Week 8
Computational Level
Computational level

- Actions
  - hierarchical
  - goal-oriented
- Representations
  - Petri Nets
- If we have time
  - Grammars
Actions

• Hierarchy present in humans
  – e.g. reflexes
  – plans are usually hierarchical

• Rod Brooks
  – hierarchical action system
  – goal-oriented
  – different levels and components interact

• e.g. exploring behavior versus safety behavior
X-Schemas and Petri Nets

- Petri nets
  - Finite State Machines – but better!
  - Places
    - hold tokens
    - have semantic meaning
  - Transitions
    - can be enabled
    - can fire
      - consume tokens at inputs
Petri Nets

• Asynchronous
  – any enabled transition can fire
  – or not fire
  – so we reason about what states are possible

• Analysis
  – determine what states are possible
  – determine how many times a transition might fire
  – determine whether deadlock is possible
X-Schemas

• Add several things
  – timed
  – stochastic
  – inhibitory arcs
  – enabling arcs
Stochastic petri nets

Diagram:
- Place 1
- Transition t1
- Place 0
- Transition t2
- Place 2
Stochastic petri nets

• Random timing of transition firing
  – exponential distribution
  – gives rise to random choice of which transition will fire
  – $P(\text{transition fires in the next tiny time} | \text{enabled}) = f(\text{transition})$
  – Then $P(\text{transition fires next} | \text{enabled}) = f(\text{transition}) / \text{sum}(f(\text{enabled transitions}))$
Stochastic petri net example

- What is $P(t_3\text{ ever fires})$?
  - Under what conditions will it fire?
  - What is $P(t_3\text{ fires}|\text{token in } p_1)$?
  - What causes there to be a token in $p_1$?
  - What is $P(t_1\text{ fires})$?
  - How do you combine
X-Schemas

- Active representation
- Has hierarchical actions
  - defined by network structure
- Actions have structure (e.g. ready, iterating, ongoing, failed, complete)
  - defined by network structure
- Properly-designed nets will be goal-directed
  - take best actions to reach goal, given current context