




## Autonomous Robust Execution of Complex Robotic Missions




Paul Robertson, Robert T. Effinger, Brian C. Williams  
MERS, CSAIL, MIT (papers on <http://mers.mit.edu>)







## Goals




1. Enable development of complex missions with contingencies (**Complexity**)
2. Provide robustness to component, temporal, and contingency failure (**Robustness**)
3. Support online optimal temporal planning with contingencies (**Optimality**)




## What I won't talk about



1. Health Management/Maintenance
2. Mapping/localization
3. Path Planning
4. Distribution over multiple robots
5. Algorithms for Performance
6. Generative Activity Planner

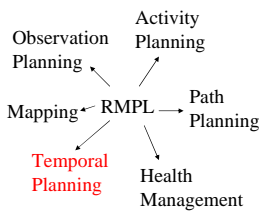


## Model-based Programming



- **Suspicious**
  - Monitors intentions and plans
- **Self-Adaptive**
  - Exploits and generates contingencies
- **State and Fault Aware**
  - Specified at level of states.
  - Achieved using failure knowledge.
- **Anticipatory**
  - Plans and verifies into the future
  - Predicts likely future states
  - Plans contingencies

⇒ “RMPL” reactive Model-Based Programming Language




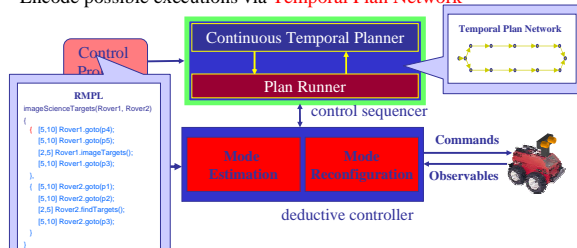
## Model-Predictive Method Selection

To ensure **safe, optimal** execution, the control sequencer:

- Dynamically selects **consistent** methods over **future horizon**,
- Adapts to uncertainty by **selecting execution times dynamically**,
- **Monitors outcomes** and plans contingencies.


Encode possible executions via **Temporal Plan Network**

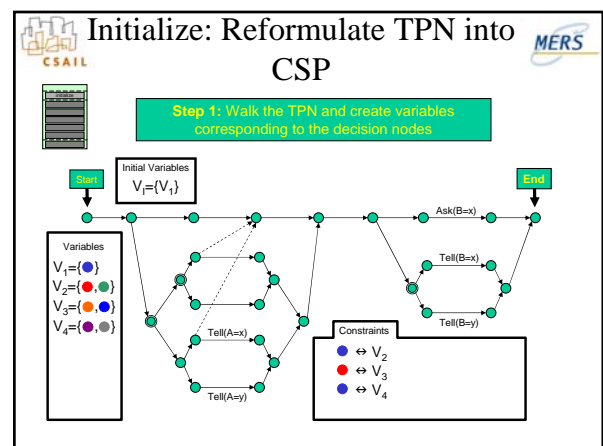
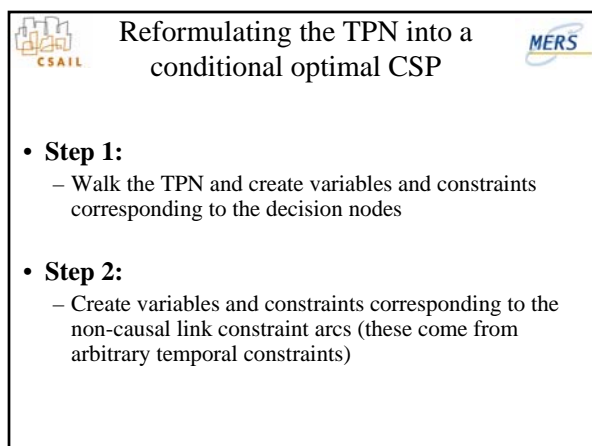
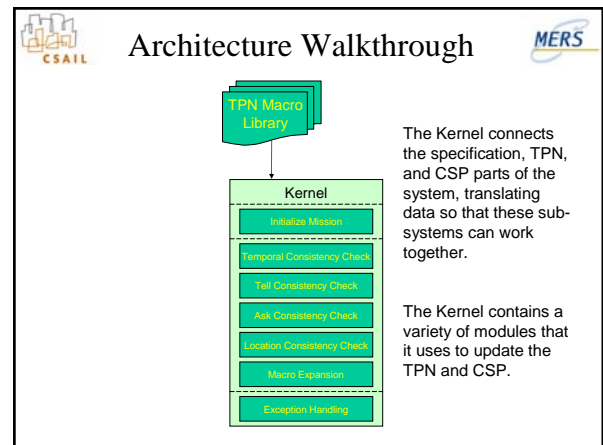
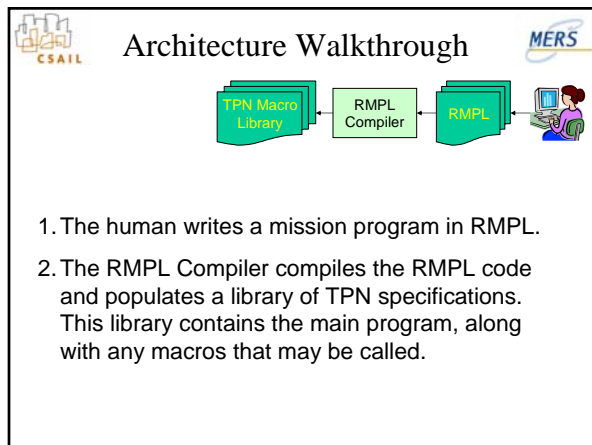
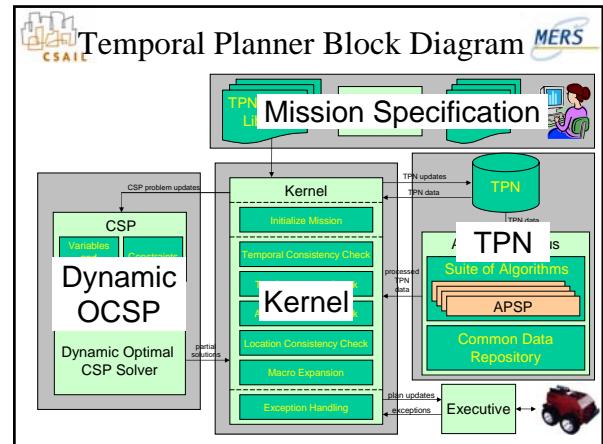
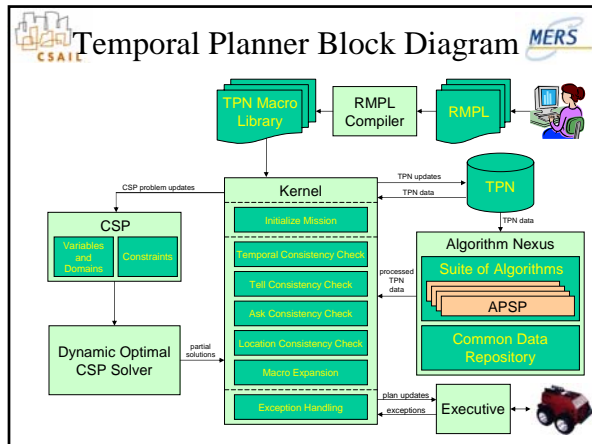


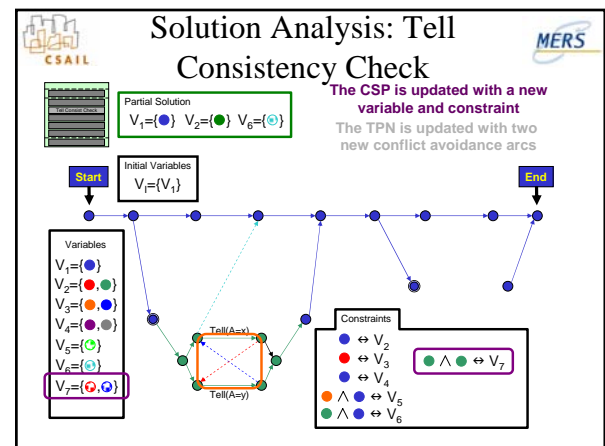
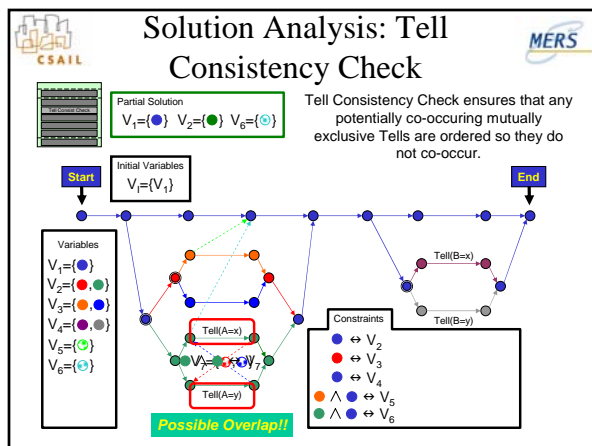
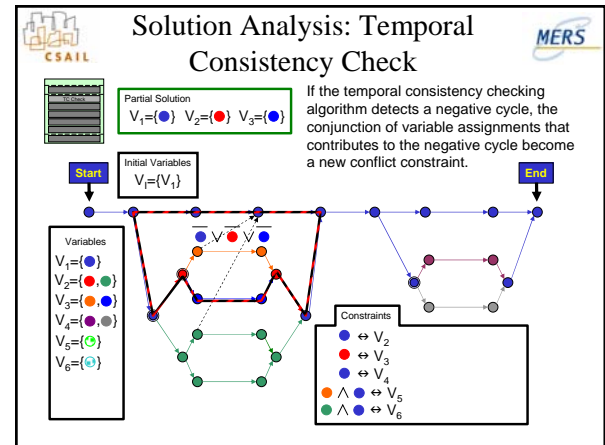
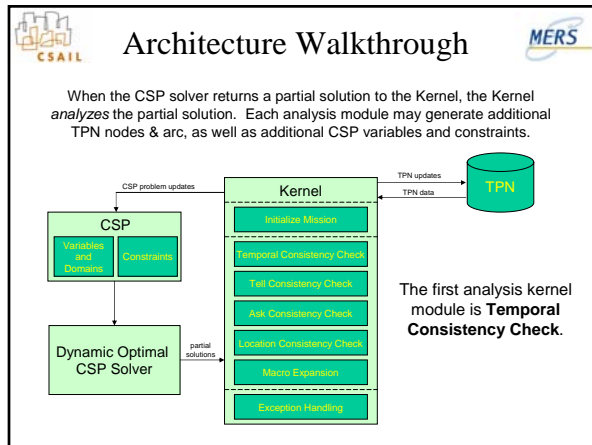
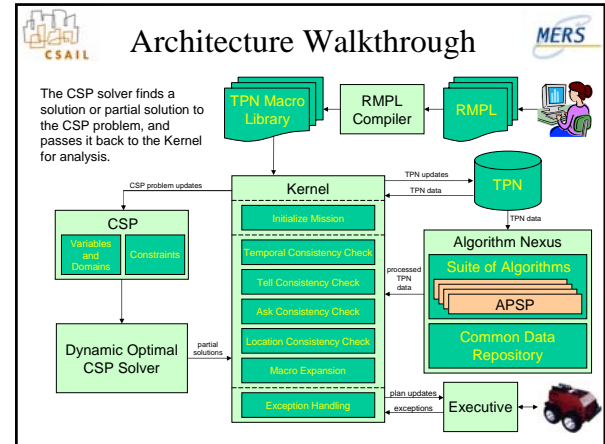
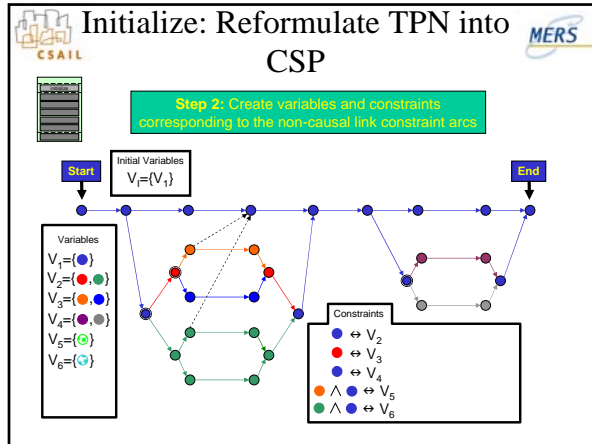


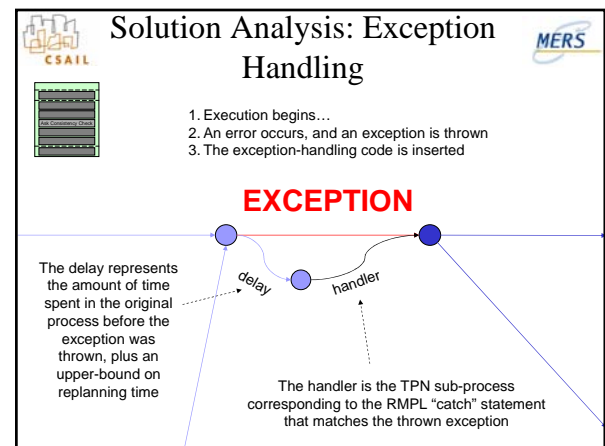
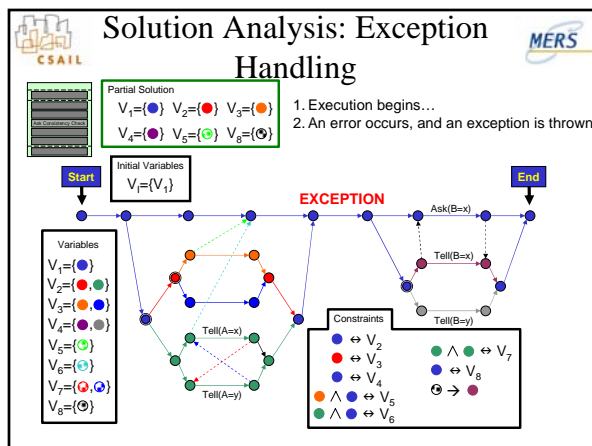
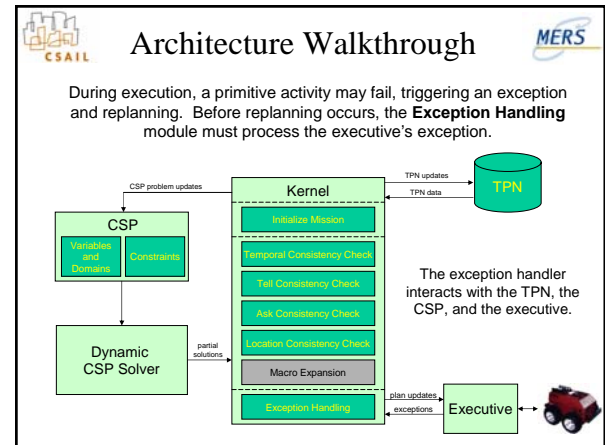
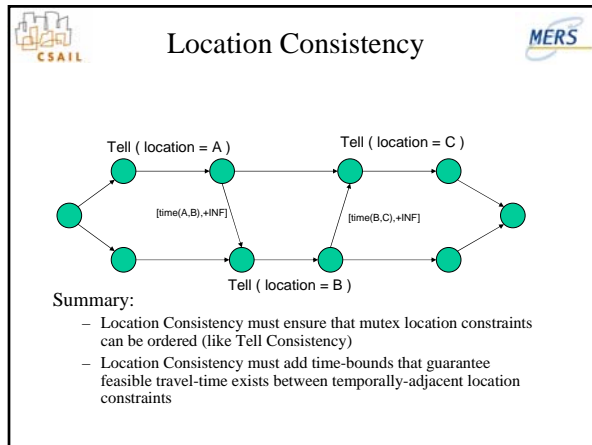
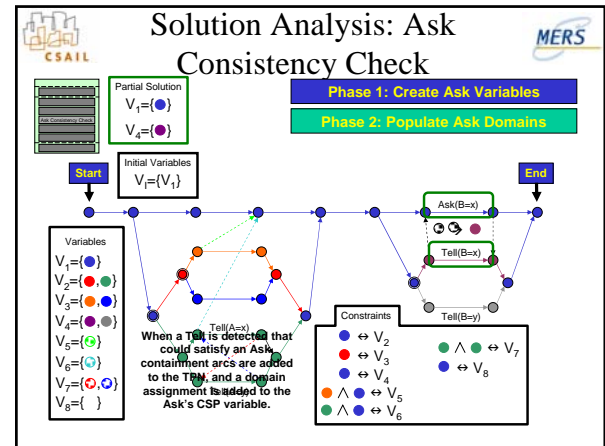
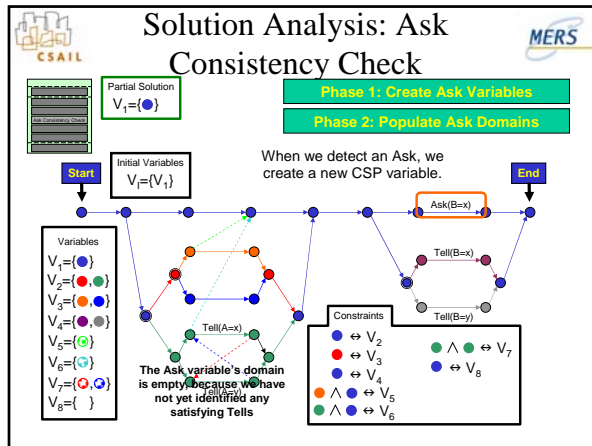
## What is a TPN?

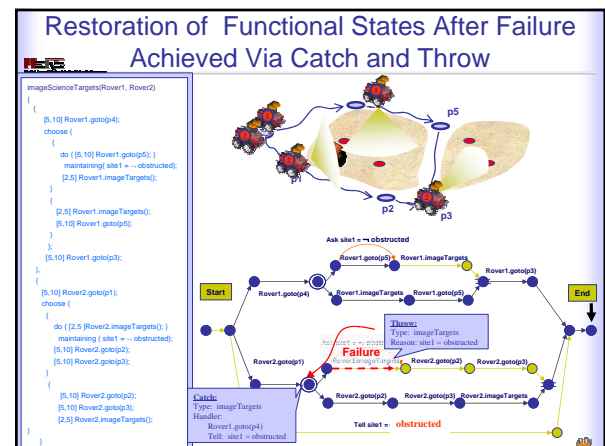
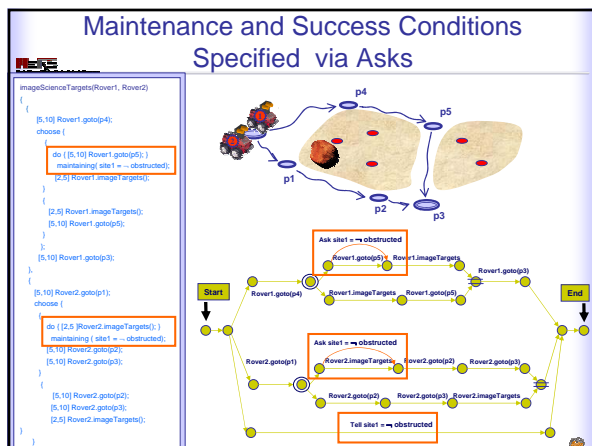
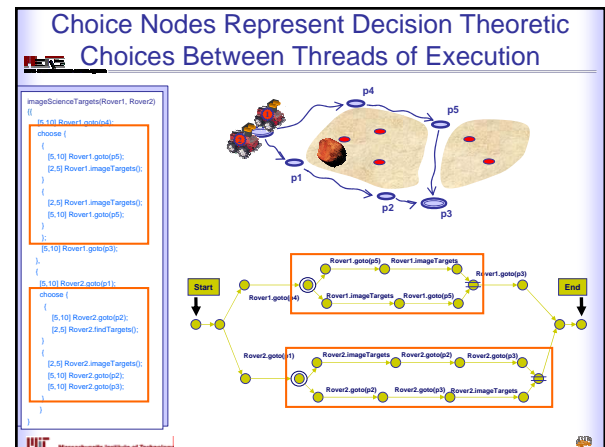
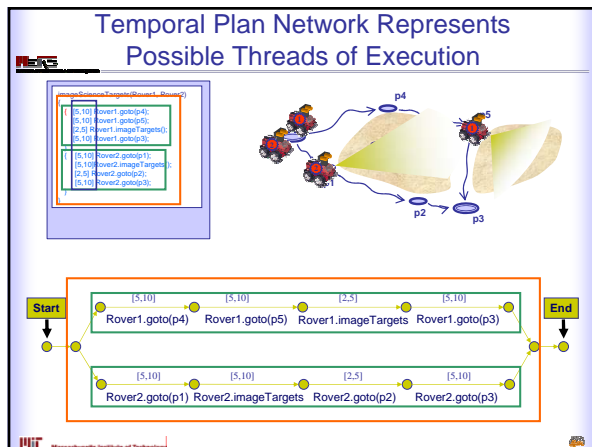
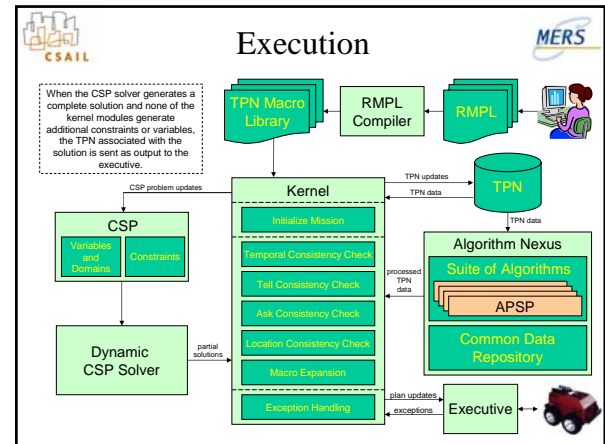
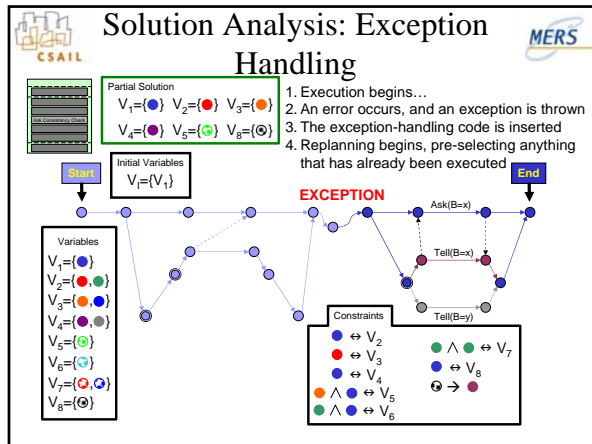
- A Temporal Plan Network (TPN) is
  - A Simple Temporal Network extended to support:
    - Multiple Redundant Methods
    - Method Deprecation and Regeneration
    - Optimal Planning
  - Adds a choice node
  - Adds exceptions
  - Adds cost/rewards to arcs
  - Adds online replanning













## Wrap up

1. Supports development of very complex missions by
  - (i) raising the level of programming to coaching (from commanding); (ii) Sub-plan and component reuse.
2. Handles recovery (with graceful degradation) from:
  - (i) Robot health failure; (ii) Temporal plan failure
  - (iii) Failed contingencies
3. Uses least commitment temporal planning for optimal solutions
  - Handles incremental replanning for contingencies
  - Continuous monitoring and replanning

● Complexity  
 ● Robustness  
 ● Optimality

