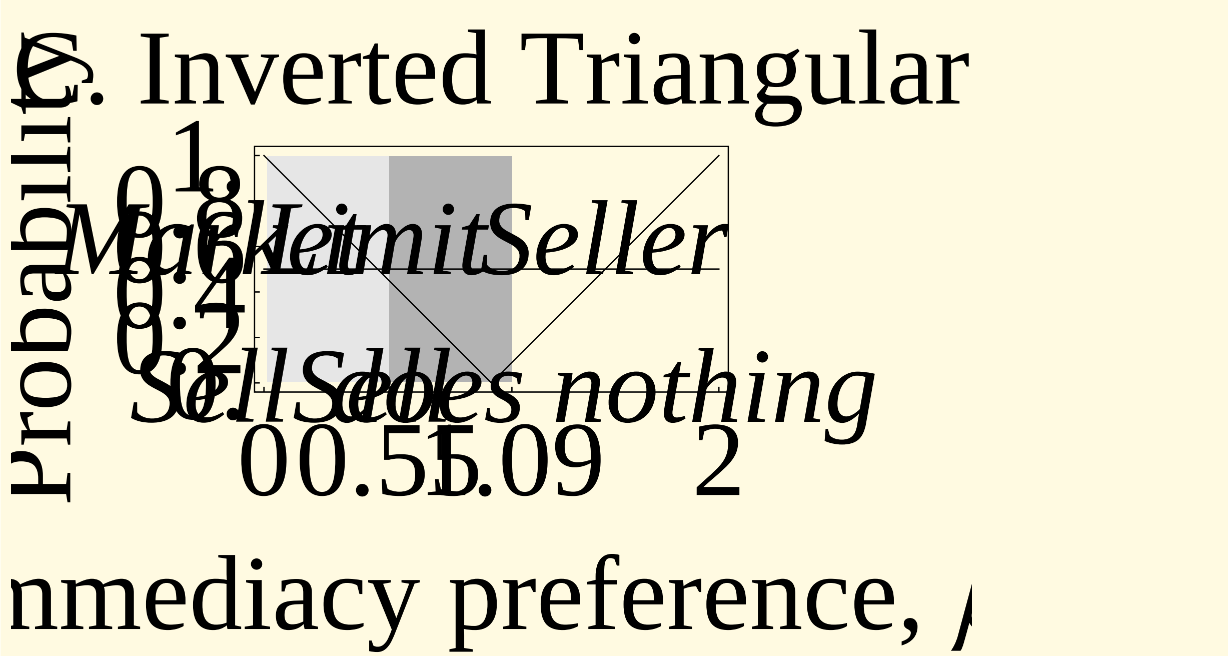
```
PrintExpectedDepth
```

t	1.000	2.000	3.000	4.000	5.000	6.000	7.000
E[Depth]	0.000	0.126	0.187	0.194	0.176	0.148	0.119

```
d = 0.005;
βC = Round[(Apply[#, {3, 0, 0}] &) /@
   {βLmtSellLow, βLmtSellHigh, βLmtBuyLow, βLmtBuyHigh}, 0.01]
```

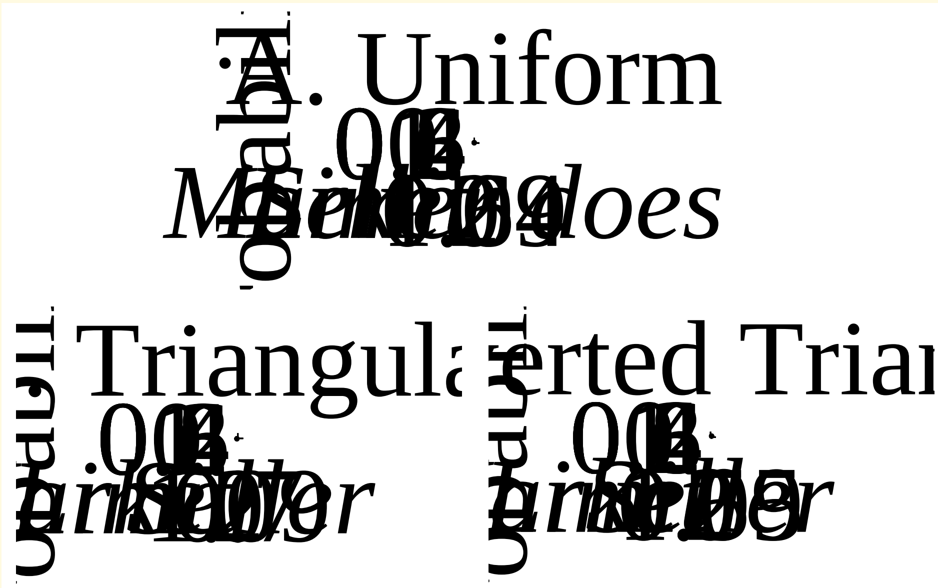
{0.55, 1.09, 0.91, 1.45}

```
p3 = Graphics[{GrayLevel[0.9`], Rectangle[{3 d, d}, {βC⟦1⟧, 1 - d}],
    GrayLevel[0], Text["Market\nSell", {βC⟦1⟧ - 0.1`, 0.9` - d}, {1, 1}],
    GrayLevel[0.7`], Rectangle[{βC⟦1⟧, d}, {βC⟦2⟧, 1 - d}],
    GrayLevel[0], Text["Limit\nSell", {βC⟦2⟧ - 0.1`, 0.9` - d}, {1, 1}],
    Text["Seller\ndoes nothing", {1.5`, 0.9` - d}, {0, 1}], GrayLevel[0.`],
    Line[{{0, 0.5`}, {2, 0.5`}}], GrayLevel[0.`], Line[{{0, 1}, {1, 0}, {2, 1}}]},
  FrameTicks → {{0, {βC⟦1⟧, βC⟦"  1.00"⟧}, {βC⟦2⟧, βC⟦"  2.00"⟧}, 2},
    Range[0, 1, 0.2`], None, None}, FrameLabel →
   {"Immediacy preference, β ", "Probability", "C. Inverted Triangular", None},
  Frame → True, BaseStyle → bStyle]
```



▢ *Output to graphics file*

```
g = GraphicsGrid[{{p1, SpanFromLeft}, {p2, p3}},
  Background → GrayLevel[1], BaseStyle → bStyle]
```



```
SetDirectory[
  "c:/Active/Empirical Market Microstructure/Mathematica/Spring 2007"];
```

```
Export["Figure124.tiff", g, "TIFF", ImageSize → imageSize,
  ColorSpace → GrayLevel, ImageResolution → tiffResolution];
```