Model-based Autonomy in the New Millenium

Brian C. Williams

NASA Ames Research Center

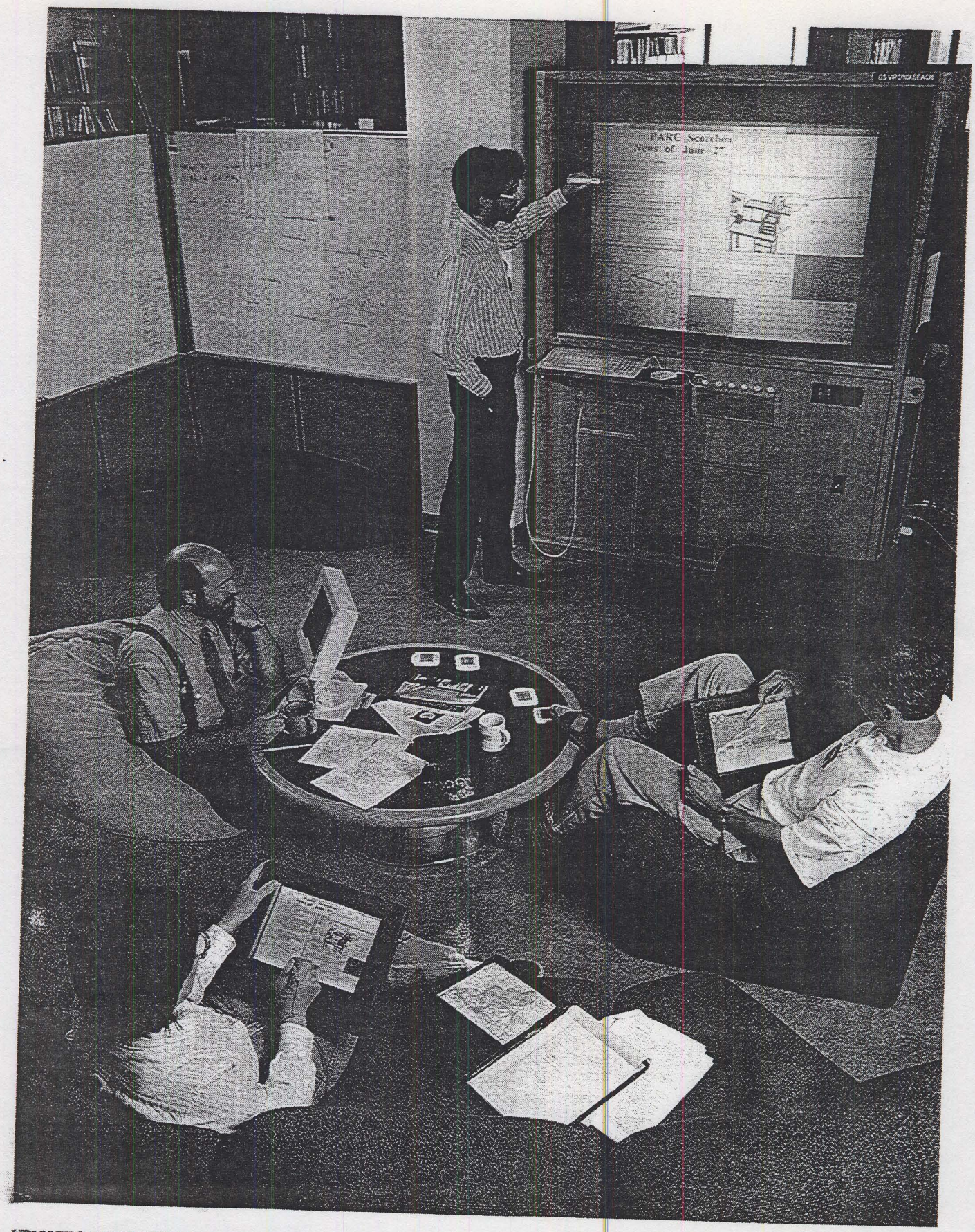
joint with P. Pandurang Nayak

AI Assertions

- Monmonotonic reasoning is essential to acting in the world.
- Deduction should be eliminated from the reactive loop.
- Qualitative modeling is too ambiguous.
- An LTMS is slow, an ATMS is fast.

Internet Agents

- Testbeds easily available
- Can explore fundamental issues
  - mobility
  - realtime interaction
  - information gathering
- Its hip
- financially lucrative



UBIQUITOUS COMPUTING begins to emerge in the form of live boards that replace chalkboards as well as in other devices at the Xerox Palo Alto Research Center. Computer scientists gather around a live board for discussion. Building boards and integrating them with other tools has helped researchers understand better the eventual shape of ubiquitous computing. In conjunction with active badges, live boards can customize the information they display.

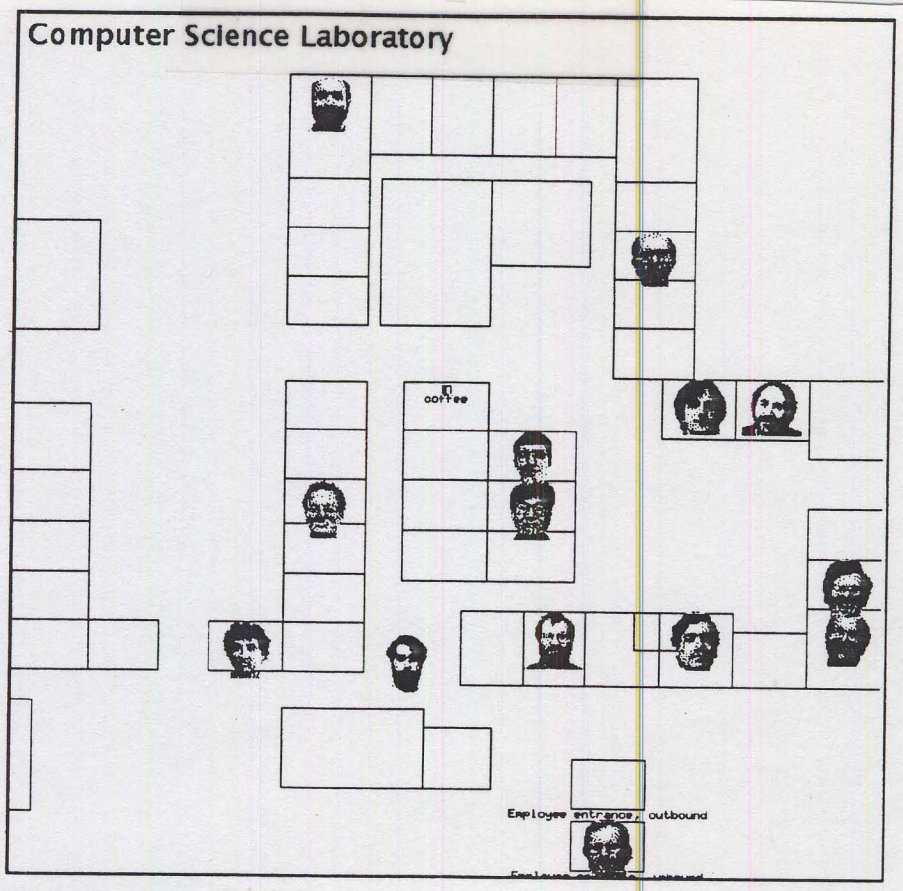
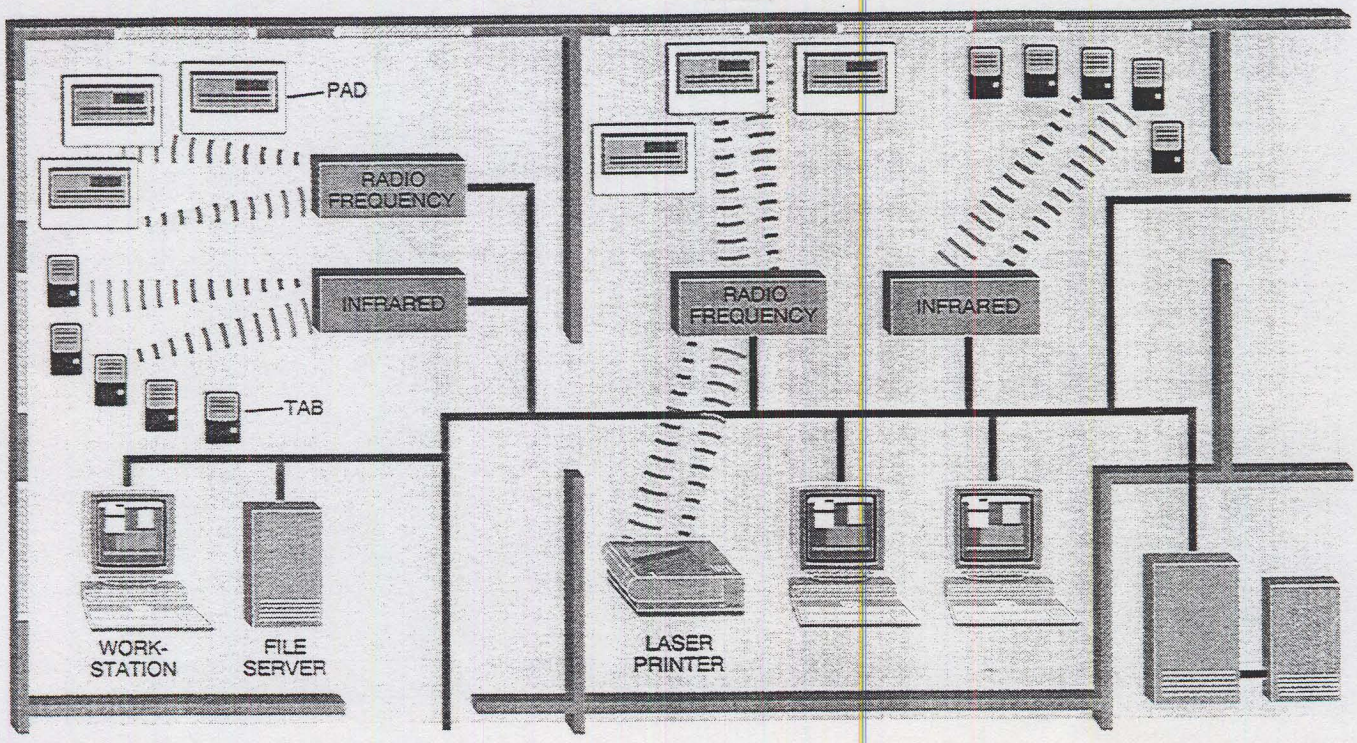


Figure 3. Display of CSL activity from personal locators.



WIRED AND WIRELESS NETWORKS link computers and allow their users to share programs and data. The computers pictured here include conventional terminals and file serv-

known as pads. Future networks must be capable of supporting hundreds of devices in a single room and must also cope with devices—ranging from tabs to laser printers or large-

Outline

- Immobile Robots
- Model-based Autonomous Systems
- The Cassini Challenge
- Model-based Programming
- Model-based Execution
- A search engine for reactive control
- Hybrid Modeling
- Formalization and Reduction
- Deep Space One

Immobile Robots

1. Physically Embedded

2. Immobile

3. Self-Absorbed

   - Tight couplings
   - high reconfigurability

4. (Massively) Distributed

5. Heterogenous

6. Hybrid Discrete - Continuous

Model-based Autonomous Systems

1. Model-based Programming
   - compositional modeling
   - qualitative modeling
2. Model-based Execution
   - Self Configuration
   - Self Modeling
   - Deduction in the reactive loop
3. Model-based hybrid systems
   - concurrent systems $\Rightarrow$ **Livingstone** (AAAI96)
   - adaptive systems $\Rightarrow$ Moriarty (QR96), AA (AAAI94

Model-based Programming

Control Code Tasks:

- monitoring
- **tracking planner goal activations**
- **confirming hardware modes**
- **reconfiguring hardware**
- **detecting anomalies**
- **isolating faults**
- **diagnosis**
- **fault recovery**
- **standby**
- **safing**
- **fault avoidance**
- parameter estimation
- adaptive control
- control policy coordination

$\Rightarrow$ perform using a single model

Model-based Autonomy

Requirements:

1. supports above functions
2. correct response to novel situations
3. 100msec reaction time
4. 4 month development time
5. 80 components, 280 modes
6. Hybrid models

A Kernel for Model-based Execution

Requirements (3-5): 100 msecs, 80 components, 280 modes, 4 months

Heritage: Conflict-based Diagnosis (Sherlock)

- Diagnoses thousand component combinatorial circuits in 1-2 minutes
- Probabilistic, best first search
- Conflicts eliminate infeasible subspaces
- Prediction uses local propagation

$$\Rightarrow \text{arg min } f(X) \text{ st } M(X)$$

A Kernel for Model-based Execution

To achieve reactivity: (100 msecs, 10,000 clauses)

- Precompile model
- Reduce model to propositional formula
- Exploit unit propagation
- ATMS $\rightarrow$ LTMS
- Best first enumeration exploits monotonic decrease of probability wrt superset

Hybrid Modeling (6)

Heritage:

- Concurrent, reactive system specification (Manna & Pnueli 91)

- qualitative algebra (Williams 88,Struss88)

Components: concurrent transition systems

Software I/O: constraints over finite domain

Hardware I/O: algebra on sign and relative values

$\Rightarrow$ Reduce incrementally to propositional logic

Failure Transition System: $\langle \Pi, \Sigma, \mathcal{T} \rangle$

- state variables Π, feasible states Σ, transitions $\mathcal{T}$.
- $\tau \in \mathcal{T} : \Sigma \rightarrow 2^{\Sigma}$
- Σ is finite
- $\tau_n \in \mathcal{T}$ denotes nominal transition all others denote failure.

Trajectory: $\sigma : s_0, s_1, \ldots$ for feasible $s_i \in \Sigma$

- nondeterministically selects τ_n or failure τ.
- $s_{i+1} \in \tau(s_i)$ for some $\tau \in \mathcal{T}$

$\langle \Pi, \Sigma, \mathcal{T} \rangle$ Specified Propositionally

- propositional state formulae
  - propositions are $y_k = e_k$, such that $y_k \in \Pi$ and $e_k \in \text{domain}(y_k)$
  - Given s_i, $y_k = e_k$ is true iff the value of y_k is e_k in s_i
- next operator $\bigcirc$
  - Given s_i, $\bigcirc\Phi$ is true if Φ is true in s_{i+1}

Π domain specified by:

$$\bigvee_i y_k = e_{ki},\ \bigwedge_{i \neq j} \neg(y_k = e_{ki} \wedge y_k = e_{kj})$$

Σ specified by state formula $\rho_{\mathcal{S}}$

$\tau \in \mathcal{T}$ specified by $\rho_\tau \equiv \bigwedge_i \rho_{\tau_i}$ where

- $\rho_{\tau_i} \equiv \Phi_i \Rightarrow \bigcirc\Psi_i$ for state formulae Φ_i, Ψ_i.

Driver Example Specification

$\Pi = \{mode, cmdin, cmdout\}$, where

1. $mode \in \{on, \mathsf{off}, resettable, failed\}$
2. $cmdin \in \{on, \mathsf{off}, reset, open, close, none\}$
3. $cmdout \in \{open, close, none\}$

$\rho_{\mathcal{S}}$:

$$
\begin{array}{lcl}
mode = on & \Rightarrow & (cmdin = open \Rightarrow cmdout = open) \\
 & & \wedge (cmdin = close \Rightarrow cmdout = close) \\
 & & \wedge \neg(cmdin = open \vee cmdin = close) \Rightarrow cmdout = none \\
mode = \mathsf{off} & \Rightarrow & cmdout = none
\end{array}
$$

ρ_{τ_n}:

$$
\begin{array}{rcl}
((mode = on) \vee (mode = \mathsf{off})) \wedge cmdin = \mathsf{off} & \Rightarrow & \bigcirc mode = \mathsf{off} \\
((mode = on) \vee (mode = \mathsf{off})) \wedge cmdin = on & \Rightarrow & \bigcirc mode = on \\
\neg(mode = failed) \wedge cmdin = reset & \Rightarrow & \bigcirc mode = on \\
mode = reset \wedge \neg(cmdin = reset) & \Rightarrow & \bigcirc mode = reset \\
mode = failed & \Rightarrow & \bigcirc mode = failed
\end{array}
$$

ρ_{τ_f}: $\bigcirc mode = failed.$

ρ_{τ_r}: $\bigcirc mode = reset.$

Concurrent Transition Systems $\mathcal{S} = \langle \Pi, \Sigma,$

Requirement: Model used reactively $\Rightarrow$ Synchronous

- composed of transition systems $\mathcal{CD}$
- $\mathcal{CD}$ are concurrent and *synchronous*
  - Each $\tau \in \mathcal{T}$ performs one τ_C for each $C \in \mathcal{CD}$:
  - $\rho_\tau \Leftrightarrow \bigwedge_{C \in \mathcal{CD}} \rho_{\tau_C} \Leftrightarrow \bigwedge_j \left(\Phi_{ij} \Rightarrow \bigcirc \Psi_{ij} \right)$

Generates trajectory $\sigma : s_0, s_1 \ldots$ defined by:

$$\rho_{st} \equiv \rho_\Theta \wedge \Box \left(\rho_\Sigma \wedge \bigvee_i \left(\bigwedge_j \left(\Phi_{ij} \Rightarrow \bigcirc \Psi_{ij} \right) \right) \right) \bigwedge_i \bigcirc_i \left(\rho_{\mathsf{obs}_i} \wedge \rho_{\mu_i} \right)$$

Hybrid Transition System

1. Signs: $\langle S', \oplus, \otimes \rangle$, $S' = \{+, 0, -, ?\}$
   (Minima AAAI88, AIJ91)

2. Relative values: $\langle R', \oplus, \otimes \rangle$, $R' \equiv \{L, N, H, ?\}$
   where $[x]_r = H, N, L$ iff $[x - x_n]_s = +, 0, -$

Example: Latched thruster

$\Pi = \{mode, cmdin, inflow, thrust\}$, where

$$\begin{aligned} mode = open \quad &\Rightarrow \quad [inflow]_{sr} = [thrust]_{sr} \\ (mode = \text{stuck-closed}) & \\ \vee (mode = closed) \quad &\Rightarrow \quad [inflow]_s = [thrust]_s = 0 \end{aligned}$$

ρ_{τ_n}:

$$\begin{aligned} ((mode = open) \vee (mode = closed)) & \\ \wedge cmdin = close \quad &\Rightarrow \quad \bigcirc \text{mode} = closed \\ ((mode = open) \vee (mode = closed)) & \\ \wedge cmdin = open \quad &\Rightarrow \quad \bigcirc \text{mode} = \text{open} \\ mode = \text{stuck-closed} \quad &\Rightarrow \quad \bigcirc \text{mode} = \text{stuck-closed} \end{aligned}$$

ρ_{τ_f}: $\bigcirc$mode = stuck-closed

Configuration System $\langle \mathcal{S}, \Theta, \sigma \rangle$

- transition system $\mathcal{S}$
- initial state $\Theta \in \Sigma$ of $\mathcal{S}$
- goal configurations $\sigma : g_0, g_1, \ldots$
  g_i specified by state formulae.

Generates *configuration* trajectory $\sigma : s_0, s_1 \ldots$

- s_0 is Θ
- s_{i+1} satisfies g_i OR
- $s_{i+1} \in \tau(s_i)$ for $\tau \neq \tau_n$

Mode Identification

Functions: confirm hardware modes, track planner goal activations, detect anomalies, isolate faults and perform diagnosis.

Given time i:

- S_i denotes possible states prior to control
- S_{μ_i} denotes states with control μ_i
- $S_i \cap S_{\mu_i}$ denotes possible states w control
- $S_{\mathcal{O}_{i+1}}$ denotes states with observations $\mathcal{O}_{i+1}$

Then:

$$
\begin{aligned}
S_0 &= \{\Theta\} \\
S_{i+1} &= \left(\bigcup_j \tau_j (S_i \cap S_{\mu_i}) \right) \cap \Sigma \cap S_{\mathcal{O}_{i+1}} \\
&= \bigcup_{s \in S_i \cap S_{\mu_i}} \left(\bigcap_k \tau_{jk}(s) \right) \cap \Sigma \cap S_{\mathcal{O}_{i+1}}
\end{aligned}
$$

Weakening:

$$S_{i+1} = \bigcup_{s \in S_i \cap S_{\mu_i}} \left(\bigcap_k \tau_{jk}(s) \right) \cap \Sigma \cap S_{\mathcal{O}_{i+1}}$$

$$\subseteq \bigcup_j \left(\bigcap_k \tau_{jk}(S_i \cap S_{\mu_i}) \right) \cap \Sigma \cap S_{\mathcal{O}_{i+1}}$$

Recall:

$$\rho_{\tau_i} \equiv \Phi_i \Rightarrow \bigcirc \Psi_i$$

In terms of state formulae:

$$\rho_{S_{i+1}} \equiv \bigvee_{\tau_j} \left(\bigwedge_{\rho_{S_i} \wedge \rho_{S_{\mu_i}} \models \Phi_{jk}} \Psi_{jk} \right) \wedge \rho_\Sigma \wedge \rho_{\mathcal{O}_{i+1}}$$

Mode Reconfiguration

Functions: reconfiguring hardware, standby, safing, fault avoidance

Given: goal g_i, possible states S_i and nominal model τ_n

Generate: control values μ_i

Let $\mathcal{M}_i$ denote possible control actions at i:

$$\begin{aligned} \mathcal{M}_i &= \{\mu_j | \tau_n(S_i \cap S_{\mu_j}) \cap \Sigma \subseteq g_i\} \\ &\supseteq \{\mu_j | \bigcap_k \tau_{nk}(S_i \cap S_{\mu_j}) \cap \Sigma \subseteq g_i\} \end{aligned}$$

In terms of state formulae:

$$\mathcal{M}_i \supseteq \{\mu_j | \quad \rho_{S_i} \wedge \rho_{\mu_j} \text{is consistent and} \bigwedge_{\rho_{S_i} \wedge \rho_{\mu_j} \models \Phi_{nk}} \Psi_{nk} \wedge \rho_\Sigma \models \rho_{g_i}\}$$

Livingstone

Task: Generate likely trajectories and optimal control actions

Solve: min $f(X)$ st $C(X)$ using conflict-directed BFS

MI:

- X: For $S \in \mathcal{CD}$ introduce $x \in X$ with domain $\mathcal{T}_S$
- $C(X)$: a state transitioned to using X is consistent with OBS

$$\rho_{S_{i+1}} \equiv \bigwedge_{\rho_{S_i} \wedge \rho_{S_{\mu_i}} \models \Phi_{jk}} \Psi_{jk} \wedge \rho_\Sigma \wedge \rho_{\mathcal{O}_{i+1}}$$

- $f(X)$: $p(\tau|\mathcal{O}_i) = \frac{p(\mathcal{O}_i|\tau)p(\tau)}{p(\mathcal{O}_i)} \propto p(\mathcal{O}_i|\tau)p(\tau)$

MR:

- X : control variables μ
- $C(X)$: μ_j satisfies:

$$\mathcal{M}_i \supseteq \{\mu_j| \quad \rho_{S_i} \wedge \rho_{\mu_j} \text{is consistent and} \bigwedge_{\rho_{S_i} \wedge \rho_{\mu_j} \models \Phi_{nk}} \Psi_{nk} \wedge \rho_\Sigma \models \rho_{g_i}\}$$

$$f(X) - p(\tau|\mathcal{O}_i) = \frac{p(\mathcal{O}_i|\tau)p(\tau)}{p(\mathcal{O}_i)} \propto p(\mathcal{O}_i|\tau)p(\tau)$$

Results

Newmaap Model Characteristics:

Number of components	80
Average modes/component	3.5
Number of propositions	3424
Number of clauses	11101

Recovery Scenario Performance:

Scenario	MI			MR	
	Checked	Accepted	Time	Checked	Time
EGA preaim failure	7	2	2.2	4	1.7
BPLVD failed	5	2	2.7	8	2.9
IRU failed	4	2	1.5	4	1.6
EGA burn failure	7	2	2.2	11	3.6
Acc failed	4	2	2.5	5	1.9
ME too hot	6	2	2.4	13	3.8
Acc low	16	3	5.5	20	6.1

Standard Combinatorial Suite:

Devices	# of components	# of clauses	Checked	Time
c17	6	18	18	0.1
c432	160	514	58	4.7
c499	202	714	43	4.5
c880	383	1112	36	4.0
c1355	546	1610	52	12.3
c1908	880	2378	64	22.8
c2670	1193	3269	93	28.8
c3540	1669	4608	140	113.3
c5315	2307	6693	84	61.2
c7552	3512	9656	71	61.5

Applications: Cassini, ASSAP, Bioreactor ...