How Minds Work
Behavior Nets

Stan Franklin
Computer Science Division &
Institute for Intelligent Systems
The University of Memphis
Desired Characteristics

- Goal oriented
- Opportunistic
- Persistent
- Able to plan ahead
- Robust
- Reactive
- Fast

- Note lack of independence
- Tradeoffs needed
- Optimality not asked for
- Good enough will do
Competence Modules

- Simple
- Interactive
- Mindless
- Each with a specific competence
- Like processors, codelets, demons, schema
A Competence Module

• Much like a production rule, demon, schema
• Preconditions — environmental facts required for the competence to be performed
• An action
• Additions and deletions — facts to be added or deleted after the action is taken
• Activation — a number, some kind of strength level
The Behavior Network

• Competence module = node of a digraph
• links are completely determined by the competence modules
  – Successor links
  – Predecessor links
  – Conflictor links
Successor links

Preconditions

\[
\begin{array}{c}
\text{l} \\
\text{k} \\
\text{i}
\end{array}
\]

Competence Module

\[
\begin{array}{c}
x
\end{array}
\]

Activation

add list

\[
\begin{array}{c}
a \\
b \\
c
\end{array}
\]

delete list

\[
\begin{array}{c}
l \\
m \\
n
\end{array}
\]

Preconditions

\[
\begin{array}{c}
b \\
c \\
d
\end{array}
\]

Competence Module

\[
\begin{array}{c}
y
\end{array}
\]

Activation

add list

\[
\begin{array}{c}
a
\end{array}
\]

delete list

\[
\begin{array}{c}
l \\
m \\
n
\end{array}
\]

successor links

HMW: Behavior Nets
Predecessor links

successor links

predecessor links

HMW: Behavior Nets
Conflictor links

successor links
predecessor links
conflictor links

HMW: Behavior Nets
Activation Comes From

- Activation stored by the competence modules
- from the environment
- from built-in, global goals
Environmental Activation

• Environment activation for each true precondition
• The more true preconditions, the more relevant the competence is
• This source of activation allows the system to be opportunistic
Goal Activation

• If a competence satisfies a goal, the goal will send activation
• This source of activation tends to make the system goal directed
• A completed goal inhibits any competence that will undo it
Activation Along Successor Links

- Activation spread from competence to competence along links.
- Along successor links, one competence strengthens those competencies whose preconditions it can help fulfill.
Activation Along Predecessor & Conflictor Links

- Along predecessor links, a competence strengthens any other that fulfills one of its own preconditions.
- Along a conflictor link a competence inhibits any other that can undo one of its true preconditions.
- Every conflictor link is inhibitory.
Executable Competence

- Call a competence module *executable* if all of its preconditions are satisfied
- The competence is ready to fire, although it may well not
Behavior Net Loop

- Add activation from environment and goals
- Spread activation forward and backward among the competence modules
- Decay - total activation remains constant
- Competence fires if
  - it's executable and
  - it's over threshold and
  - it's the maximum such

- If one competence fires, its activation is zero, and threshold returns to normal
- If none fires, reduce threshold by 10%
Global Parameters

• Activation threshold
  – Raising it makes the system more thoughtful,
  – Lowering makes it more reactive
• Activation added for each satisfied precondition
  – Increasing makes the system more opportunistic
• Activation for being able to satisfy a goal
  – Increasing leads to more goal-oriented behavior
• Last two parameters tradeoff, since goal orientation and opportunism are opposite
• Parameters remain constant during a run
Plans

• Sequence of competencies transform present situation into desired one
• Sequence can become highly activated by forward spreading from current state & backward spreading from a goal state
• May occur in competition with other sequences striving towards other goals
Really a Plan?

• An outside observer might call it a plan
• System doesn't use it as a plan
• Plan seems to exist only in the likelihood for execution of its competencies
• No centralized preprogrammed search processing results in cheaper operation than traditional planning methods
Problems with Behavior Nets

- No variables — preconditions & add & delete list composed of propositions
- Causes computational explosion
- No learning — must be hand crafted
- Expensive to produce
Readings

Email and Web Addresses

• Stan Franklin
  – franklin@memphis.edu
  – www.cs.memphis.edu/~franklin

• “Conscious” Software Research Group
  – www.csrg.memphis.edu/