

Trí Tuệ Nhân Tạo

Nguyễn Nhật Quang

quangnn-fit@mail.hut.edu.vn

Viện Công nghệ Thông tin và Truyền thông

Trường Đại học Bách Khoa Hà Nội

Năm học 2009-2010

Nội dung môn học:

- Giới thiệu về Trí tuệ nhân tạo
- Tác tử
- Giải quyết vấn đề: Tìm kiếm, Thỏa mãn ràng buộc
- Logic và suy diễn
- Biểu diễn tri thức
- Suy diễn với tri thức không chắc chắn
- Học máy
- **Lập kế hoạch**

Lập kế hoạch của con người (1)

- Hành động mà không có kế hoạch (rõ ràng)
 - Đối với các *mục tiêu gần (tức thì)*
 - Ví dụ: Bật máy tính lên
 - Đối với các hành động (hoạt động) *đã được huấn luyện (tập luyện) kỹ*
 - Ví dụ: Lái xe ô-tô
 - Khi một quá trình hành động *có thể được thay đổi thoải mái*
 - Ví dụ: Đi mua sắm ở siêu thị

Lập kế hoạch của con người (2)

■ Hành động theo kế hoạch

- ❑ Khi giải quyết một tình huống mới – Vd: Chuyển nhà (đến một nơi ở mới)
- ❑ Đối với các nhiệm vụ (công việc) phức tạp – Vd: Xây dựng kế hoạch học tập cho một học kỳ
- ❑ Khi môi trường hoạt động có chứa nguy cơ rủi ro/chi phí cao – Vd: Quản lý nhà máy năng lượng hạt nhân
- ❑ Khi cần hoạt động cộng tác với những người khác – Vd: Xây dựng một ngôi nhà

■ Con người thường chỉ lập kế hoạch khi thực sự cần thiết

- ❑ Bởi vì việc lập kế hoạch là phức tạp và tốn thời gian (chi phí vs. lợi ích)
- ❑ Con người thường chỉ cần các kế hoạch tốt hơn là tối ưu

Lập kế hoạch của máy tính

- Lập kế hoạch (Planning) của máy tính
 - Là quá trình tính toán của máy tính để lựa chọn và tổ chức các hành động, dựa trên dự đoán (trước) các kết quả của các hành động đó
 - Nhằm mục đích đạt được một số mục tiêu (objectives) xác định
 - Lập kế hoạch tối ưu: Đạt được các mục tiêu một cách tối ưu nhất
- Các kiểu bài toán lập kế hoạch
 - Với các ràng buộc về thời gian (Time constraints)
 - Với các ràng buộc về thời gian và tài nguyên (Time and resource constraints) – Bài toán lập lịch (Scheduling)
 - Với các ràng buộc về thời gian và tài nguyên, các điều kiện áp dụng (applicability conditions) của các hành động, và các tác động (effects) của các hành động đối với trạng thái của bài toán

Mô hình khái niệm

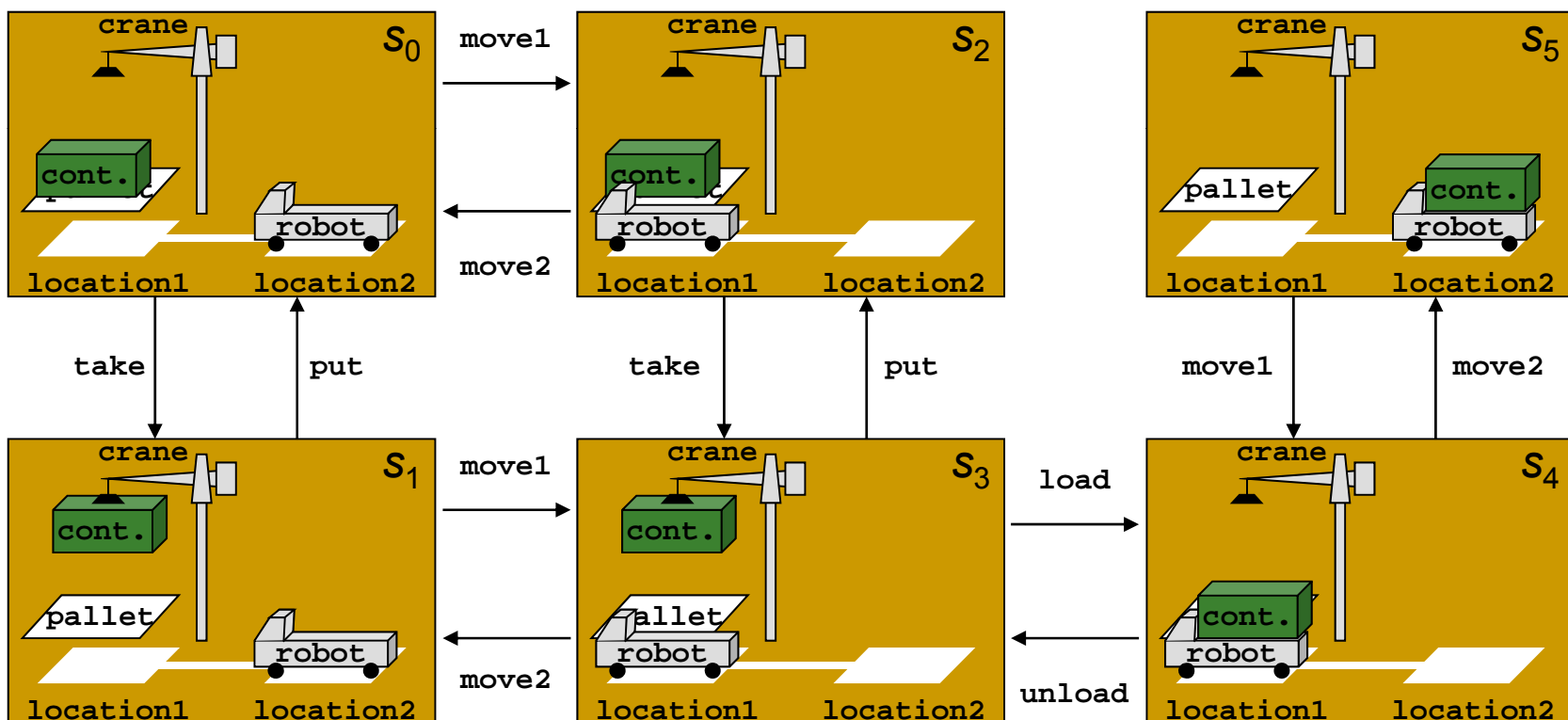
- **Mô hình khái niệm (Conceptual model)** cho bài toán lập kế hoạch được biểu diễn bằng một hệ chuyển trạng thái
- **Một hệ chuyển trạng thái (State-transition system)** được biểu diễn bởi 4 thành phần: $\Sigma = (S, A, E, \gamma)$
 - $S = \{s_1, s_2, \dots\}$: Tập hữu hạn *các trạng thái (states)* có thể của bài toán
 - $A = \{a_1, a_2, \dots\}$: Tập hữu hạn *các hành động (actions)* có thể được thực hiện
 - $E = \{e_1, e_2, \dots\}$: Tập hữu hạn *các sự kiện (events)* có thể xảy ra trong quá trình thực hiện kế hoạch
 - $\gamma: S \times A \times E \rightarrow S$: *Hàm chuyển trạng thái (a state-transition function)*

Biểu diễn bằng đồ thị

- Một hệ chuyển trạng thái $\Sigma = (S, A, E, \gamma)$ có thể được biểu diễn bằng một đồ thị có nhãn và có hướng (a directed labelled graph) $G = (N_G, E_G)$, trong đó:
 - Mỗi nút biểu diễn một trạng thái trong tập S
 - Tức là: $N_G \equiv S$
 - Mỗi cạnh liên kết từ s đến s' ($s, s' \in N_G$) có nhãn là (a, e) với $a \in A$, $e \in E$
 - Tức là: Một cạnh $(s, (a, e), s') \in E_G$, nếu và chỉ nếu $s' \in \gamma(s, a, e)$

Biểu diễn bằng đồ thị – Ví dụ

- Các trạng thái S : $s_0 \dots s_5$ (phụ thuộc vào các đối tượng: rô-bốt, cần trục, thùng hàng, ...)
- Các hành động A : cần trục nhắc (đặt) thùng hàng khỏi (lên) tấm nâng, rô-bốt di chuyển giữa 2 vị trí, cần trục nhắc (đặt) thùng hàng khỏi (lên) rô-bốt
- Không có sự kiện $E = \emptyset$



(<http://www.inf.ed.ac.uk/teaching/courses/plan/>)

Hệ chuyển trạng thái

- Chuyển trạng thái: $\gamma(s,a,e)$
- **Sự kiện trung tính (neutral event)**, được ký hiệu là ε , được dùng để biểu diễn đối với các chuyển trạng thái gây nên bởi chỉ các hành động
 - Chuyển trạng thái $\gamma(s,a,\varepsilon)$, với $s \in S$, $a \in A$: được ký hiệu (ngắn gọn) bằng $\gamma(s,a)$
- **Hành động trung tính (neutral action)**, được ký hiệu là **no-op**, được dùng để biểu diễn đối với các chuyển trạng thái gây nên bởi chỉ các sự kiện
 - Chuyển trạng thái $\gamma(s,\text{no-op},e)$, với $s \in S$, $e \in E$: được ký hiệu (ngắn gọn) bằng $\gamma(s,e)$
- Trong biểu diễn bằng đồ thị
 - Nhãn (a,ε) được ký hiệu ngắn gọn bằng a
 - Nhãn $(\text{no-op},e)$ được ký hiệu ngắn gọn bằng e

Các hành động vs. Các sự kiện

- Cả hành động (action) và sự kiện (event) đều ảnh hưởng đến sự tiến triển của một hệ chuyển trạng thái
- **Các hành động:** Thể hiện các chuyển trạng thái **được điều khiển bởi bộ thực hiện kế hoạch (plan executor)**
 - Nếu hành động $a \in A$ và $\gamma(s, a, \varepsilon) \neq \emptyset$, thì a được gọi là có thể áp dụng được đối với trạng thái s
 - Áp dụng hành động a ở trạng thái s sẽ làm hệ thống chuyển đến trạng thái $s' \in \gamma(s, a)$
- **Các sự kiện:** Thể hiện các chuyển trạng thái **xảy ra ngẫu nhiên**
 - Nếu sự kiện $e \in E$ và $\gamma(s, \text{no-op}, e) \neq \emptyset$, thì e được gọi là có thể xảy ra đối với trạng thái s
 - Sự xuất hiện (xảy ra) sự kiện e ở trạng thái s sẽ làm hệ thống chuyển đến trạng thái $s' \in \gamma(s, e)$

Kế hoạch và Mục tiêu

- **Kế hoạch (Plan):** Là một cấu trúc (chuỗi có thứ tự) các hành động phù hợp cần thực hiện để đạt được một mục tiêu từ một trạng thái ban đầu
- Các kiểu mục tiêu (Objective)
 - Trạng thái đích (goal state), hoặc một tập các trạng thái đích
 - Thỏa mãn một tập các điều kiện (conditions) đối với chuỗi các trạng thái đã duyệt qua
 - Tối ưu hóa một hàm tiện ích (utility function) được định nghĩa đối với các trạng thái đã duyệt qua
 - Một nhiệm vụ (task) cần hoàn thành

Lập kế hoạch và thực hiện

■ Bộ lập kế hoạch (Planner)

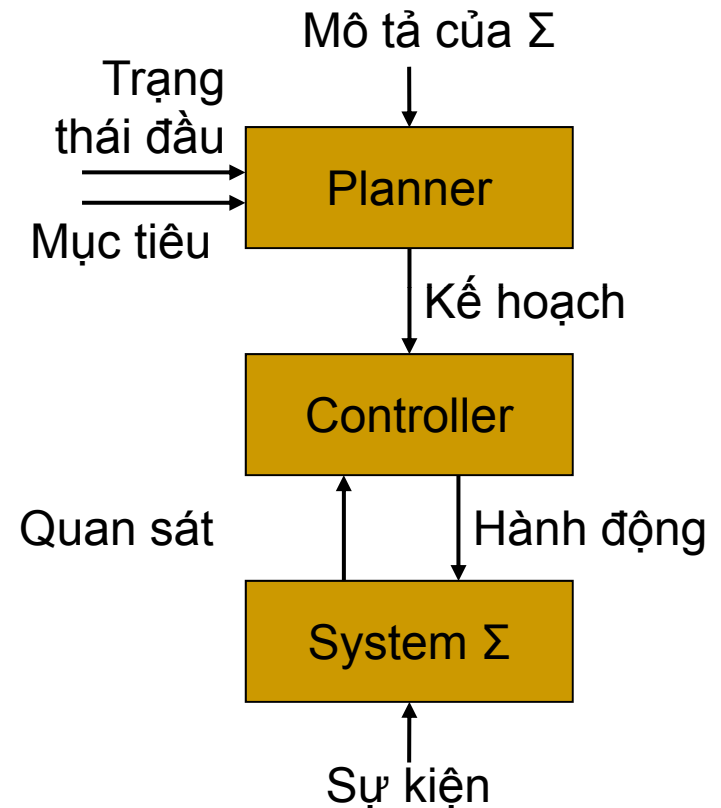
- Đầu vào: Mô tả của hệ chuyển trạng thái Σ , trạng thái đầu, mục tiêu
- Đầu ra: Kế hoạch để đạt được mục tiêu đó

■ Bộ điều khiển (Controller)

- Đầu vào: Kế hoạch cần thực hiện, trạng thái hiện thời (thông qua hàm quan sát $\eta: S \rightarrow O$, với O là tập các quan sát có thể)
- Đầu ra: Hành động (tiếp theo) cần thực hiện

■ Hệ chuyển trạng thái (State-transition system)

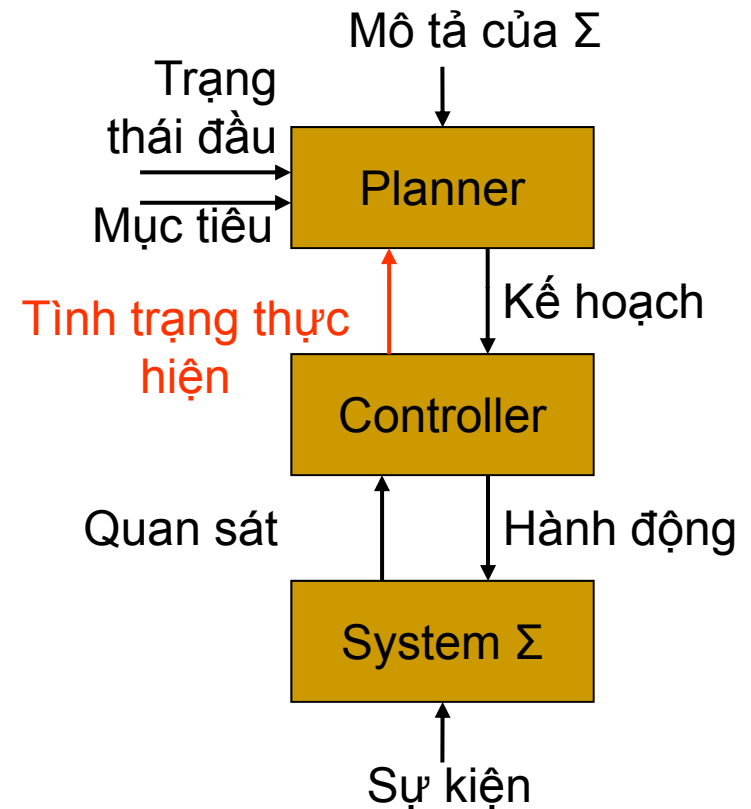
- Tiến triển tùy thuộc vào các hành động được thực hiện và các sự kiện xảy ra



(<http://www.inf.ed.ac.uk/teaching/courses/plan/>)

Lập kế hoạch động (1)

- Vấn đề: Thế giới thực tế có thể khác với mô hình được biểu diễn bởi Σ
 - Sự khác biệt này cần phải được xử lý bởi bộ điều khiển (Controller)
- Mô hình thực tế: **Kết hợp đan xen giữa lập kế hoạch và thực hiện**
 - *Giám sát kế hoạch (Plan supervision)*: để phát hiện những khi kết quả quan sát được khác với mong muốn



(<http://www.inf.ed.ac.uk/teaching/courses/plan/>)

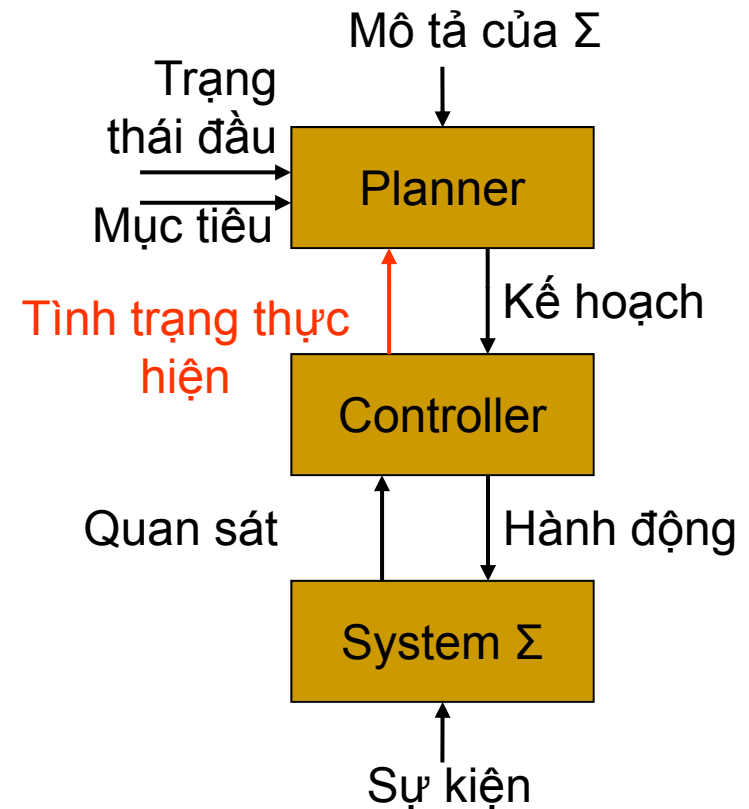
Lập kế hoạch động (2)

...

- ❑ *Sửa lại kế hoạch (Plan revision)*: để thay đổi thích nghi kế hoạch (đã lập) theo những tình huống mới
- ❑ *Lập lại kế hoạch (Re-planning)*: để sinh ra kế hoạch mới từ trạng thái hiện thời (hoặc từ trạng thái đầu)

■ **Lập kế hoạch động (Dynamic planning)**: Quá trình lặp lại giữa bộ lập kế hoạch và bộ điều khiển thực hiện

- ❑ Tùỵ thuộc tình trạng của việc thực hiện kế hoạch



(<http://www.inf.ed.ac.uk/teaching/courses/plan/>)

Giả thiết A0: Tập các trạng thái

- **Giả thiết A0:** Tập các trạng thái (trong biểu diễn của Σ) là hữu hạn
- Nói lỏng điều kiện của giả thiết A0
 - Để biểu diễn các hành động tạo nên (làm xuất hiện) các đối tượng mới của bài toán
 - Để cho phép làm việc với các biến trạng thái kiểu số (numerical-valued state variables) – chứ không chỉ với các biến trạng thái kiểu định danh (nominal-valued state variables)
- Vấn đề gặp phải khi nói lỏng điều kiện của giả thiết A0
 - Khả năng quyết định được (decidability) và khả năng kết thúc (termination) của các bộ lập kế hoạch sẽ không được bảo đảm

Giả thiết A1: Quan sát đầy đủ

- **Giả thiết A1:** Hệ thống có thể quan sát được đầy đủ (fully observable)
 - Planner và Controller có tri thức (quan sát) đầy đủ về trạng thái của thế giới xung quanh
 - Hàm quan sát η cho phép xác định chính xác trạng thái tương ứng với quan sát
- Nói lỏng điều kiện của giả thiết A1
 - Để làm việc với các trạng thái mà trong đó các khía cạnh (của một trạng thái) không được quan sát (nhận biết) đầy đủ
 - Ví dụ: Trong bài toán lập kế hoạch lộ trình có thể xảy ra tắc đường, thì trạng thái của bài toán là không thể nhận biết (quan sát) được đầy đủ
- Các vấn đề gặp phải khi nói lỏng điều kiện của giả thiết A1
 - Với hàm quan sát η : $\eta(s)=o$, thì $\eta^{-1}(o)$ thường sẽ trả về nhiều hơn một trạng thái có thể – vấn đề nhập nhằng
 - Các quan sát không hoàn toàn xác định được trạng thái hiện tại
 - Làm sao để xác định trạng thái tiếp theo là gì?

Giả thiết A2: Chuyển trạng thái

- **Giả thiết A2:** Kết quả của mỗi chuyển trạng thái là một trạng thái xác định (duy nhất)
 - Với trạng thái $s \in S$, hành động $a \in A$, và sự kiện $e \in E$, thì không thể xảy ra $\gamma(s, a, e) = \{s', s''\}$ (với s', s'' là 2 trạng thái khác nhau)
- Nói lỏng điều kiện của giả thiết A2
 - Để lập kế hoạch cho các bài toán mà trong đó việc thực hiện một hành động có thể đưa đến các kết quả khác nhau
- Các vấn đề gặp phải khi nới lỏng điều kiện của giả thiết A2
 - Controller phải quan sát được các kết quả thực tế của một hành động
 - Kế hoạch được xây dựng có thể bao gồm thêm các thành phần để xử lý với các hành động có nhiều kết quả

Giả thiết A3: Các sự kiện

- **Giả thiết A3:** Hệ thống là tĩnh, không có các sự kiện (ngẫu nhiên) xảy ra ($E=\emptyset$)
 - $\Sigma = (S, A, \emptyset, \gamma)$; hay biểu diễn ngắn gọn là: (S, A, γ)
- Nói lỏng điều kiện của giả thiết A3
 - Để biểu diễn (mô hình hóa) các bài toán lập kế hoạch trong đó có xảy ra các sự kiện ngẫu nhiên
- Vấn đề gặp phải khi nói lỏng điều kiện của giả thiết A3
 - Đối với Planner, thì môi trường hoạt động của hệ thống trở nên không xác định

Giả thiết A4: Mục tiêu hạn chế

- **Giả thiết A4:** Planner làm việc với một trạng thái đích, hoặc với một tập các trạng thái đích
- Nói lỏng điều kiện của giả thiết A4
 - Để lập kế hoạch cho các bài toán mà trong đó mục tiêu được biểu diễn bằng:
 - các ràng buộc đối với các trạng thái và kế hoạch, hoặc
 - hàm tiện ích, hoặc
 - nhiệm vụ cụ thể
- Vấn đề gặp phải khi nói lỏng điều kiện của giả thiết A4
 - Làm thế nào để biểu diễn và suy diễn đối với các ràng buộc, các hàm tiện ích, và các nhiệm vụ?

Giả thiết A5: Chuỗi hành động

- **Giả thiết A5:** Một kế hoạch được xây dựng là một chuỗi hữu hạn có thứ tự của các hành động được thực hiện liên tiếp nhau
- Nói lỏng điều kiện của giả thiết A5
 - Để lập kế hoạch cho các bài toán trong đó có xảy ra các sự kiện ngẫu nhiên (Xem giả thiết A3)
 - Để tạo nên các kiểu kế hoạch khác nhau (kế hoạch có điều kiện rẽ nhánh, kế hoạch trong đó có những phần mà các hành động được thực hiện không theo thứ tự,...)
- Các vấn đề gặp phải khi nói lỏng điều kiện của giả thiết A5
 - Không được tạo thêm công việc (yêu cầu) đối với Controller
 - Suy diễn với các cấu trúc dữ liệu phức tạp hơn

Giả thiết A6: Thời gian

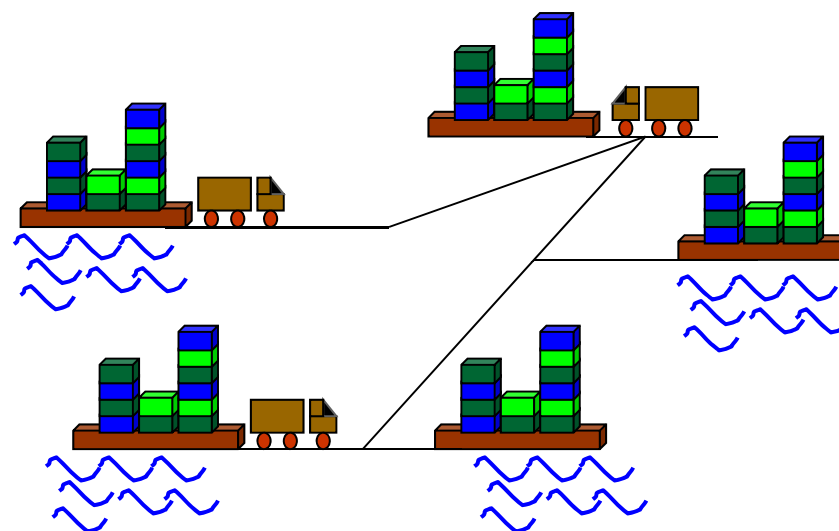
- **Giả thiết A6:** Trong hệ chuyển trạng thái Σ , các hành động và các sự kiện không có khoảng thời gian xảy ra
 - Các chuyển trạng thái xảy ra tức thì – không có biểu diễn rõ ràng về thời gian
- Nói lỏng điều kiện của giả thiết A6
 - Để làm việc với các hành động thực hiện trong một khoảng thời gian, các hành động xảy ra đồng thời, và các thời hạn hoàn thành (deadlines)
- Các vấn đề gặp phải khi nói lỏng điều kiện của giả thiết A6
 - Biểu diễn và suy diễn về thời gian (do các kế hoạch chứa các thông tin về thời gian)
 - Controller phải đợi cho đến khi có được kết quả (tác động) của các hành động (vì mỗi hành động xảy ra trong một khoảng thời gian, chứ không tức thời)

Mô hình giới hạn

- Mô hình giới hạn (restricted model) sẽ đặt ra tất cả các điều kiện trong 7 giả thiết nêu trên (A0-A6)
 - Mô hình giới hạn được sử dụng trong lập kế hoạch cổ điển (classical planning)
- Biểu diễn hình thức của bài toán lập kế hoạch $\mathcal{P}=(\Sigma, s_i, S_g)$
 - $\Sigma = (S, A, \gamma)$: là hệ chuyển trạng thái
 - $s_i \in S$: là trạng thái đầu
 - $S_g \subset S$: là tập các trạng thái đích
- Yêu cầu: Tìm một chuỗi các hành động $\langle a_1, a_2, \dots, a_k \rangle$, tương ứng với một chuỗi các chuyển trạng thái $\langle s_i, s_1, \dots, s_k \rangle$, sao cho: $s_1 = \gamma(s_i, a_1)$, $s_2 = \gamma(s_1, a_2)$, ..., $s_k = \gamma(s_{k-1}, a_k)$, and $s_k \in S_g$

Ví dụ: Rô-bốt bốc dỡ hàng

- Bài toán lập kế hoạch cho rô-bốt bốc dỡ hàng ở một cầu cảng
- Một cầu cảng với một số vị trí (cho việc cập bến, các con tàu cần bốc dỡ hàng, các nhà kho, các nơi đỗ xe tải,...)
- Các thùng hàng (containers) cần được bốc lên/dỡ xuống tàu hàng (hoặc kho hàng)
- Rô-bốt và cần trục (crane) di chuyển các thùng hàng để bốc/dỡ hàng lên/xuống tàu



(M. Ghallab, ESSLLI 2002)

Các đối tượng (1)

- Các vị trí **locations** {loc1, loc2, ...}
 - Nhà kho, các chỗ neo đậu tàu, tàu được bốc dỡ hàng, hay nơi đỗ xe tải
- Các rô-bốt bốc dỡ hàng **robots** {robot1, robot2, ...}
 - Các rô-bốt (xe di chuyển thùng hàng)
 - Các rô-bốt có thể di chuyển giữa các vị trí gần kề
 - Tại mỗi vị trí (location), chỉ có tối đa 1 rô-bốt đang làm việc
- Các cần trục bốc dỡ hàng **cranes** {crane1, crane2, ...}
 - Mỗi cần trục gắn (cố định) với một vị trí nhất định
 - Di chuyển các thùng hàng giữa các rô-bốt và các cọc hàng (piles)
 - thuộc cùng một vị trí (the same location)

Các đối tượng (2)

- Các cọc hàng **piles {pile1, pile2, ...}**
 - Mỗi cọc hàng nằm ở (thuộc về) một vị trí nhất định
 - Tấm nâng hàng (pallet) nằm ở dưới đáy của các cọc hàng – trên một tấm nâng hàng có thể có các thùng hàng
- Các thùng hàng **containers {cont1, cont2, ...}**
 - Được xếp trong một cọc hàng nằm trên một tấm nâng hàng
 - Được đặt trên rô-bốt, hoặc nắm (giữ) bởi cần trục
- Tấm nâng hàng **pallet**
 - Được đặt dưới đáy của một cọc hàng
 - Chỉ cần một ký hiệu duy nhất (pallet), vì mỗi tấm nâng được xác định (gắn với) một cọc hàng

Các quan hệ cố định

- Các quan hệ cố định (topology) là các quan hệ không thay đổi theo thời gian
- Quan hệ **adjacent**(l, l')
 - Vị trí l là gần kề với vị trí l'
- Quan hệ **attached**(p, l)
 - Cọc hàng p thuộc về (gắn với) vị trí l
- Quan hệ **belong**(k, l)
 - Cần trục k làm việc ở (gắn với) vị trí l

Các quan hệ có thể thay đổi (1)

- Quan hệ **occupied(*l*)**
 - Vị trí *l* hiện tại có một rô-bốt đang làm việc
 - Lưu ý: Mỗi vị trí chỉ có tối đa 1 rô-bốt đang làm việc
- Quan hệ **at(*r*, *l*)**
 - Rô-bốt *r* đang làm việc ở vị trí *l*
- Quan hệ **loaded(*r*, *c*)**
 - Rô-bốt *r* hiện đang chở thùng hàng *c*
- Quan hệ **unloaded(*r*)**
 - Rô-bốt hiện đang không chở thùng hàng nào cả
 - Lưu ý: **loaded(*r*, *c*)** có nghĩa là **not unloaded(*r*)**

Các quan hệ có thể thay đổi (2)

- Quan hệ **holding(k, c)**
 - Cần trục k hiện tại đang giữ thùng hàng c
- Quan hệ **empty(k)**
 - Cần trục k hiện tại không giữ bất kỳ thùng hàng nào
- Quan hệ **in(c, p)**
 - Thùng hàng c hiện tại đang ở trong cộc hàng p
- Quan hệ **on(c, c')**
 - Thùng hàng c hiện tại đang nằm ở trên cộc hàng/tấm nâng c'
- Quan hệ **top(c, p)**
 - Thùng hàng/tấm nâng c hiện tại đang nằm trên đỉnh của cộc hàng p

Các hành động

- Hành động **move**(r, l, l')
 - Rô-bốt r di chuyển từ vị trí l đến vị trí l' (không có rô-bốt khác đang làm việc ở l')
- Hành động **take**(c, k, p)
 - Tại cùng vị trí l : cần trục k (hiện không giữ thùng hàng nào) lấy thùng hàng c khỏi cọc hàng p
- Hành động **put**(c, k, p)
 - Tại cùng vị trí l : cần trục k đặt thùng hàng c lên đỉnh của cọc hàng p
- Hành động **load**(r, c, k)
 - Tại cùng vị trí l : cần trục k đặt thùng hàng c vào rô-bốt r (hiện không giữ thùng hàng nào)
- Hành động **unload**(r, c, k)
 - Tại cùng vị trí l : cần trục k (hiện không giữ thùng hàng nào) lấy thùng hàng c khỏi rô-bốt r

Các ứng dụng thực tế

- Các kết quả nghiên cứu của lĩnh vực Lập kế hoạch (planning) đã được áp dụng thành công trong nhiều bài toán thực tế
 - Không gian vũ trụ, Hàng không, Đường sắt
 - Rô-bốt và lập kế hoạch di chuyển
 - CAD/CAM
 - Lập kế hoạch giải quyết các tình huống khẩn cấp
 - Điều khiển các quá trình công nghiệp
 - Các trò chơi (games)
 - ...