Logic Programming

Dynamic Logic Programs

Michael Genesereth
Computer Science Department
Stanford University
Example - Blocks World
Symbols: a, b, c, d, e

Unary Predicates:
  clear - blocks with no blocks on top
  table - blocks on the table

Binary Predicates:
  on - pairs of blocks in which first is on the second
  above - pairs in which first block is above the second

  u(x, y) - means that x is moved from y to the table.
  s(x, y) - means that x is moved from the table to y.
Operations:
\( u(x, y) \) means that \( x \) is moved from \( y \) to the table.
\( s(x, y) \) means that \( x \) is moved from the table to \( y \).

Transition Rules:
\[
\begin{align*}
  u(X, Y) & \land clear(X) & \land on(X, Y) \\
                      & \implies \neg on(X, Y) & \land table(X) & \land clear(Y)
\end{align*}
\]
\[
\begin{align*}
  s(X, Y) & \land table(X) & \land clear(X) & \land clear(Y) \\
                      & \implies \neg table(X) & \land \neg clear(Y) & \land on(X, Y)
\end{align*}
\]
Dataset:

- clear(c)
- on(c,a)
- table(b)
- clear(d)
- on(a,b)
- table(e)
- on(d,e)

Transition Rule:

\[ u(X,Y) \land \text{clear}(X) \land \text{on}(X,Y) \implies \neg\text{on}(X,Y) \land \text{table}(X) \land \text{clear}(Y) \]

Result of \( u(c,a) \):

- clear(a)
- on(a,b)
- table(b)
- clear(c)
- on(d,e)
- table(c)
- clear(d)
- table(e)
Example - Tic Tac Toe
Tic Tac Toe

cell(1,1,x)  
cell(1,2,b)  
cell(1,3,b)  
cell(2,1,b)  
cell(2,2,o)  
cell(2,3,b)  
cell(3,1,b)  
cell(3,2,b)  
cell(3,3,x)  
control(o)
mark(J,K) & cell(M,N,R) ==> ~cell(M,N,R)

mark(J,K) & control(R)  
  ==> cell(J,K,R)

mark(J,K) & cell(M,N,R) & distinct(R,b)  
  ==> cell(M,N,R)

mark(J,K) & cell(M,N,R) & distinct(J,M)  
  ==> cell(M,N,b)

mark(J,K) & cell(M,N,R) & distinct(K,N)  
  ==> cell(M,N,b)
The Game of Life
World
(1) Any live cell with two or three live neighbors lives on to the next generation.

(2) Any live cell with fewer than two live neighbors dies (as if caused by underpopulation).

(3) Any live cell with more than three live neighbors dies (as if by overpopulation).

(4) Any dead cell with exactly three live neighbors becomes a live cell (as if by reproduction).
Symbols: \( c_{11}, c_{12}, \ldots \)

Unary Predicates:
- \( \text{on} \) - cell is live
- \( \text{cell} \) - true of cells

Binary Predicates:
- \( \text{neighbor} \) - cells are neighbors
- \( \text{supports} \) - first cell is alive and a neighbor of the second

Definition of support

\[
\text{supports}(X,Y) := \text{neighbor}(X,Y) \land \text{on}(X)
\]
Any live cell with fewer than two live neighbors dies.

\[
on(Y) \land \text{countofall}(X, \text{supports}(X, Y), N) \land \text{leq}(N, 1) \implies \neg \text{on}(Y)
\]

Any live cell with more than three live neighbors dies.

\[
on(Y) \land \text{countofall}(X, \text{supports}(X, Y), N) \land \text{geq}(N, 4) \implies \neg \text{on}(Y)
\]

Any dead cell with exactly three live neighbors becomes a live cell.

\[
\text{cell}(Y) \land \neg \text{on}(Y) \land \text{countofall}(X, \text{supports}(X, Y), 3) \implies \text{on}(Y)
\]
http://logicprogramming.stanford.edu/examples/gameoflife.html
http://logicprogramming.stanford.edu/examples/gameoflife.html