Computation, Information, and Intelligence (COMS/ENGRI/INFO/COGST 172), Fall 2005
9/2/05: **Lecture aid – Implicit specifications; search**

Agenda: more examples of the design decisions involved in creating implicit specifications. We may also get to introduce search using path trees.

Announcements: The following drop-in office hours will be held in the near term: Friday 9/2 (today) 11:15-12 in 4152 Upson (Prof. Lee); Monday 9/5 12:45-1:45 in 4152 Upson (Prof. Lee); Monday 9/5 3:30-4:30 in Upson 328C (Anton Morozov); Tuesday 9/6 3-4 in Upson 328C (Marek Janicki). Regular office hours will be announced soon.

Follow-ups to last time:

- For reference, here was the rhetorical structure of the latter part of last lecture. (I) Give an example of a (more or less) full implicit specification (#1 on the lecture aid). (II) Demonstrate how delicate implicit specifications can be by showing the impact that different design choices can have. (IIa) Example: consider a different specification (let's now call it #1a) where we remove time information from the actions, so that they are of the form $\langle course \rangle$.[1] (IIb) Show examples of two types of violations of well-definedness that result: we can't tell what single state is the result of applying ⟨MATH191⟩ to the initial state, and can't tell whether ⟨MATH191⟩ is applicable to state [engri: —; science: —; math: MATH191(9); other: —].

- Note that *if* we had had a definition of problem-space specification that allowed 0 states (*which we don't! You may not change the definitions we have been given to work with, and you may not choose to work with different definitions!*), we would have had to change our definition of the initial-state component, i.e., that exactly one of the states be designated as the initial state, to take this possibility into account. Similar changes would be needed for the definition of the action-set component. This is not to say that such an alternate definition would be wrong or inferior or internally inconsistent, if done correctly, though. I just want to point out again that small changes in one detail can require fixes in other places.

- While our definition requires that applying a given action to a given state clearly results in exactly one state, it's worth pointing out that more complex formalisms — that we will not cover in class — do allow for ambiguity in effect of an action.

- The "$3x + 1$" problem (if you divide by 2 if x is even, multiply by 3 and add 1 otherwise, does every positive integer eventually return to 1?), is still unsolved. The June 2005 version of Lagarias's bibliography (http://arxiv.org/abs/math.NT/0309224) states that all positive integers up to 1.441×10^{18} have been verified to indeed return to 1.

---

[1] For the purposes of lecture brevity, we tacitly assumed that the application and effect descriptions would be changed to something like: "An action $\langle course \rangle$ applies to any state [engri: x_{engri}; science: $x_{science}$; math: x_{math}; other: x_{other}]. such that none of the x_i's lists the time that *course* meets at" and "If the class *course* is of type i, then the result of applying $\langle course \rangle$ to an applicable state is to transition to the state in which the pair $course(time)$ has been added to the appropriate location in the list x_i as specified by the state-set definition, where *time* is the time that *course* meets. (Of course, if x_i is blank, then the new state has the blank replaced by $course(time)$.)" The problem is that talking about "*the*" time that a course meets doesn't make sense, since one course had more than one meeting time. This is where the problems with well-definedness stem from. As was asked by someone last time, if it had been the case that there was no ambiguity with respect to course meeting times, then specification #1a would not have this particular problem with well-definedness, although in that case it would be prudent to explicitly explain that the non-ambiguity allows the omission.

While we're on the subject, another way one fix #1a to be well-defined would be to change the effect description to resolve the time ambiguity, for example, by choosing "the" time to be the *earliest* possible meeting time. This specification might run into problems with faithfulness, though (do you see why?).

The course-requirements problem (summary) Must have an ENGRI, science, and math course; no time conflicts allowed; start with no requirements fulfilled.

Time	Courses available
9 MTWRF	ENGRI 111, MATH 171, MATH 191
10 MTWRF	CHEM 207, ENGRI 172
11 MTWRF	CHEM 211, MATH 191, MATH 192
12 MTWRF	ART 151, FWS 270, PHYS 116

Sketch of state-set and action-set components for specification #1.

States: all possible "advising worksheets" of the form

$$[\text{engri: } x_{engri}\text{; science: } x_{science}\text{; math: } x_{math}\text{; other: } x_{other}]$$

where

- each x_i is either a blank ("—") or a list of items of the form $course(time)$ such that $course$ is a class of type i that meets at time $time$;
- (no-conflict constraint) No time appears more than once among all the x_is; and
- (ordering constraint) if x_i lists multiple courses, they are listed alphabetically and then by ascending numerical order and then by ascending course-meeting time.

Actions: all pairs of the form $\langle course, time\rangle$ where $course$ is a class meeting at time $time$. Constraints given last time implied that essentially each action has the effect of adding another course/time item to the checklist in the appropriate place in the progress-towards-requirement slot, assuming no time conflict would arise.

Question 1a: (answered last time) Can we omit time information from the actions?

Question 1b: Can we omit the "other" information from the states (no change to action-set definition)? (Illuminating example: Assume a reasonable implementation of this change as an alternate specification, call it #1b. If we apply $\langle \text{ART151}, 12\rangle$ to the initial state, can we apply $\langle \text{PHYS116}, 12\rangle$?)

Important moral: a state carries no information beyond its definition, and hence cannot tell what previous states or actions brought the world to the situation the state represents.

Question 1c: Can we omit the ordering constraint? (Illuminating example: Assume a reasonable implementation of this change as an alternate specification, call it #1c. What if we apply $\langle \text{MATH191}, 9\rangle$ to the state [engri: —; science:—; math: MATH171(9); other:—]?

Implicit specification #2. Italics denote variables. This specification exhibits the minimum level of explanations and descriptions of motivation that we require of you. There are some modifications, essentially cosmetic, from the version given in last lecture's handout.

A. The set of states consists of checklists of the form

$$[\text{engri: } x_{engri}\text{; science: } x_{science}\text{; math: } x_{math}\text{; 9: } t_9\text{; 10: } t_{10}\text{; 11: } t_{11}\text{; 12: } t_{12}]$$

where each x_i and t_i is either "—" or "✓". The intent is that $x_i =$ ✓ if and only if a course of type i has been scheduled, and that $t_j =$ ✓ if and only if a course has been scheduled for time j.

B. The initial state is [engri: —; science: —; math: —; 9: —; 10: —; 11: —; 12: —].

C. The set of goal states is the set of states of the form [engri: ✓; science: ✓; math: ✓; 9: t_9; 10: t_{10}; 11: t_{11}; 12: t_{12}] where each t_i may be either ✓ or —.

D. The set of actions corresponds to all pairs of the form $\langle course, time\rangle$ where *course* is a class that meets at time *time*.

An action $\langle course, time\rangle$ applies to any state [engri: x_{engri}; science: $x_{science}$; math: x_{math}; 9: t_9; 10: t_{10}; 11: t_{11}; 12: t_{12}] such that $t_{time} =$ —; that is, we disallow time conflicts, as required. The result of applying $\langle course, time\rangle$ to such a state is to transition to the state in which all the variables have the same value as in the application state except that t_{time} has been changed from — to ✓, and, if *course* is a class of type i and x_i is blank in the application state, then $x_i =$ ✓ in the new state. The idea is that every course scheduled counts toward its type and its time period.

Implicit specification #3. (Motivations amd explanations omitted for lecture conciseness)

A. The states consist of checklists of the form

$$[\text{engri: } x_{engri}\text{; science: } x_{science}\text{; math: } x_{math}\text{; other: } x_{other}]$$

where each x_i is either "—" or "✓".

B. The initial state is [engri: —; science: —; math: —; other: —.].

C. Goal states: those of the form [engri:✓; science:✓; math:✓; other:x], where x is ✓ or —.

D. The set of actions are all those of the following types

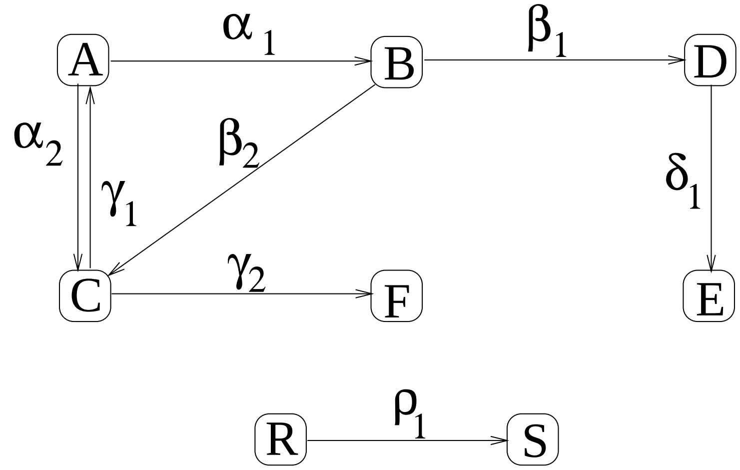
$\langle 9\text{: } y_9\text{; 10: } y_{10}\text{; 11: } y_{11}\rangle$	$\langle 10\text{: } y_{10}\text{; 11: } y_{11}\text{; 12: } y_{12}\rangle$
$\langle 9\text{: } y_9\text{; 10: } y_{10}\text{; 12: } y_{12}\rangle$	$\langle 10\text{: } y_{10}\text{; 11: } y_{11}\text{; 12: } y_{12}\rangle$
$\langle 9\text{: } y_9\text{; 11: } y_{11}\text{; 12: } y_{12}\rangle$	$\langle 9\text{: } y_9\text{; 10: } y_{10}\text{; 11: } y_{11}\text{; 12: } y_{12}\rangle$

where each y_j is a course meeting at time j.

All the operators apply only to the initial state. Applying any action $\langle j\text{: } y_j\text{; } k\text{: } y_k\text{; } \ell\text{: } y_\ell\rangle$ to this state results in the state [engri: x_{engri}; science: $x_{science}$; math: x_{math}; other: x_{other}] where each x_i has the value "✓" if and only if at least one of y_j, y_k, or y_ℓ is a class of type i. Applying action $\langle 9\text{: } y_9\text{; 10: } y_{10}\text{; 11: } y_{11}\text{; 12: } y_{12}\rangle$ to the initial state results in the state [engri: x_{engri}; science: $x_{science}$; math: x_{math}; other: x_{other}] where each x_i has the value "✓" if and only if at least one of y_9, y_{10}, y_{11}, or y_{12} is a class of type i.

From problem space specifications to path trees Below is a problem space specificationand a corresponding *path tree*, which represents by downward-pointing "tendrils" every possible sequence of actions that can be taken starting from the initial state. Next time we'll be talking about how path trees are constructed from problem spaces, so you may find it useful to see if you can figure out ahead of time how this construction probably proceeded.

Example problem space specification: The states and actions are depicted below; the operators have been given Greek-letter names. A is the initial state, and for now, let there be no goal state.



A corresponding path tree (goal states not indicated, operator labels omitted, assume numerical order on actions):