

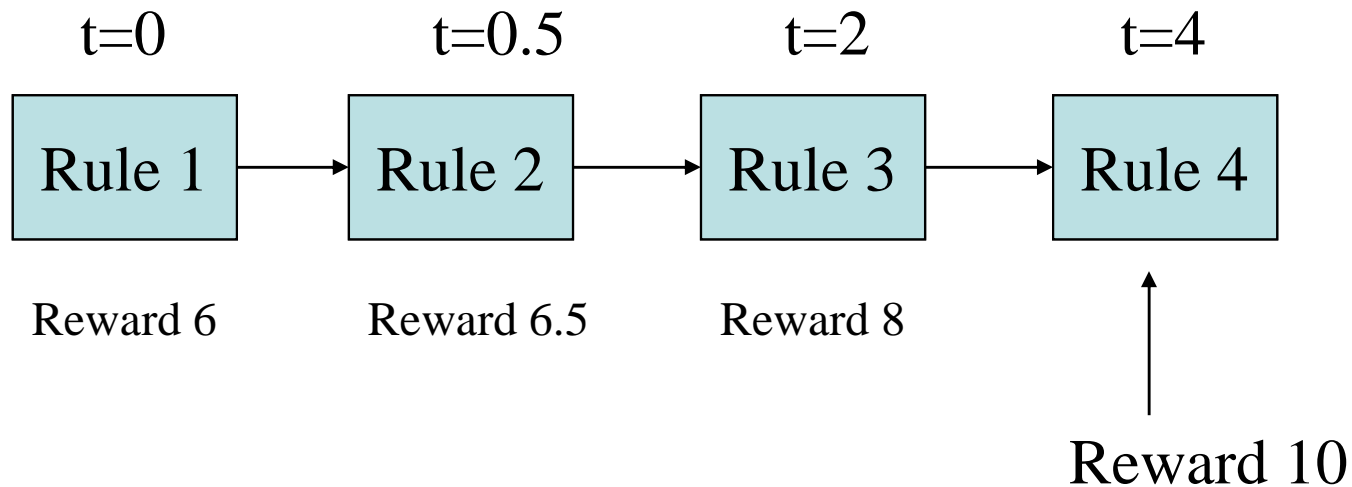
Unit 6: Subsymbolic procedural knowledge

Utility of Production rules

- Each production has a utility parameter that is an estimate of the reward it will get minus the time it takes to get to the reward

Production Utility learning

- Utility = Expected Reward of a Rule, discounted by time



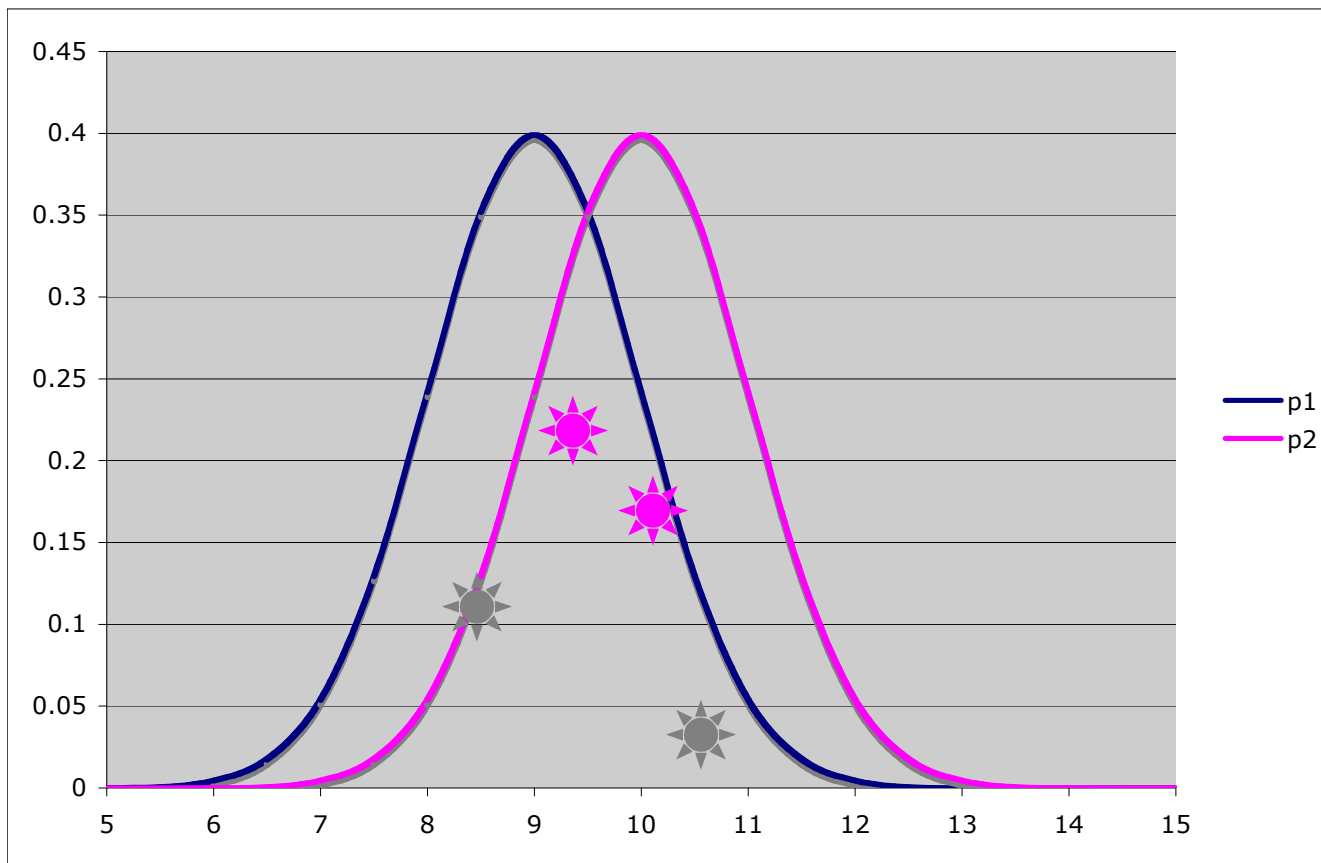
Utility is used for conflict resolution

- Similar to activation, the rule with the highest utility has the highest probability to be selected

$$P(\text{rule}_i) = \frac{e^{U_i / \sqrt{2s}}}{\sum_k e^{U_k / \sqrt{2s}}}$$

The parameter s is set by (sgp :**egs**)

Example for two rules with Utilities 9 and 10



Example of production Utility:

Probability matching

- In the next ten slides, you will see two buttons.
- “Press” one of these buttons
- If you have pressed the correct button, you will receive a “reward”, otherwise nothing
- The “correct” button is determined more or less random

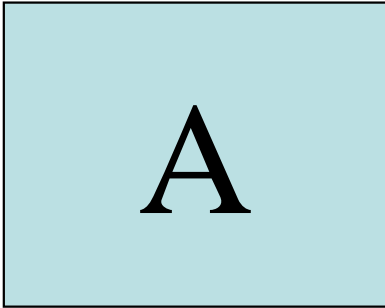
Choose a button



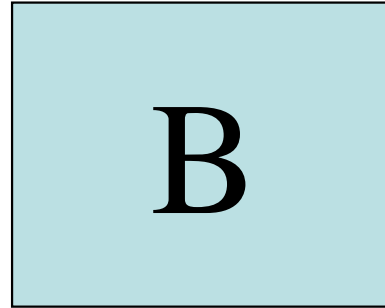
A



B



Nothing



Reward!

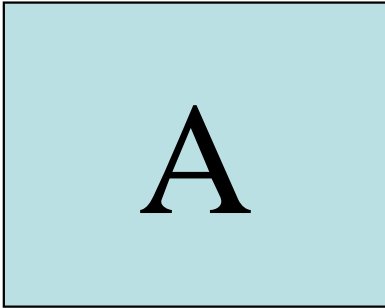
Choose a button



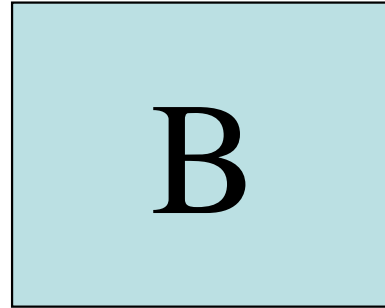
A



B



Nothing



Reward!

Choose a button



A



B



A

Reward!



B

Nothing

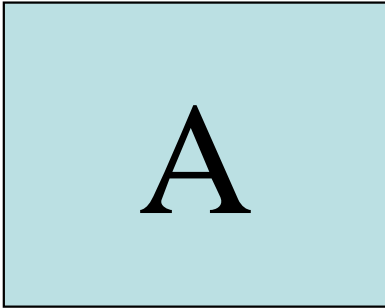
Choose a button



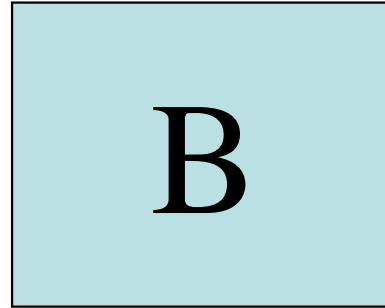
A



B



Reward!



Nothing

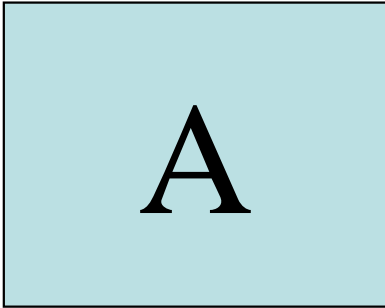
Choose a button



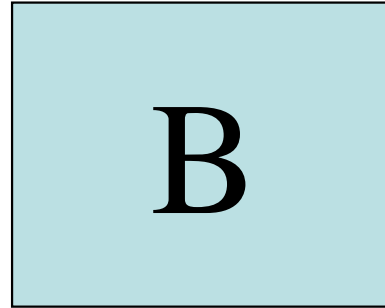
A



B



Nothing



Reward!

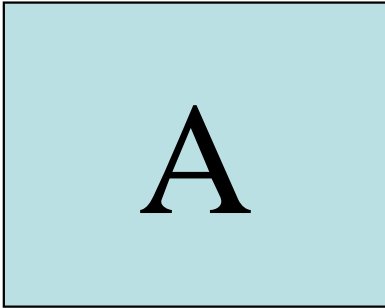
Choose a button



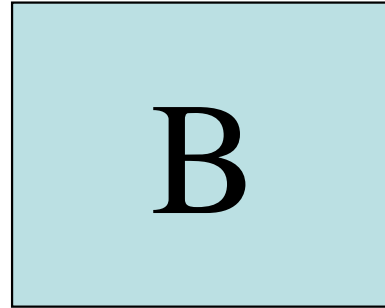
A



B



Nothing



Reward!

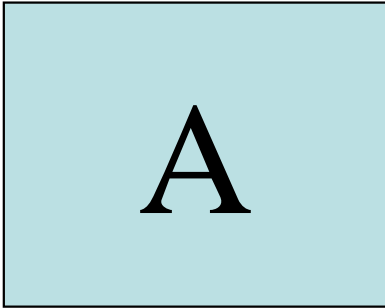
Choose a button



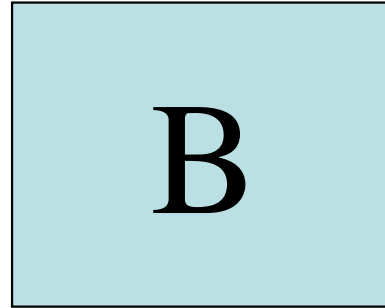
A



B



Nothing



Reward!

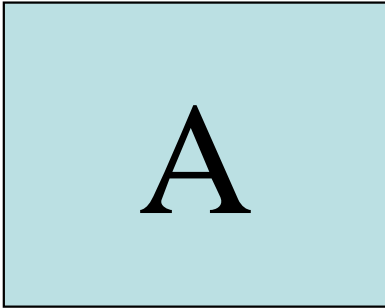
Choose a button



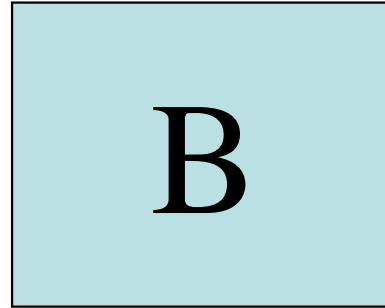
A



B



Nothing



Reward!

Choose a button



A



B



A

Reward!



B

Nothing

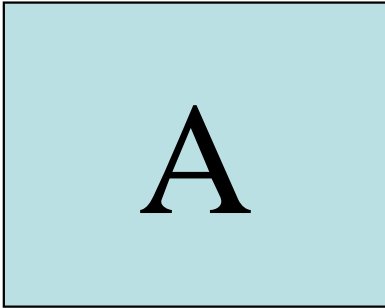
Choose a button



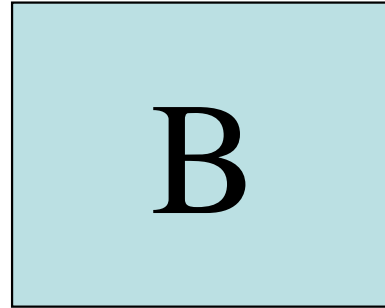
A



B



Reward!



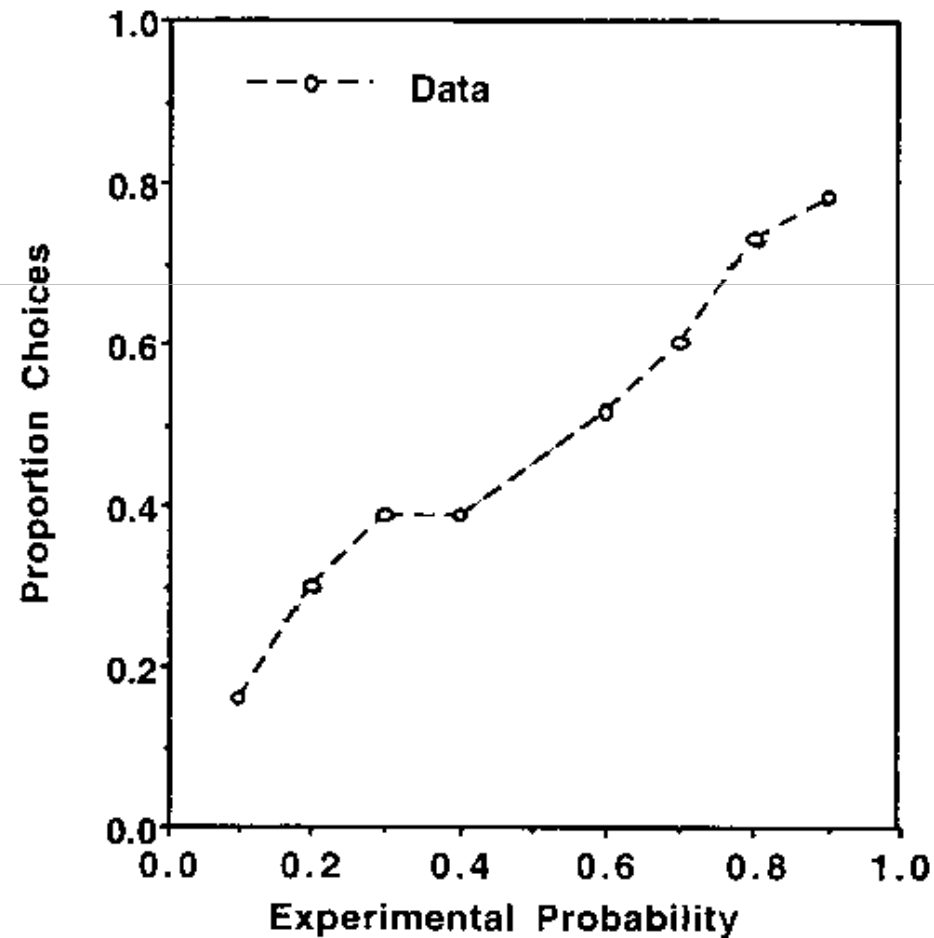
Nothing

What was the best button?

Probabilities in this experiment

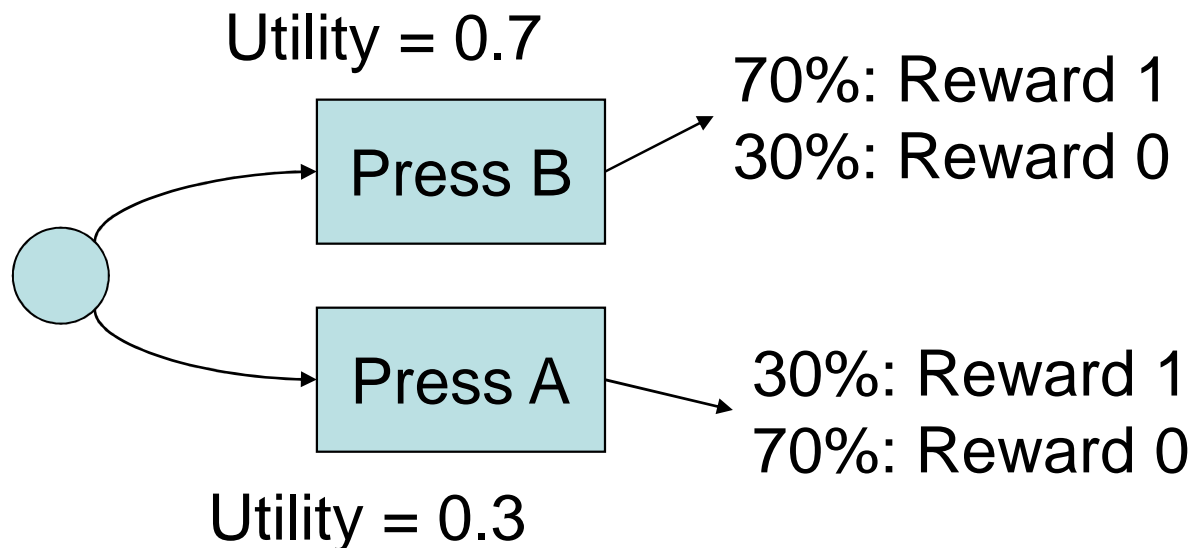
- The probability for A is 30%, and the probability for B is 70%
- From the 10 trials just shown, 6 were B and 4 were A (because the selection process was random)
- In the real experiment, subjects converge to a certain percentage of choosing A and B depending on the probability of finding the reward there

Results of experiment (Friedman)



Model of probability matching

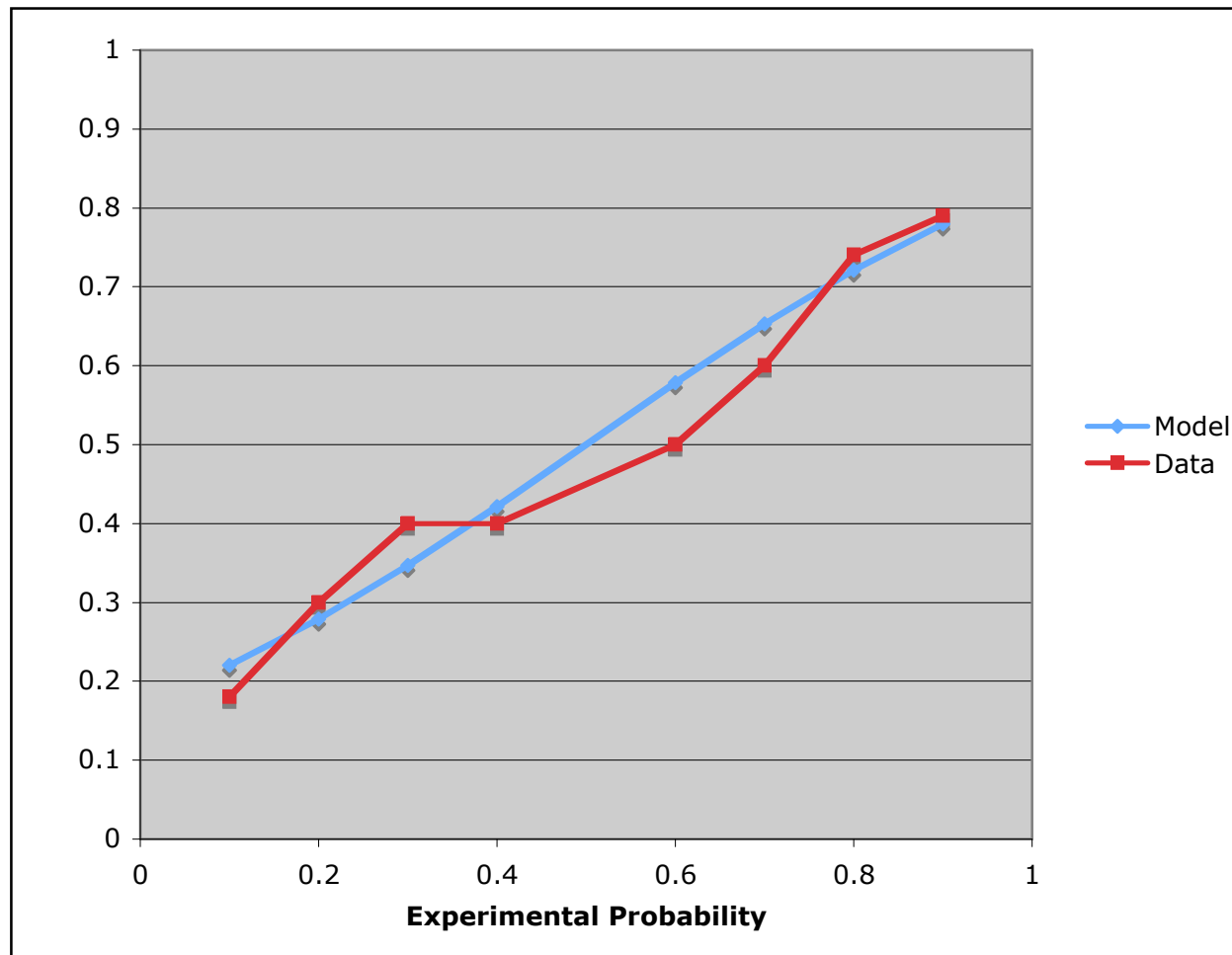
- Two (competing) production rules:
 - Press button A
 - Press button B



Probability equation

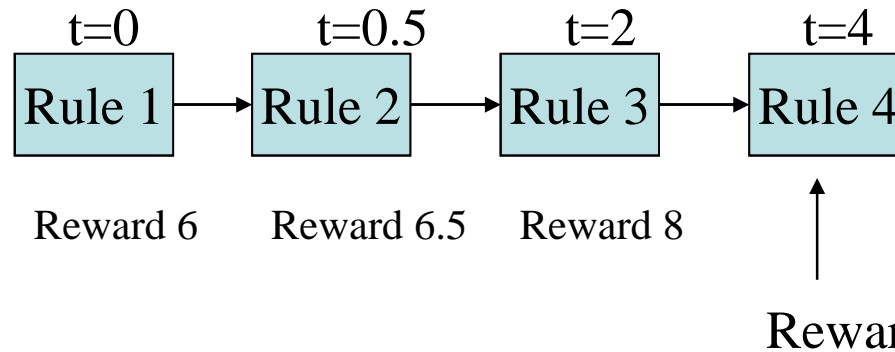
$$P(\text{rule}_A) = \frac{e^{U_A/\sqrt{2s}}}{e^{U_A/\sqrt{2s}} + e^{U_B/\sqrt{2s}}}$$

Results of model



Learning production rule parameters

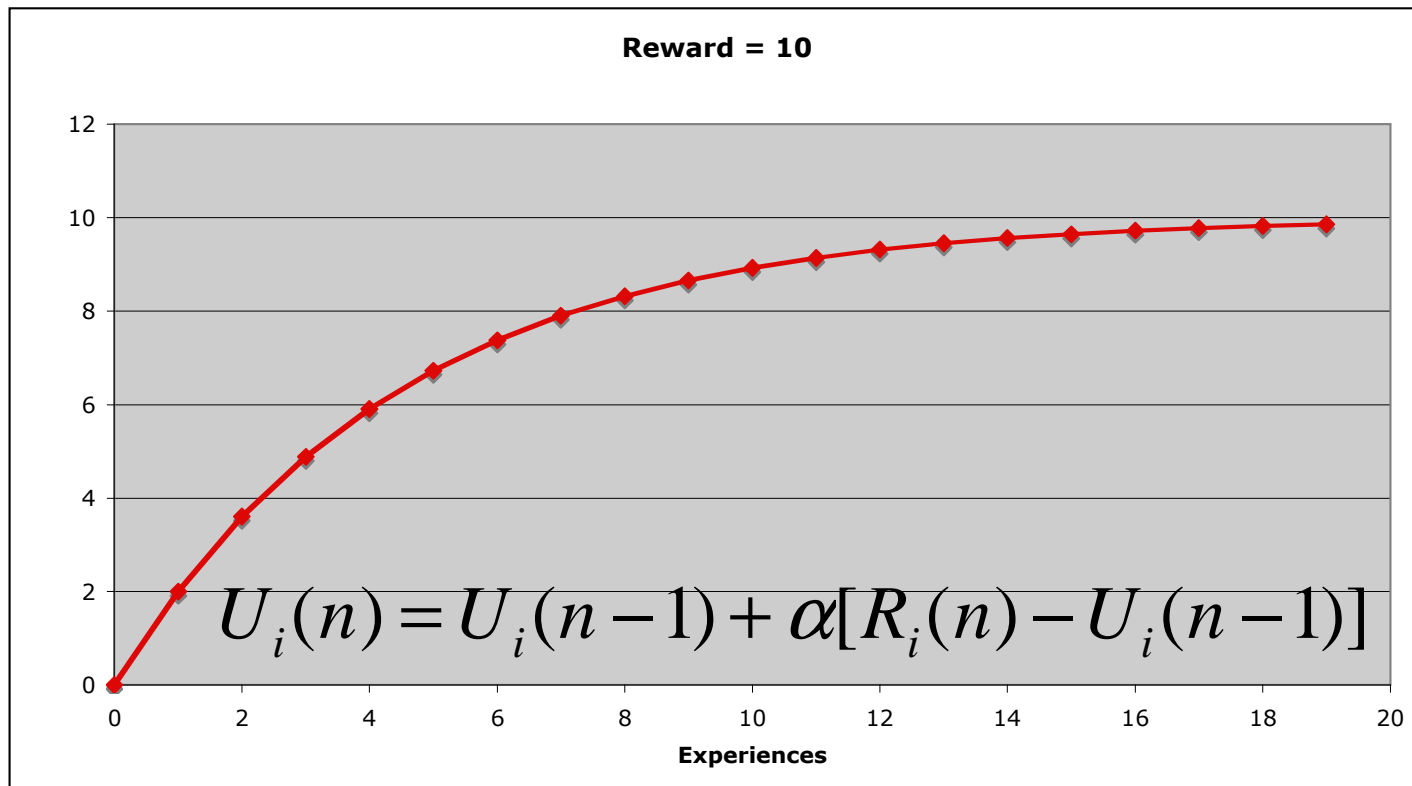
- Some productions are associated with a reward

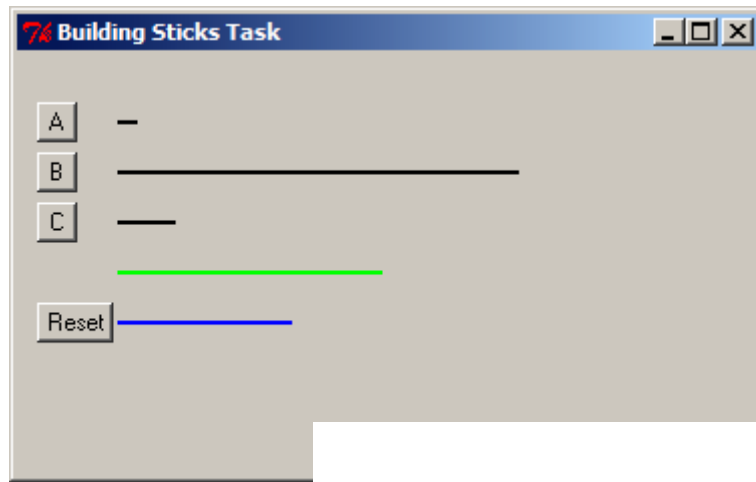


- Production rule utility is learned on the basis that reward

$$U_i(n) = U_i(n-1) + \alpha[R_i(n) - U_i(n-1)]$$

Example



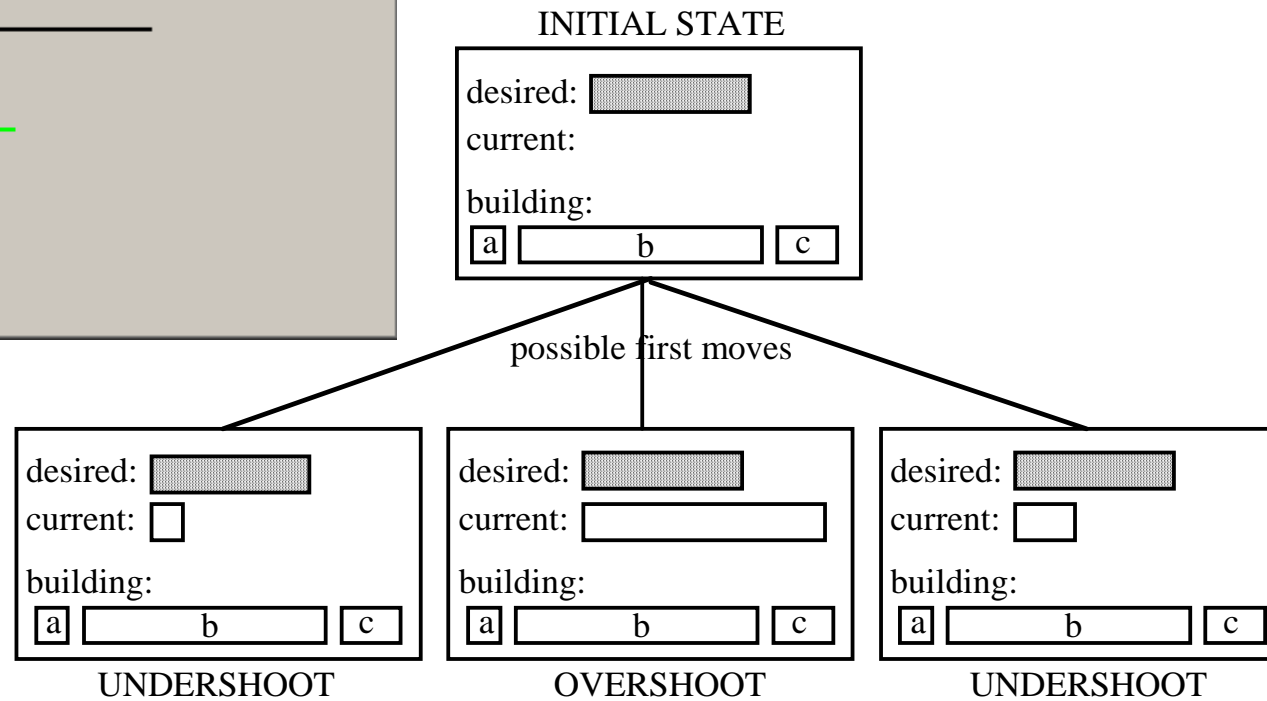
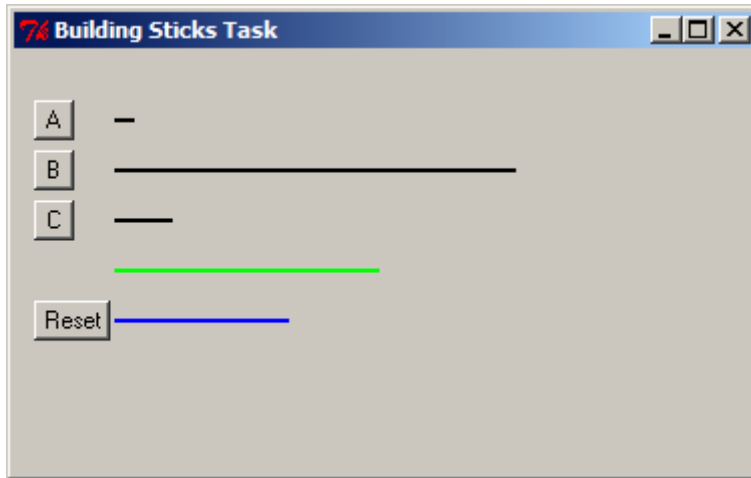


INITIAL STATE

desired:

current:

building:



$A = 15 \quad B = 200 \quad C = 41 \quad \text{Goal} = 103$

$A = 10 \quad B = 200 \quad C = 29 \quad \text{Goal} = 132$

Rules to decide strategy (1)

```
(p force-under
  =goal>
    isa      try-strategy
    state    choose-strategy
  - strategy under
==>
  =goal>
    state    prepare-mouse
    strategy under
+visual-location>
  isa      visual-location
  kind     oval
  screen-y 85)
```

```
(p force-over
  =goal>
    isa      try-strategy
    state    choose-strategy
  - strategy over
==>
  =goal>
    state    prepare-mouse
    strategy over
+visual-location>
  isa      visual-location
  kind     oval
  screen-y 60)
```

Rules to decide strategy (2)

```
(p decide-under
  =goal>
    isa      try-strategy
    state    choose-strategy
    strategy nil
    under    =under
    over     =over
!eval! (< =under(- =over 25))
==>
  =goal>
    state    prepare-mouse
    strategy under
+visual-location>
  isa      visual-location
  kind     oval
  screen-y 85)
```

```
(p decide-over
  =goal>
    isa      try-strategy
    state    choose-strategy
    strategy nil
    under    =under
    over     =over
!eval! (< =over(- =under 25))
==>
  =goal>
    state    prepare-mouse
    strategy over
+visual-location>
  isa      visual-location
  kind     oval
  screen-y 60)
```

Initial utilities

(spp decide-over :u 13)

(spp decide-under :u 13)

(spp force-over :u 10)

(spp force-under :u 10)

Setting rewards:

- Success: (spp read-done :reward 20)
- Failure: (spp pick-another-strategy :reward 0)

Sample Calculation

Before

(spp decide-over :u 13)

(spp decide-under :u 13)

(spp force-over :u 10)

(spp force-under :u 10)

$$\Pr(\textit{Over} \mid \textit{Under}) =$$

After

DECIDE-OVER : 13.1984

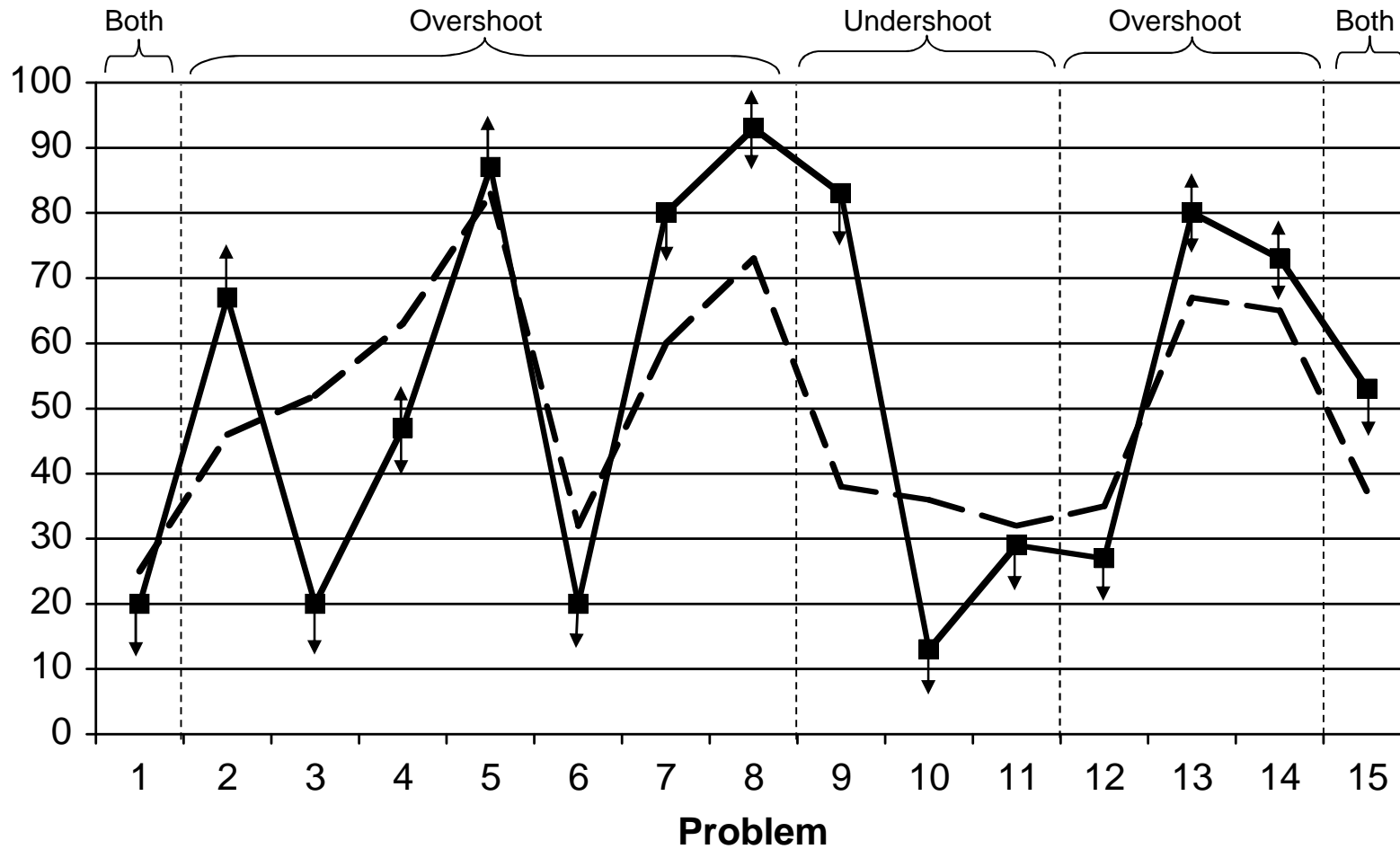
DECIDE-UNDER: 11.3041

FORCE-OVER : 12.0823

FORCE-UNDER : 6.5455

$$\Pr(\textit{Over} \mid \textit{Under}) =$$

Experiment & Results



Unit exercise

- Model the *learning* of probability matching

Unit Assignment: Probability matching with learning

$P(A) = .90$	Proportion of selecting option A
Trial 1-12	.66
Trial 13-24	.78
Trial 25-36	.82
Trial 37-48	.84

Solution

1. Write code to run single trial.
2. Write model.
3. Write code to collect block of 12 and then 4 blocks and then n subjects.
4. Do parameter search -- can do pretty well just trying to determine value of a success and use default parameters for remainder.
5. Remember to mark a reward production.