CSAIL
MIT COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE LABORATORY

Model-based Embedded & Robotic Systems group

Resolving Unachievable Goals through Collaborative Diagnosis

- Aero & Astro PhD Proposal Defense

Peng Yu

September 30th, 2014

Opportunity

- We are working with robots in industry and in our daily lives.

Challenges

- We don’t trust our robots:
  - **Communication barrier**: they do not understand plain English.
  - **Unreliable**: they often can’t do what they are told.
  - **Uncommunicative**: they do not communicate why they fail.
  - **Risky**: robots do not understand risk in real world.

Objective

- Enable robots to be **trustworthy teammates**.
  - Working with robots should be as easy and as safe as with humans.
- Trusted robots should embody four behaviors:
  - Communicate simply.
  - Explain causes.
  - Propose alternatives.
  - Sensitive to risk.
- Solution: Uhura executive.
- Focus: real-world task planning problems.

Outline

- Example Interaction
- Problem Statement
- Approach
- Progress
- Thesis Plan

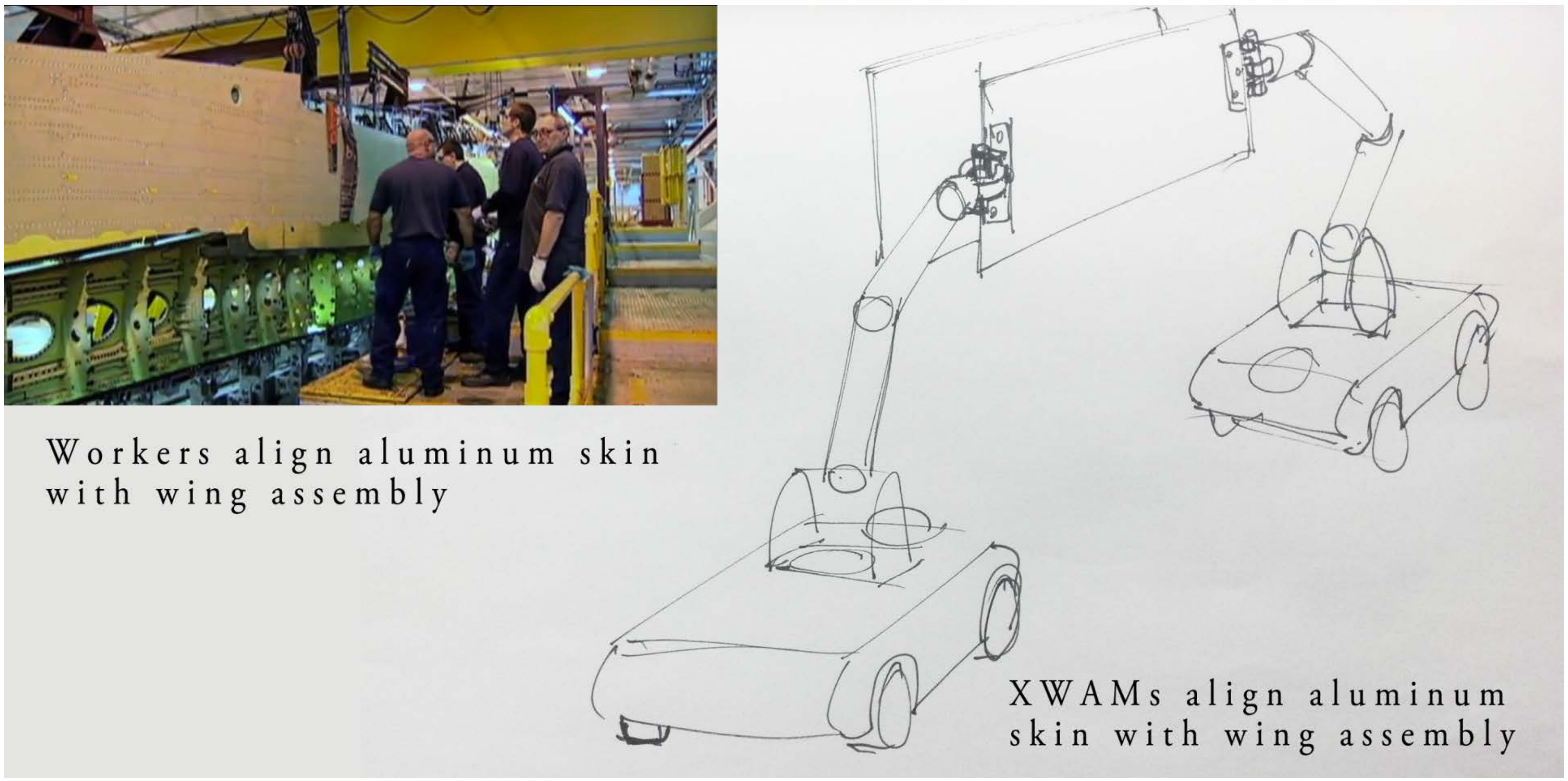
Example: Wing Sub-assembly for Aircraft Manufacturing

- A technician is working with a team of industrial robots on the production line.
- His task is to assemble two wing boxes.

A Completed Wing Sub-assembly

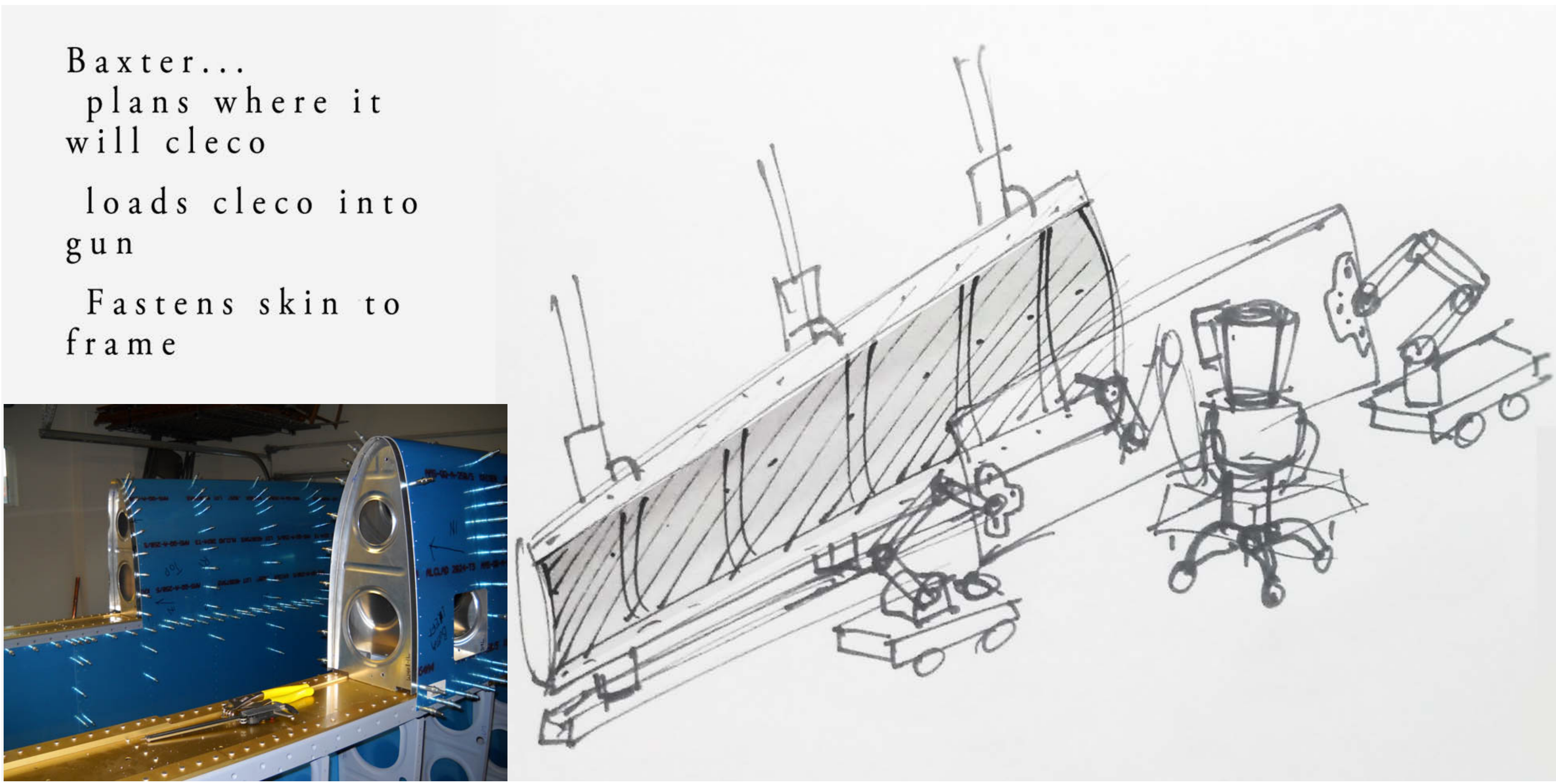
(Simplified) Wing Assembly Procedures

- Step 1: **Lift and Align** aluminum skin with a wing box assembly.
  - This is done by a pair mobility platforms.



(Simplified) Wing Assembly Procedures

- Step 2: **Rivet** the skin to the wing box using Cleco fixture.
  - This is done by a riveting robot (Baxter).



An Unexpected Failure Just Occurred

- The assembly process is interrupted by the unexpected failure of the riveting robot.
  - The plan is no long feasible, since no other robot is capable of riveting wing boxes.

- The technician turns to his **decision support system**, Uhura, to discuss about the impact and options to repair the broken plan:

A Failure-Driven Dialog For Problem Resolution

I want to assemble two wingboxes by 7pm.

1 Communicates Simply

I cannot find a plan to complete our task.

Why? Because ***Riveting is necessary*** *for making the wing box, but the remaining robots are incapable of riveting.*

2 Explains causes

OK, what options do we have?

A Failure-Driven Dialog For Problem Resolution

3 Propose Alternatives

Alternatively, we can still complete the tasks for today if the riveting robot is back online by noon.

Can we have the riveting robot

hour?

No. The workshop is repairing it, but it will be offline for ***at least 12 hours.***

A Failure-Driven Dialog For Problem Resolution

4 Sensitive to Risk

*OK, can you **call in off-duty staff** to help us?*

*If he arrives in 2 hours, there is **80% chance** that we can complete the task by 8pm.*

*I can call in staff to help, but the wing assembly **must be completed** before 7pm.*

A Failure-Driven Dialog For Problem Resolution

Could you ***help rivet the wing boxes*** *until the staff arrives?*

You are qualified for this task. We can then complete the wing repair on time.

OK, lets go with that plan.

Recap of Objective

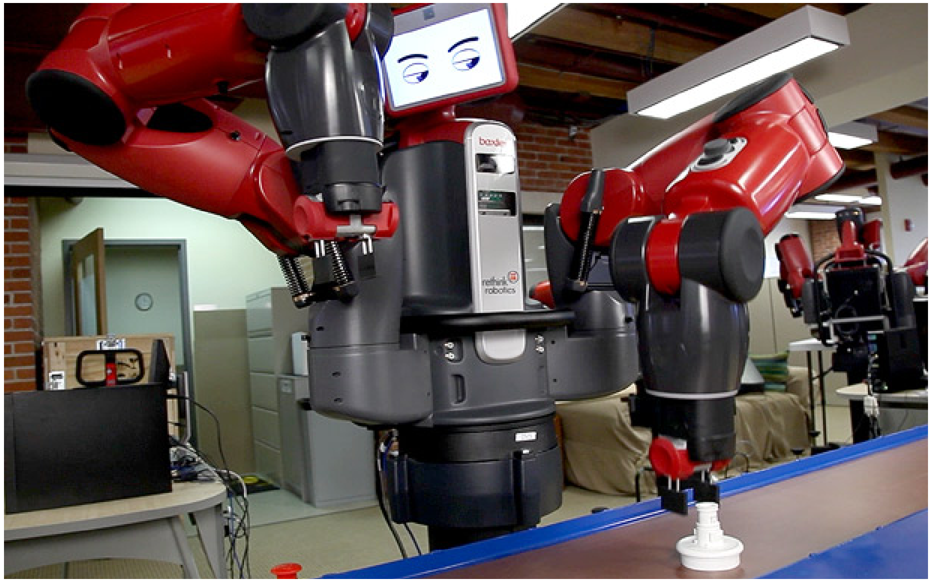
- Enable robots to be **trustworthy teammates**.
  - Working with robots should be as easy and as safe as with humans.

- Trusted robots should embody four behaviors:
  - Communicate simply.
  - Explain causes.
  - Propose alternatives.
  - Sensitive to risk.

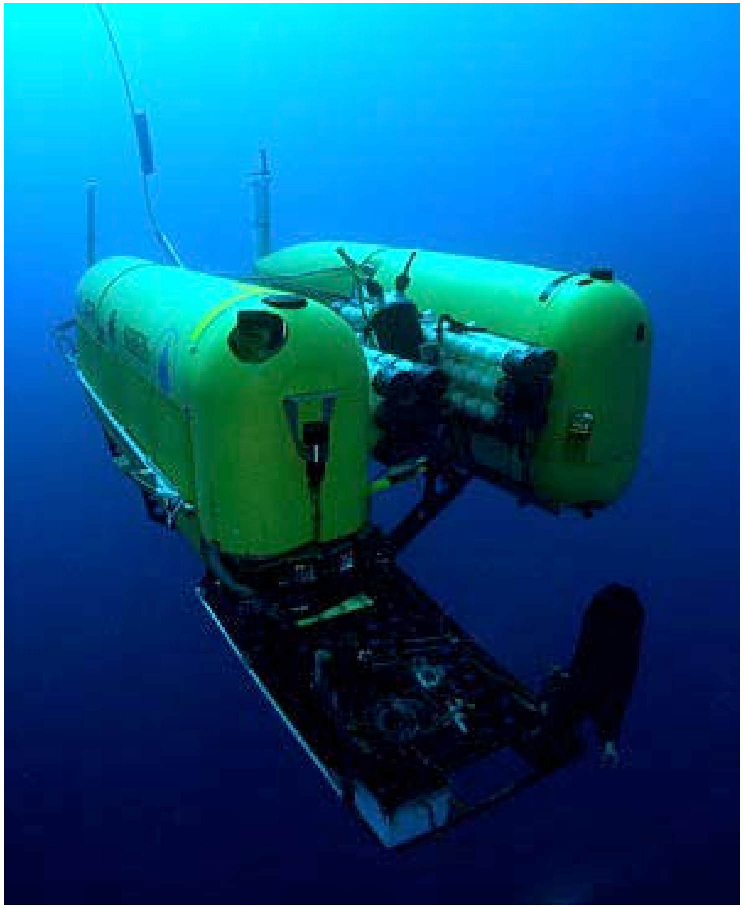
- Solution: Uhura executive.
- Focus: real-world task planning problems.

Demonstration

- Demonstrate Uhura on: 



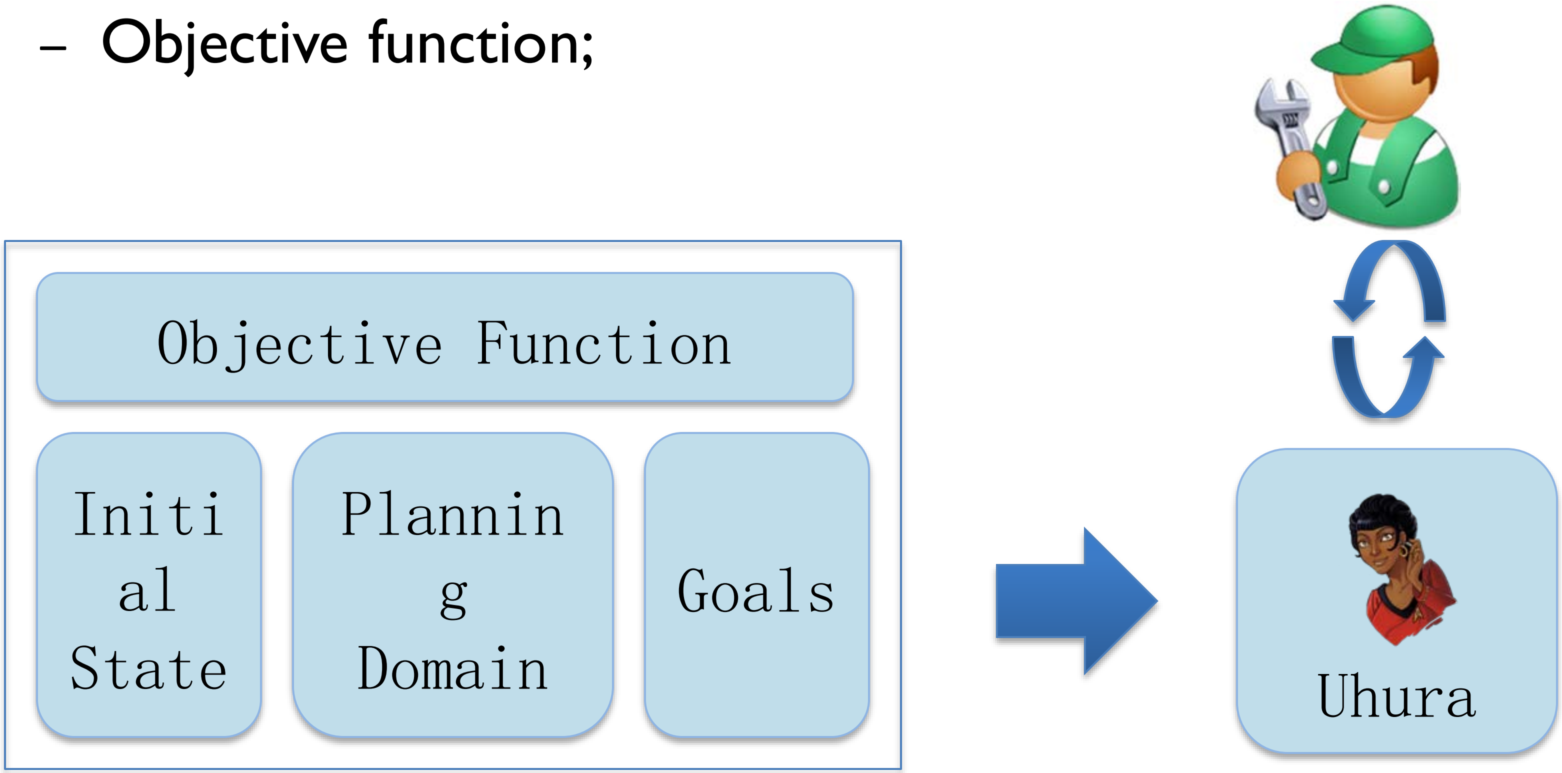
Collaborative manufacturing tasks



Planning missions for deep-sea expeditions

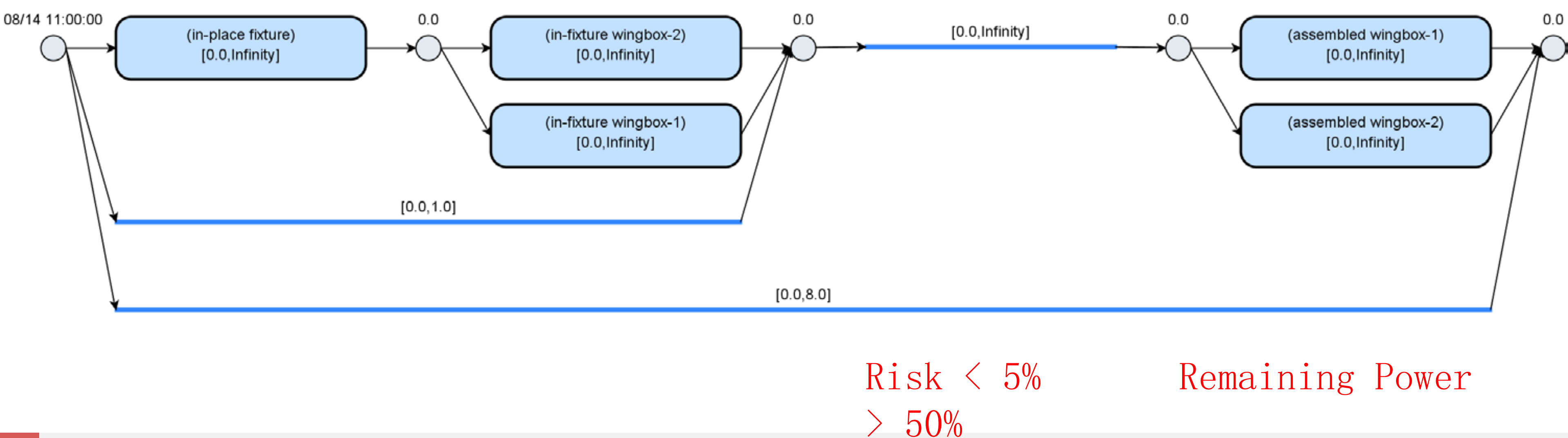
Problem Statement

- Input to Uhura is a **temporal planning problem**[5]:
  - Goals;
  - Initial state;
  - Planning domain;
  - Objective function;



Time-evolved Goal Representation

- Goals are represented as Qualitative State Plans (QSPs), with three key elements:
  - **Episodes**: desired state trajectories.
  - **Temporal**: timing requirements between episodes.
  - **Resource** and **Chance** constraints: desired resource levels and risk bound over episodes.



Planning Domain Specifications

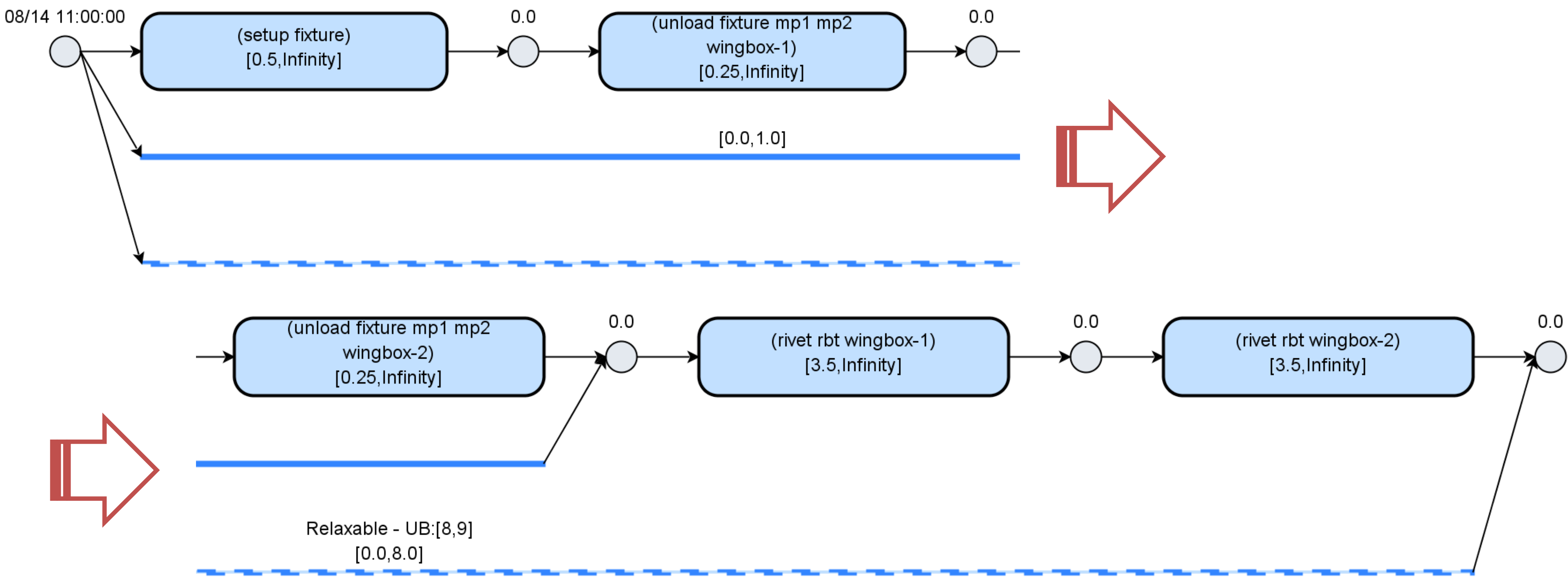
- Extend PDDL descriptions ("STRIPS")

```
(:durative-action rivet
 :parameters (?r – rivet-robot
   ?w - wingbox)
 :condition
         (and (at start (in-fixture ?w))
                  (at start (available ?r)))
 :effect
         (and (at start (not (available ?r)))
(at end (available ?r))
(at end (assembled ?w))
(at end (decrease (battery-level ?r) (normal 0.1 0.05))))
:duration (= ?duration (normal 3.5 0.5)))
:resource ())
```

- with
  - Simple temporal constraints (Simple Temporal Network).
  - Resource constraints (Simple Resource Network).
  - Uncertainty (pSTN, pSRN).

Solutions

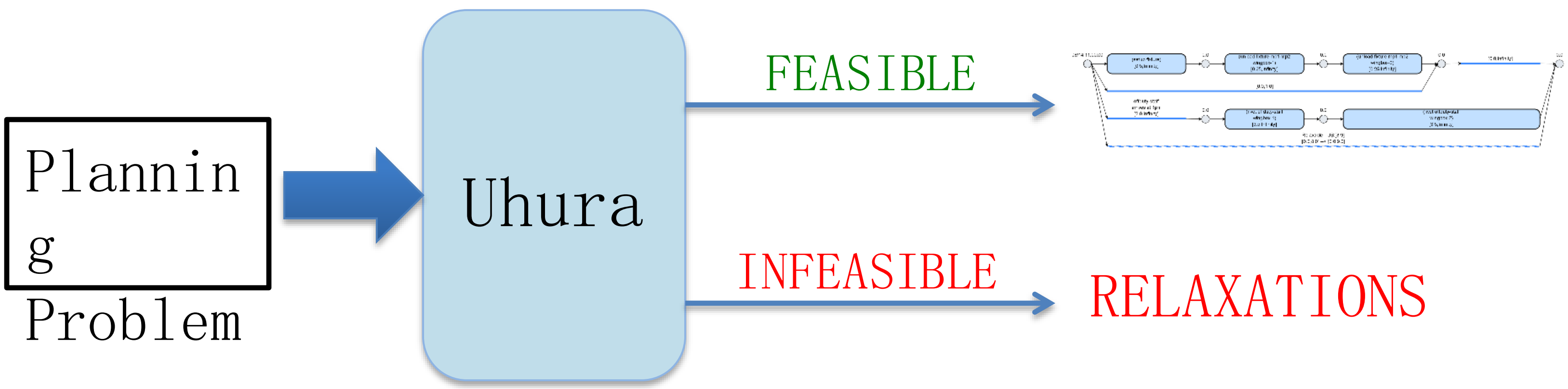
- A solution is a candidate plan that:
  - Satisfies all goal states and constraints in the QSP.
  - Meets all specifications in the planning domain.
  - Is executable.



For Over-subscribed Problems

- A planning problem maybe over-subscribed.
  - that is, no feasible plan can be found for it.

- Uhura can compute a set of **relaxations** to the problem that resolves the over-subscription.



- Relaxations apply to both goals and planning domains.

Types of Relaxations

✓ **Time**

‘Extend the assembly time by one hour.’

– **Resource**

‘Reduce the power consumption by 5 kWh.’

✓ **Risk**

‘Increase the acceptable risk level from 5% to 20%.’

– **Goal**

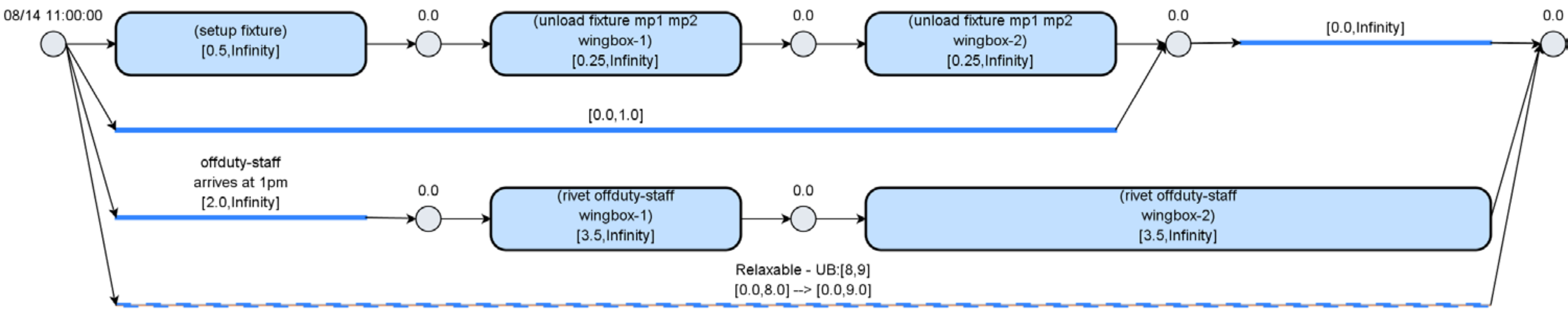
‘Assemble only one instead of two wingboxes today.’

– **Action**

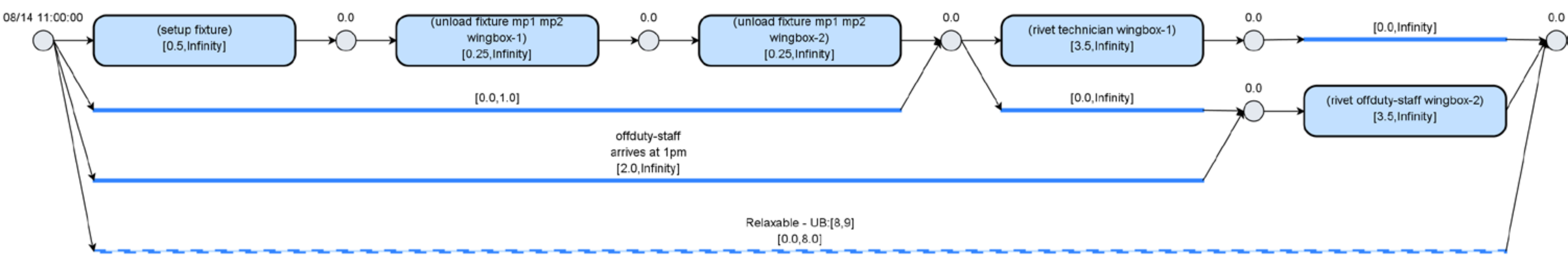
‘Ask an off-duty staff to rivet the parts.’

Example Relaxations

- 'Asking the technician to call in an off-duty staff'.



- 'Asking the technician to rivet the wingbox'.



Approach: Research Questions

1. Collaborative plan diagnosis and repair.

2. Risk assessment and management.

3. Generate succinct explanations.

1. Collaborative Plan Diagnosis and Repair

- Over-subscription is caused by **incompleteness** and/or **inconsistency**.

- Diagnosis:
  - Incompleteness: find the subset of goals that are not supported by the planning domain.
  - Inconsistency: detect sets of goals and constraints that are in conflicts.

- Repair:
  - Find a set of relaxations that addresses **all** issues.
  - Restore the feasibility of the problem.

Uhura as a PAV Pilot

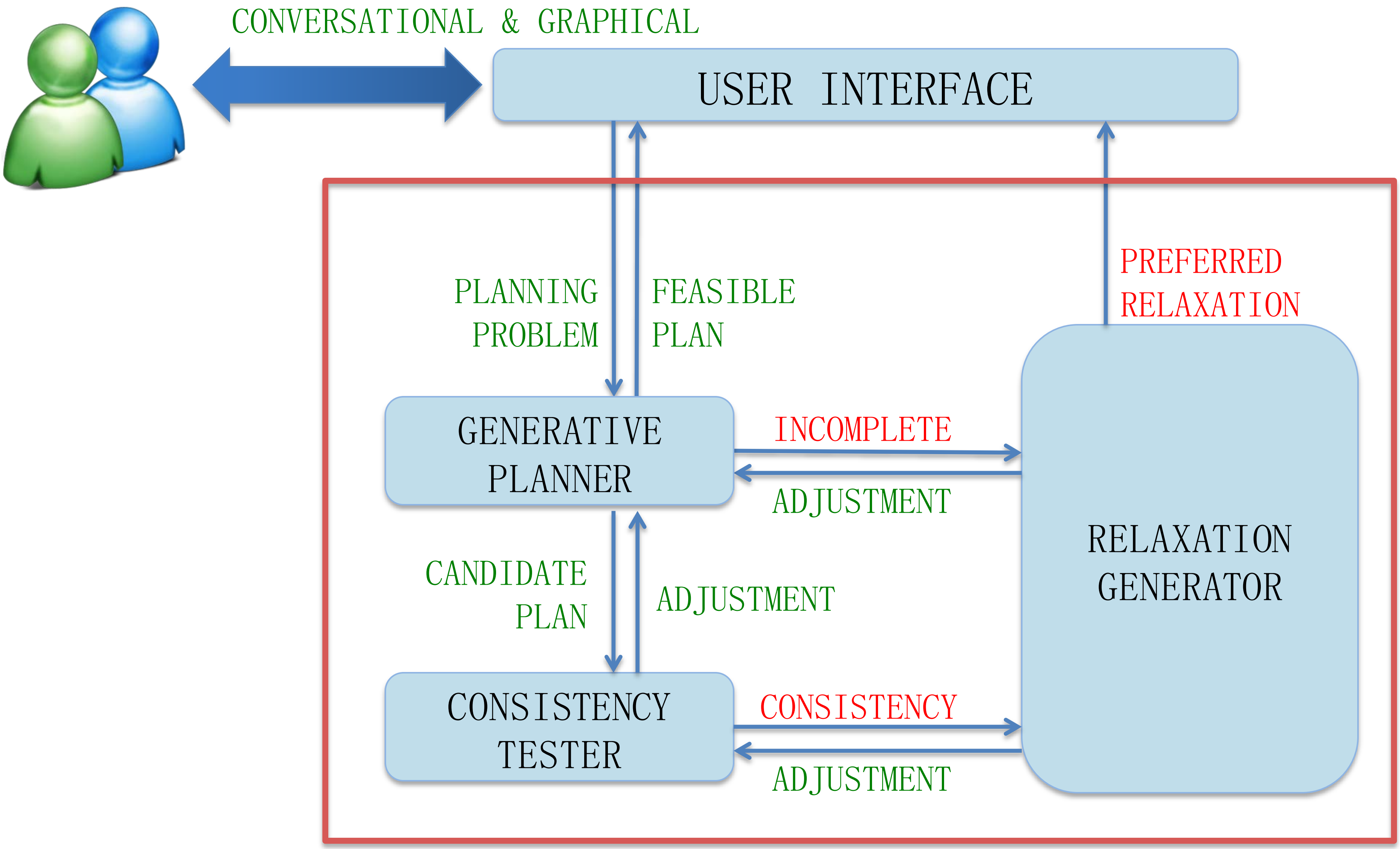
Possible Collaborations with Humans

- For approval to a candidate relaxation.
  - "Extend the assembly time by one hour, is it ok?"
- For preferences and constraints.
  - "How long does it take to repair the riveting robot?"
- For assistance.
  - "Can you rivet the wingbox before the staff arrives?"

Finding Good Relaxations

- Good relaxations are **similar** alternatives.
- Temporal and resource relaxations: similarity is measured by the linear distance.
  - "30-minute delay" is better than "3-hour delay".
- Goal and action relaxations: similarity is measured by **semantic relations**[7].
  - "Chinese → Thai restaurant" is better than "Burger King".
  - "Trader Joe's → Star Market".
  - "roast turkey → roast chicken".

Key Elements of Uhura Architecture



Key Elements of Uhura Architecture

- **Planner**:
  - Generating complete plans.
  - Detecting **incompleteness**.
- **Tester**:
  - Evaluating plan activities, temporal and resource constraints.
  - Detecting **inconsistency** between goals and constraints.
- **Relaxation generator:**
  - Generating relaxations to resolve any incompleteness and inconsistency.

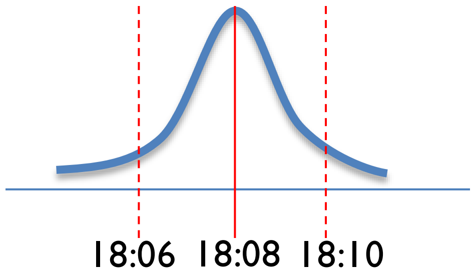
2. Risk Assessment and Management

- Pin-point the source of risk in a plan:
  - ‘Off-duty staff will arrive in 2 hours with only 80% chance.’

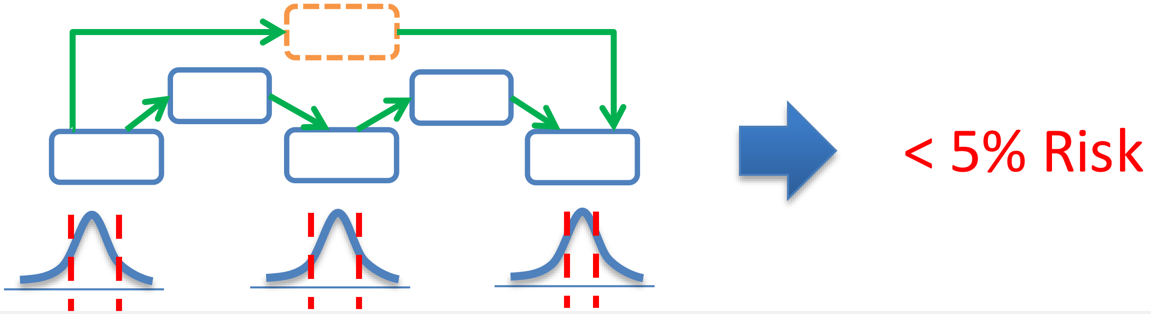
- Suggest trade-offs between risk-taken and performance:
  - “Accept 5% more risk of not having the staff helping assembly, so that I can complete the task on time”

Chance-constrained Temporal and Resource Problems

- Uncertainty exists in both time and resource.
  - "Bus is likely to arrive at 6:08, with a SD=2 minutes."

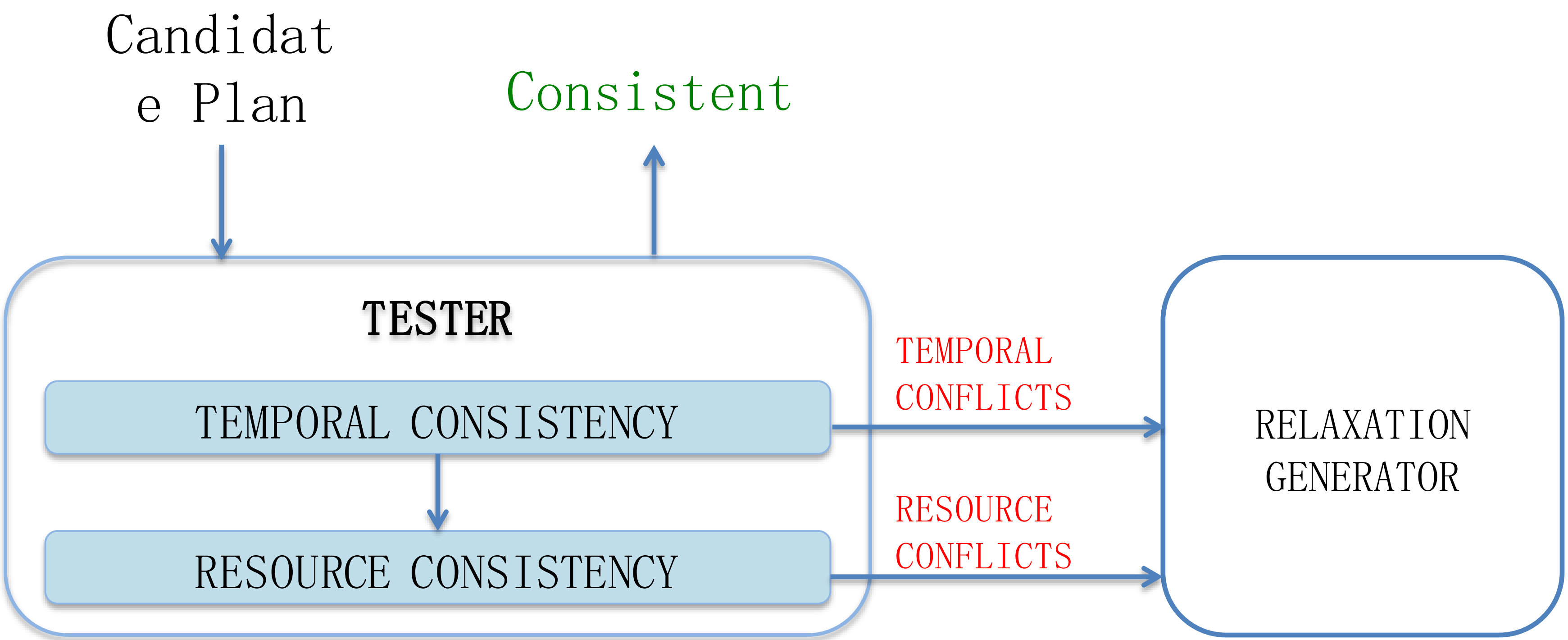


- Chance-constrained probabilistic STN and SRN:
  - Uncertainty in both temporal duration and resource consumption/generation.
  - A risk-bound on the chance of violating any constraint.



Consistency Tester

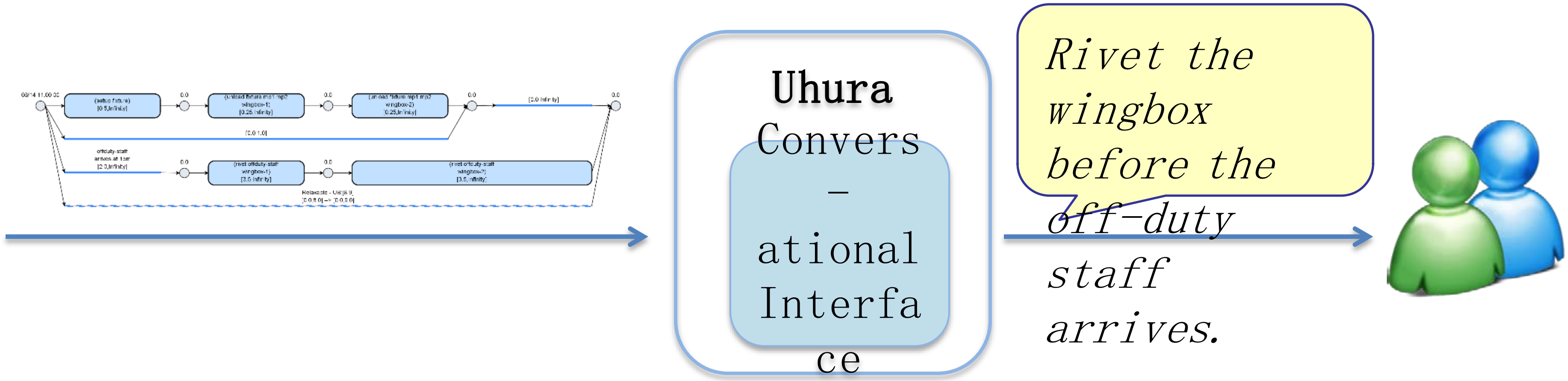
- Evaluate the consistency of candidate plans:
  - If inconsistent, return a set of temporal and resource constraints that are in conflict.
  - The conflict also includes the chance constraint.



3. Generate Succinct Explanations

- Uhura communicates:
  - **explanations** on failures and resolutions.
  - **reasons** for the robot' decisions and actions.

- Present only key ideas that users can draw conclusion from:
  - Include everything that is hard to find.
  - Omit less important or easy to infer results.



Challenges in Generating Succinct Explanations

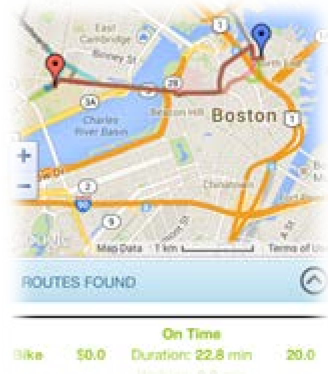
- Definition of 'succinct' explanation.
- Identify key ideas in the explanations for an over-subscribed problem.
- Identify what the users already know, and only present what they do not know.

Progress

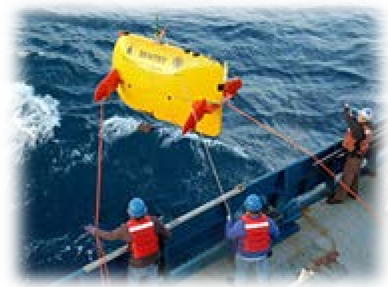
Personal Transportation System

Flexible Manufacturing Test-bed

Commuter Advisory System



Deep-sea Expedition Plan Advisor



Application

Research

Resolving ***Over-constrained*** *Temporal Problems*

Yu, Williams IJCAI 2013

Resolving Temporal Problems with ***Uncertainty***

Yu, Fang, Williams ICAPS 2014

Scheduling Chance-constrained ***Probabilistic*** *Temporal Problems*

Fang, Yu, Williams AAAI 2014

Relaxing ***Chance-constraints*** *to resolve Probabilistic Temporal Problems*

Yu, Fang, Williams AAAI 2015 (submitted)

Research Questions

1. Collaborative plan diagnosis and repair.
   - ✓ Temporal problems.
   - – **Planning** problems

2. Risk assessment and management.
   - ✓ Temporal uncertainty and risk.
   - – **Resource** uncertainty and risk.

3. Generate succinct explanations.

Thesis Plan

- **Current time: 09/2014.**
  - Semantic relaxation of goals: 11/2014 (ICAPS15).
    - Relaxing resource constraints: 01/2015 (IJCAI15).
      - Generating succinct explanations: 09/2015 (AAAI16).
        - System demonstration and thesis writing: Spring 2016.
          - Thesis defense: 06/2016.

09/2014 2015 2016 **06/2016**

Appendix

- Additional examples and technical details.
- Reference publications.
- More demos.

Objective Function

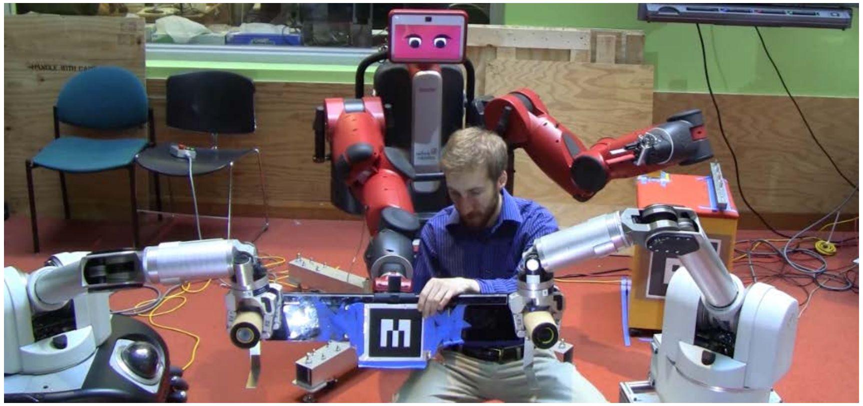
- The objective function specifies the users' preferences over the relaxations for the problem, and is defined over alternatives to the goals, constraints and actions.

- Example: in the assembly example, different relaxations have different costs for the technician:
  - Delaying the completion time costs $10,000.
  - Calling in an off-duty staff costs $300.
  - Joining the riveting himself costs $0.

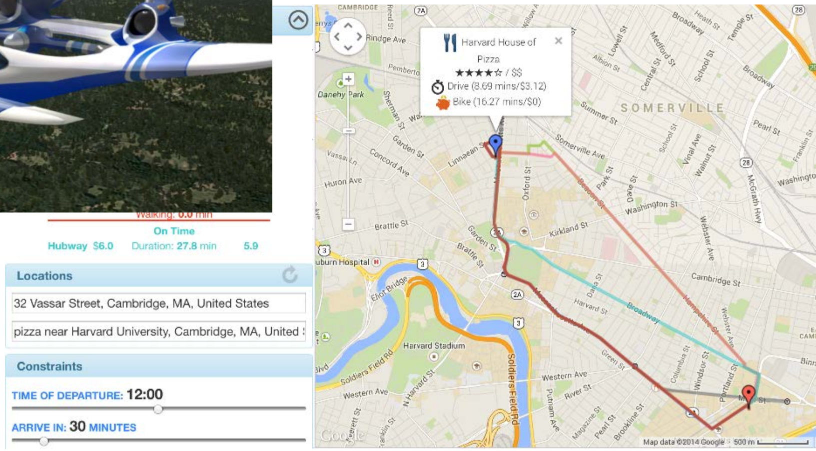
- This helps us prioritize the resolutions and simplify the interaction.

Applications

- **Peer-to-peer**: Human-Robot Collaboration in manufacturing tasks.



- **Supervision**: Trip advisor for WHOI deep-sea explorations, PTS and city commuters.

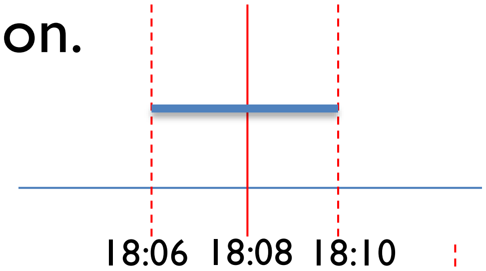


Remaining Work

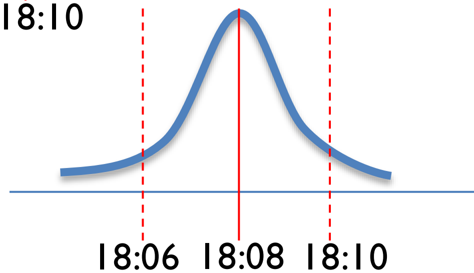
- Plan generation and incompleteness detection.
- Resource uncertainty.
- Semantic Relaxation.
- Simple explanation.

Uncertainty in Time and Resource

- For each action, the duration and resource consumption may be random cannot be determined by Uhura.
- Example: transit from MIT to Logan airport may take any time between 10 to 100 minutes.
- We model such uncertainty using two approaches:
  - A simple set-bounded representation.

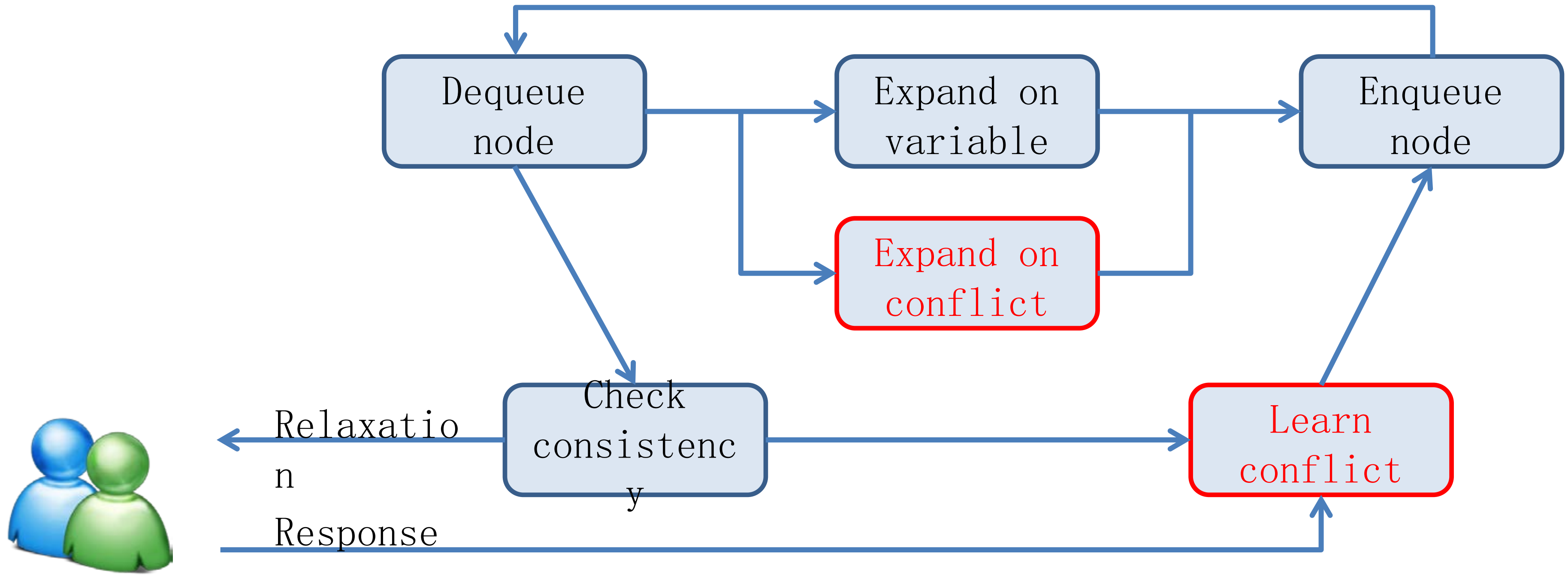


  - A probabilistic distribution representation.



Resolving Over-constrained Temporal Problems

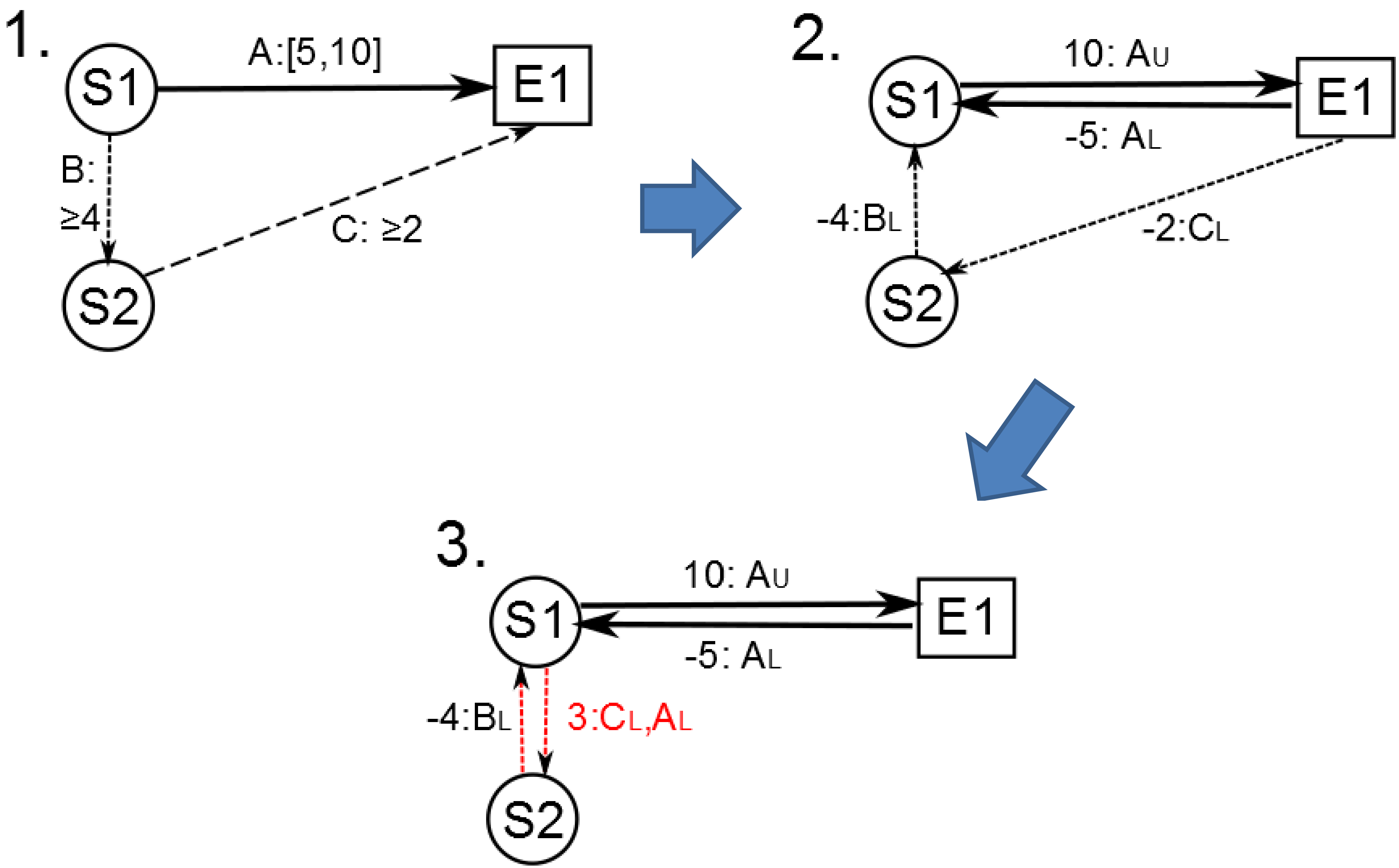
- Uhura enumerates relaxations in best-first order:
  - It searches over subsets of constraints by making different variable assignments.
  - It resolves a conflict by relaxing a constraint, partially and completely.



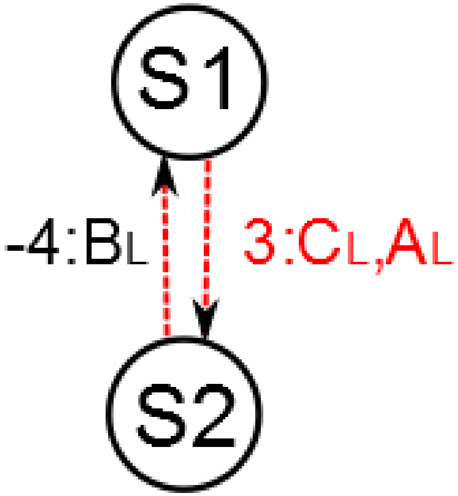
Learn Conflicts From Uncontrollable Problems

- Learning conflicts from controllability checking algorithms is more difficult.
  - For consistency checking, there is a **one-to-one mapping** between the distance edges and the bounds of constraints.
  - No such mapping exists for controllability checking (strong and dynamic) **due to the reduction procedures**, making it difficult to extract conflicts from the reduced graph.
- Key: during the reduction, **record the 'contribution'** of each constraint and duration in the temporal problem.

A Strong Controllability Example



Resolving Uncontrollable Conflicts

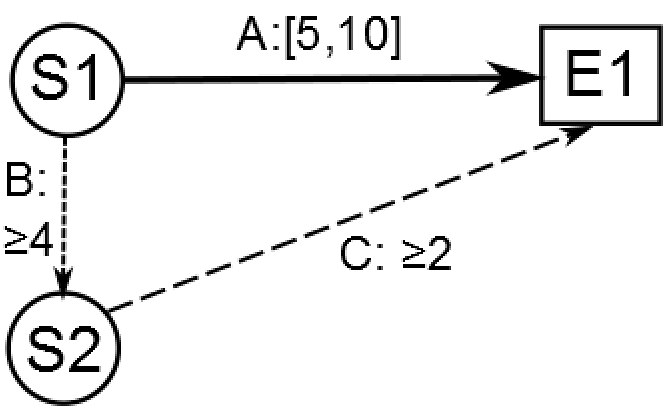


- Constraint for resolving continuous conflict (negative value -1):

$$\Delta C_L + \Delta B_L + \Delta A_L \geq 1$$

where:

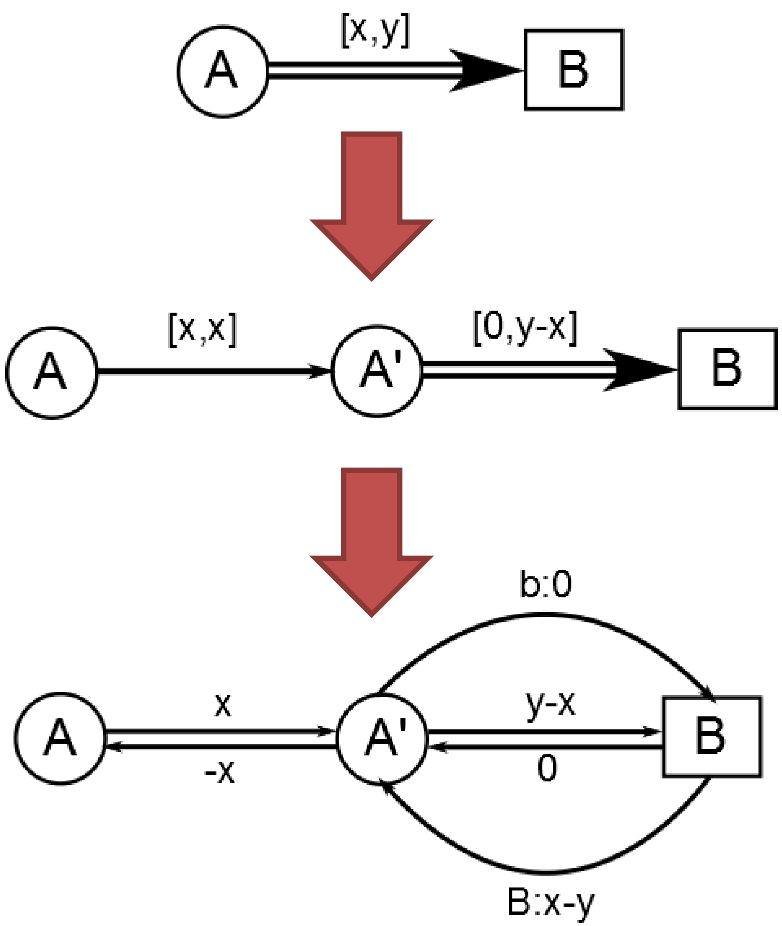
- $\Delta B_L, \Delta C_L$ are relaxations for B and C.
- ΔA_L is tightening for A.



and

$$\Delta A_L \leq 5$$

Learning Dynamically Uncontrollable Conflict



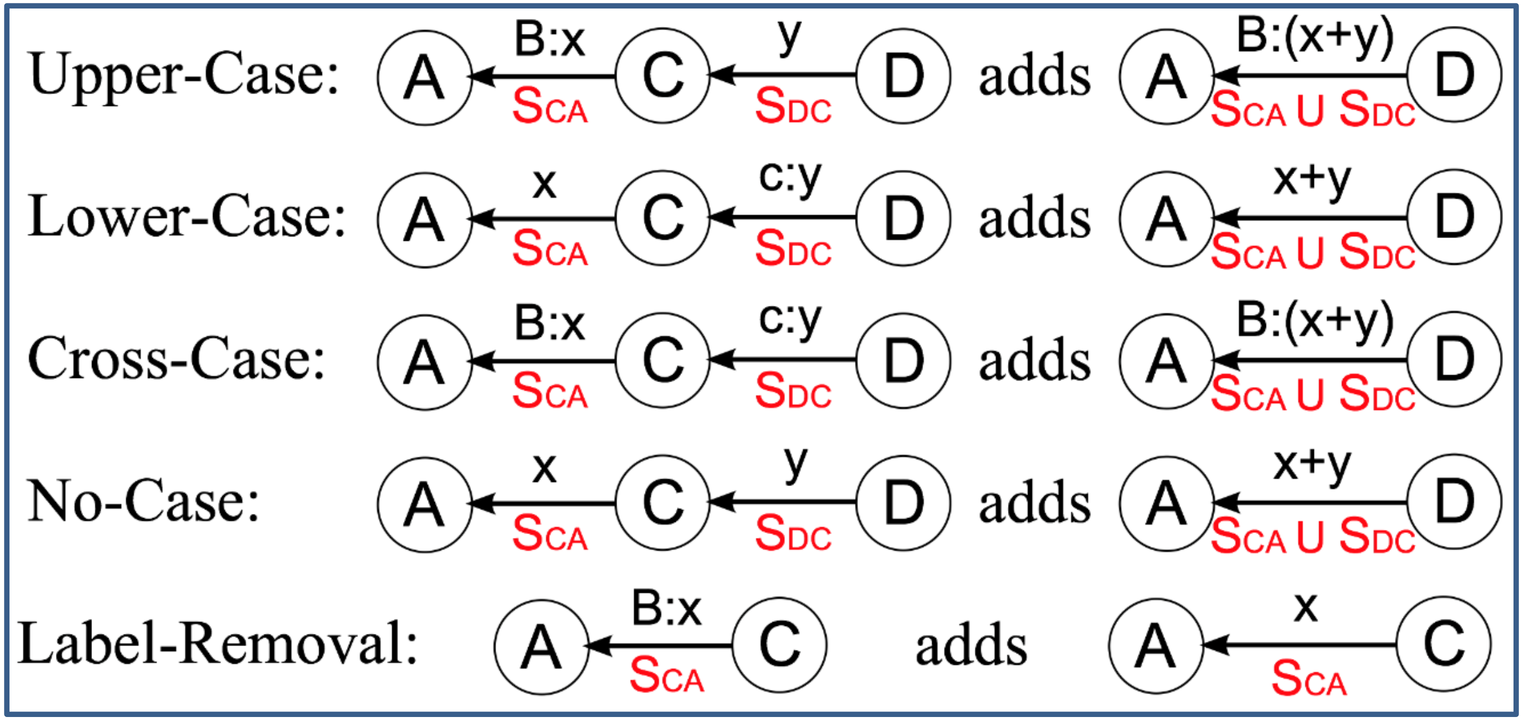
- Record supporting constraints for both **requirement** and **conditional** edges while generating the directed graphs.

$A - (x) \rightarrow A'$	AB_{Lower}
$A' - (-x) \rightarrow A$	$-AB_{Lower}$
$A' - (y - x) \rightarrow B$	$-AB_{Upper}$
	$-AB_{Lower}$
$A' - (b{:}\,0) \rightarrow B$	None
$B - (0) \rightarrow A'$	None
$B - (B{:}\,x - y) \rightarrow A'$	AB_{Upper}

Paul Morris. A structural characterization of temporal dynamic controllability. In Proceedings of the 12th International Conference on Principles and Practice of Constraint Programming (CP-2006), pages 375–389, 2006

Learning Dynamically Uncontrollable Conflict

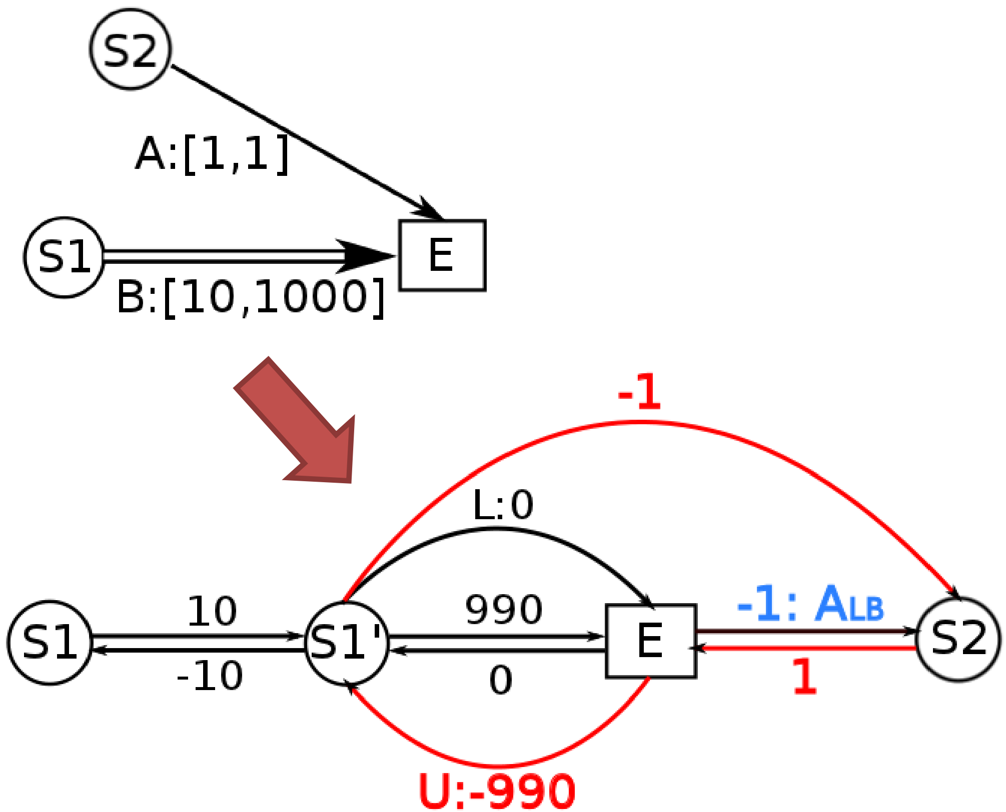
- **Record supporting** constraints and durations during the iterative reduction procedure.



- Note that a constraint may be recorded multiple times during reduction.

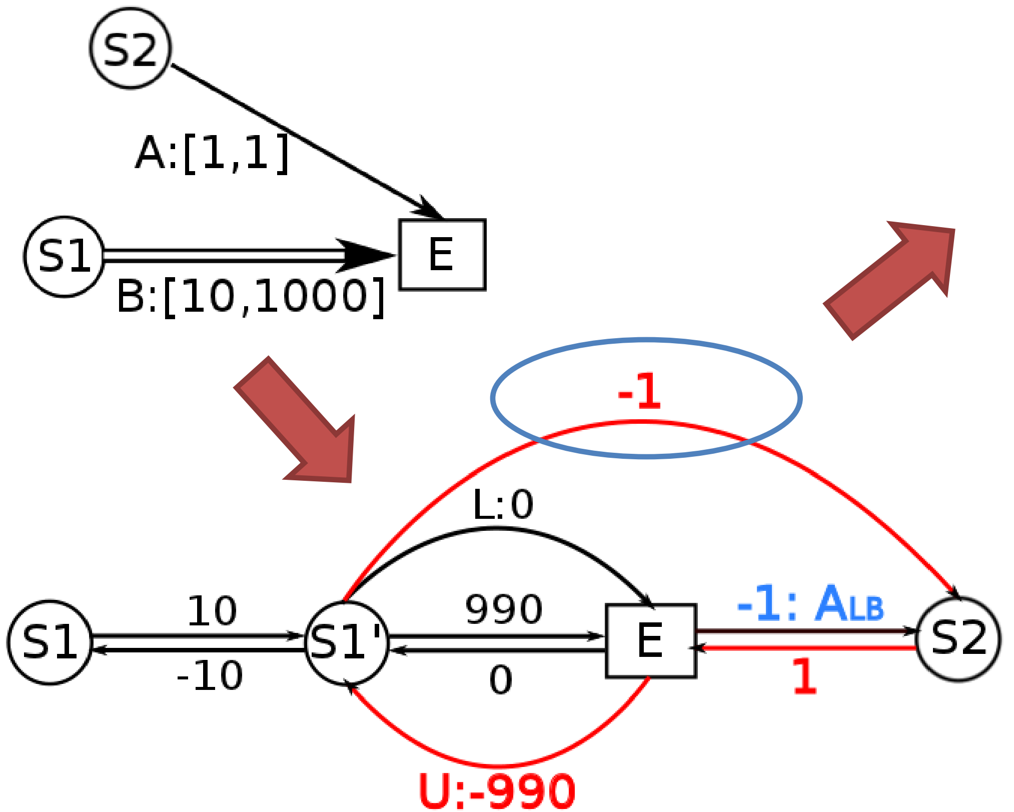
Another Way to Resolve DC Conflicts

- A STNU is dynamically controllable if and only if it does not have a **semi-reducible** negative loop [Morris 2006].
  - We can resolve a conflict by **disabling** reductions that lead to edges in the negative loop.

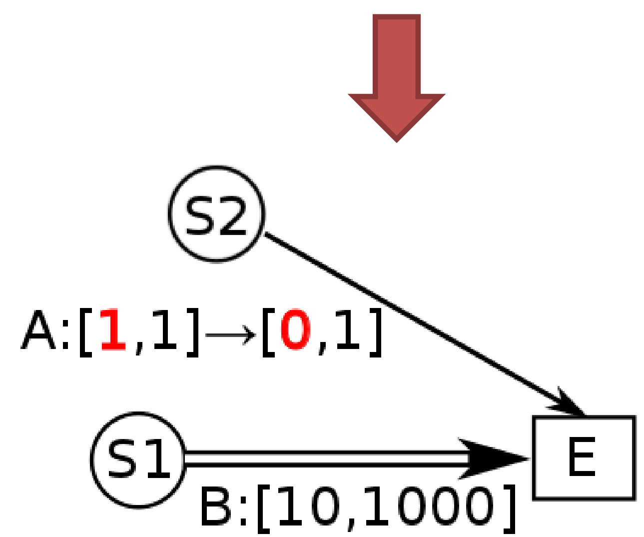


Another Way to Resolve DC Conflicts

- We can resolve a conflict by **disabling** reductions that lead to edges in the negative loop*.



Relax the lowerbound of A ***from 1 to 0*** *to disable the reduction.*



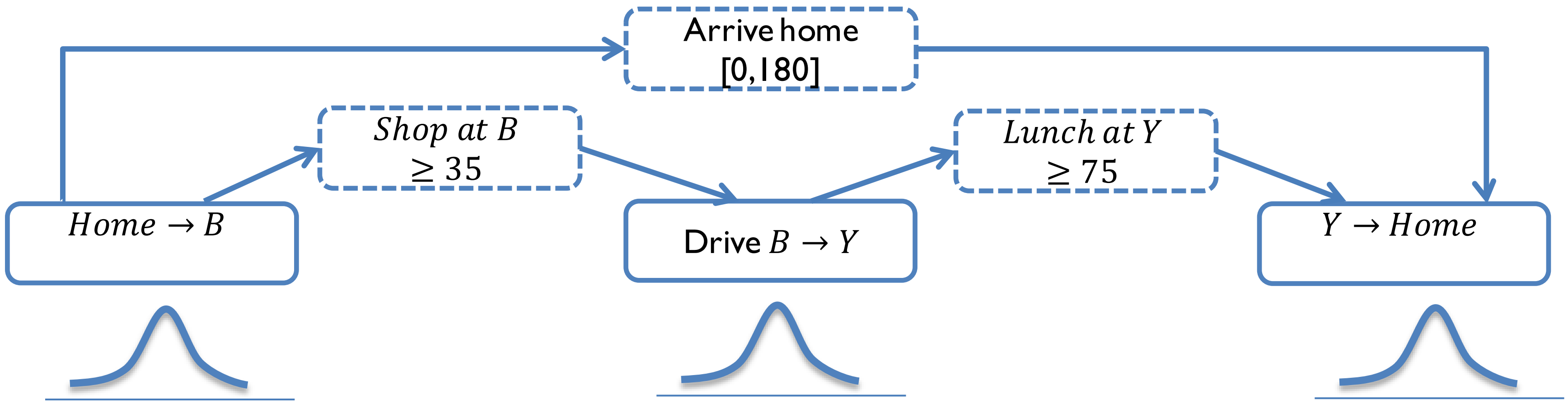
*A STNU is dynamically controllable if and only if it does not have a **semi-reducible** negative loop [Morris 2006].

Chance-constrained Relaxation

- No existing temporal feasibility checking algorithm applies to probabilistic problems.

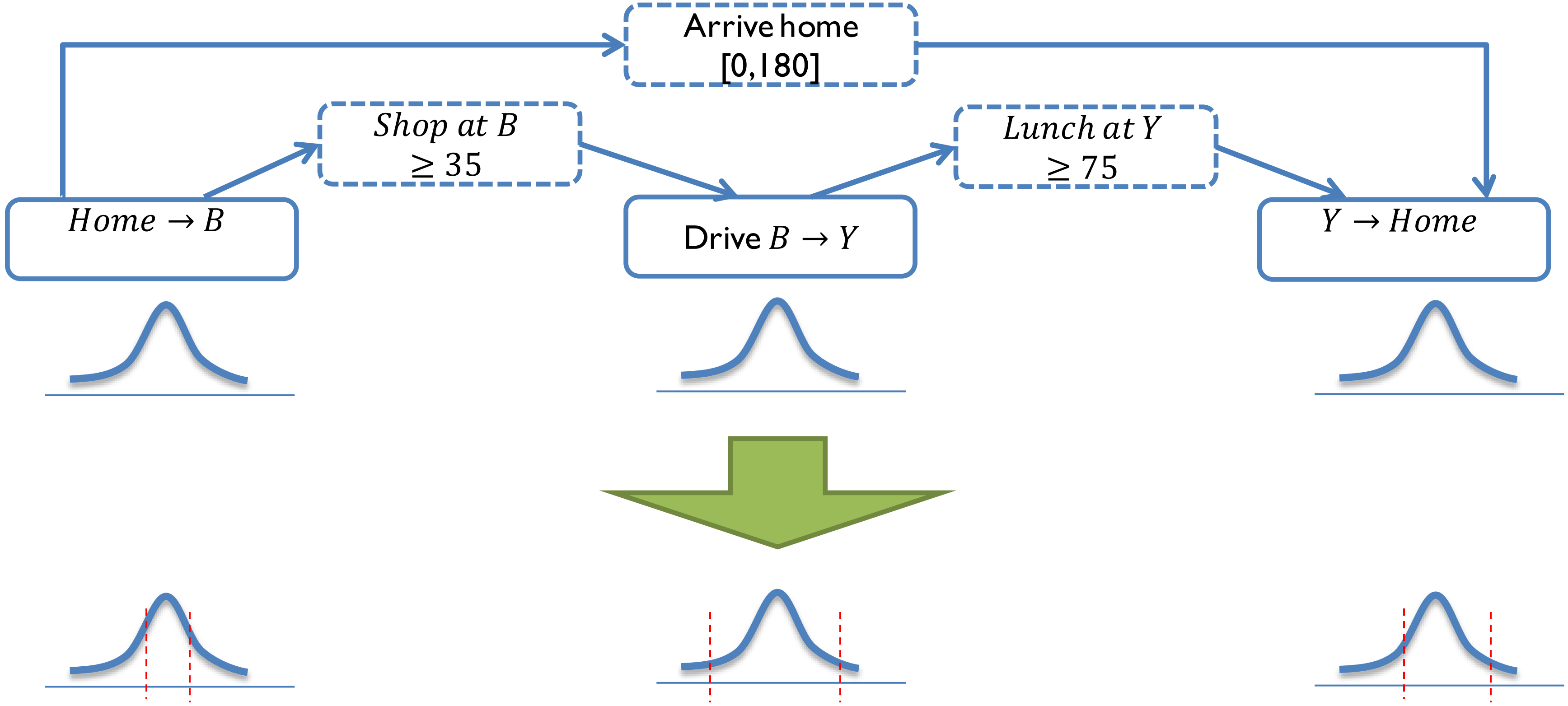
- Key: **ground** the problem to a deterministic one with set-bounded uncertainty.
  - Then iteratively adjust the bounds to satisfy both **chance** and **temporal** constraints.
  - During the process, we can relax constraints to achieve

Example

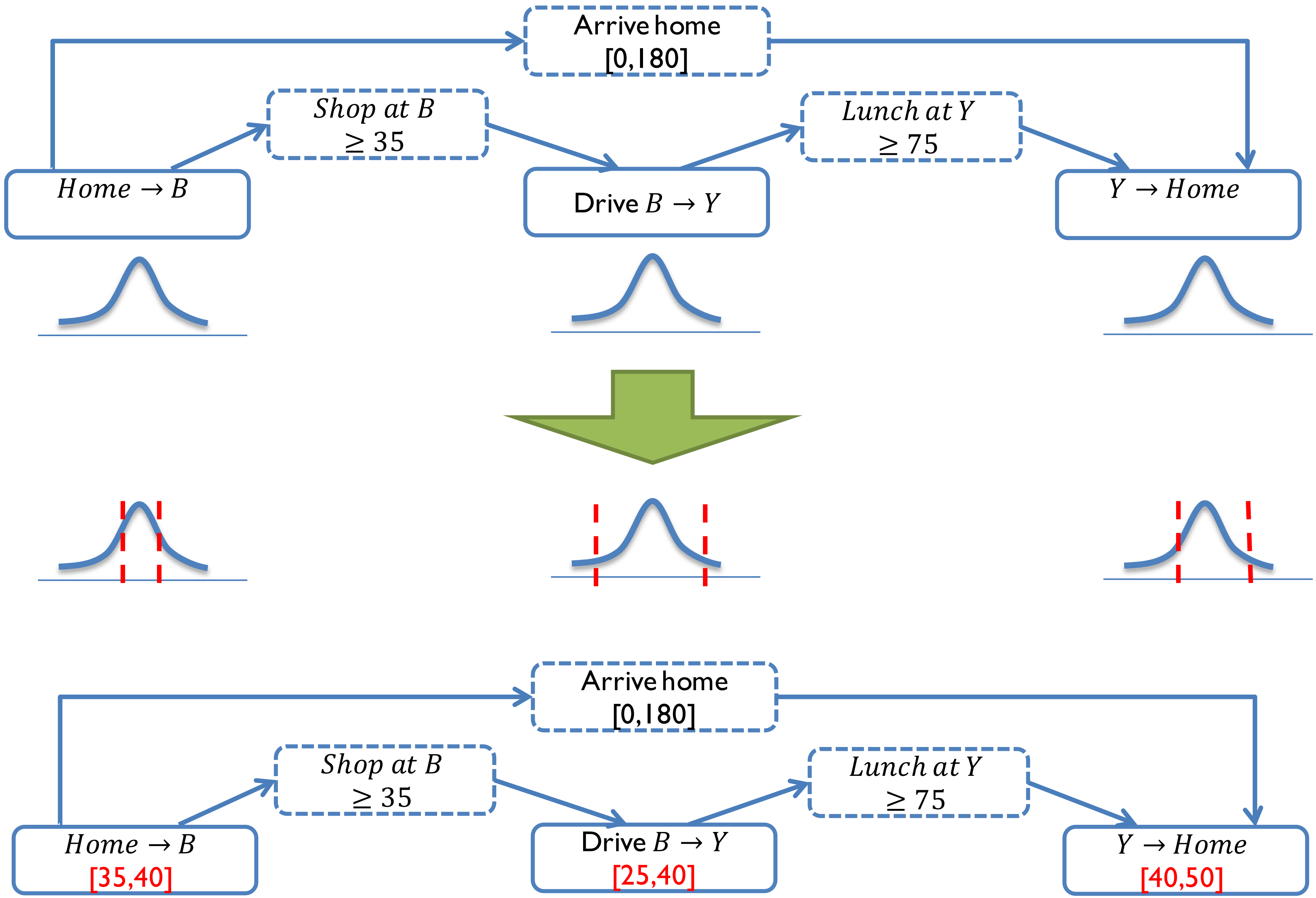


Arrive home
[0,180]
$Shop\ at\ B$
≥ 35
$Lunch\ at\ Y$
≥ 75
$Home \rightarrow B$
Drive $B \rightarrow Y$
$Y \rightarrow Home$

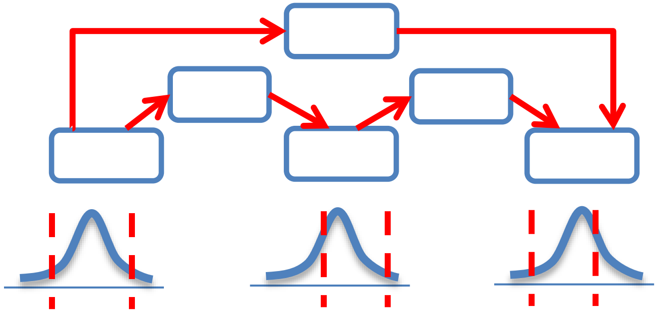
Example


Arrive home
[0,180]
$Shop\ at\ B$
≥ 35
$Lunch\ at\ Y$
≥ 75
$Home \rightarrow B$
Drive $B \rightarrow Y$
$Y \rightarrow Home$

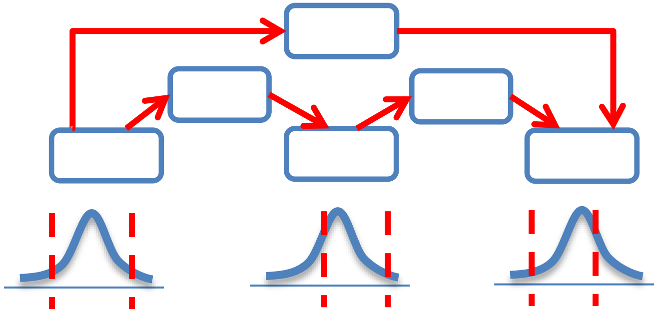
Example

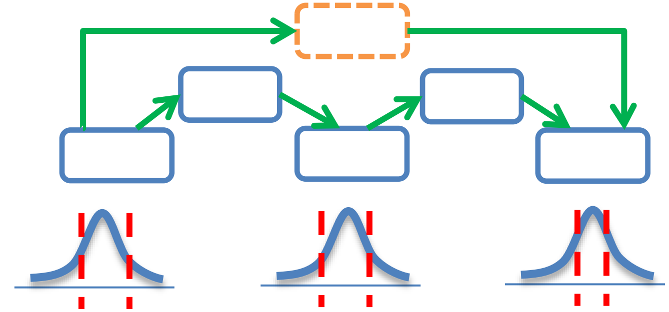

Arrive home
[0,180]
Shop at B
≥ 35
Lunch at Y
≥ 75
$Home \rightarrow B$
Drive $B \rightarrow Y$
$Y \rightarrow Home$
Arrive home
[0,180]
Shop at B
≥ 35
Lunch at Y
≥ 75
$Home \rightarrow B$
[35,40]
Drive $B \rightarrow Y$
[25,40]
$Y \rightarrow Home$
[40,50]

Conflict-directed Risk Allocation + Relaxation



Conflict-directed Risk Allocation + Relaxation





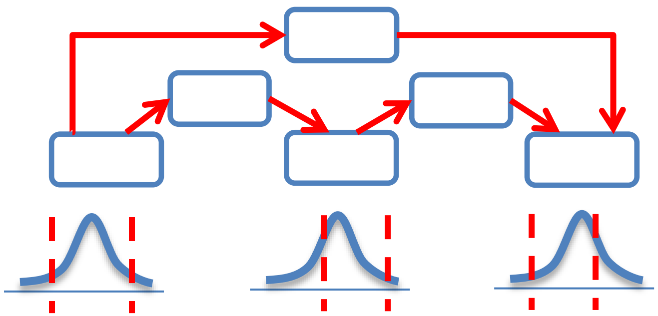
Extract Conflict

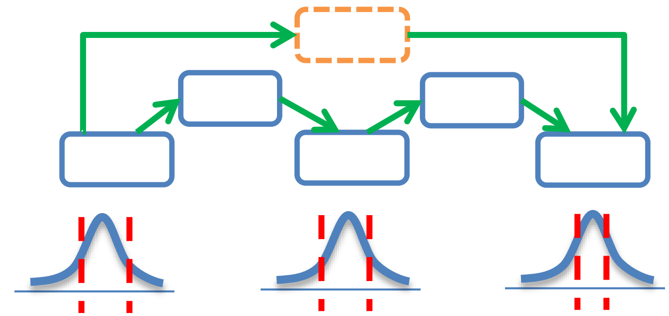
$$Conflict_1 > n_1;$$
$$Conflict_2 > n_2;$$
$$Conflict_3 > n_3;$$
$$\dots \dots$$

Re-allocate Risk
+ Relax Constraints

$$TC: \; LB \downarrow \; or \; UB \uparrow;$$
$$CC: \; Risk \; bound \; \uparrow.$$

Conflict-directed Risk Allocation + Relaxation





Extract Conflict

Re-allocate Risk
+ Relax Constraints

- We start with a very conservative allocation, then iteratively tighten them to reach an agreement between temporal and chance constraints.

Formulation of the Optimization Problem

- We construct a non-linear optimization problem from known conflicts (**Linear** constraints), chance constraint (**non-linear** constraints) and user preferences (**linear** objective).

Temporal conflicts:

$$\Delta_{ShopB} + \Delta_{LunchY} + \Delta_{DriveH} \geq 30;$$
$$\Delta_{DriveB} + \Delta_{DriveBY} + \Delta_{Time} \geq 10;$$
$$\dots \dots$$

Chance constraints:

$$\boldsymbol{Risk}(\boldsymbol{DriveB}) + \boldsymbol{Risk}(\boldsymbol{DriveBY}) + \boldsymbol{Risk}(\boldsymbol{DriveH}) \leq \mathbf{CC};$$
where $Risk(\alpha) = 1 - \int_{LB}^{UB} pdf(Duration(\alpha))$

Objective function:

$$\text{min}imize\left(f(\Delta_{ShopB}) + f(\Delta_{LunchY}) + f(\Delta_{Time}) + \boldsymbol{f}(\boldsymbol{CC})\right).$$

Deep-sea Expeditions

- Uhura as a task scheduling assistant for WHOI scientists.
- http://youtu.be/yuVEUvFZENQ

Robust Trip Planning and Execution of PAV

- PTS with integrated Kirk, Uhura and pSulu for robust trip planning and execution.

- http://youtu.be/cxbYCrd5ho4

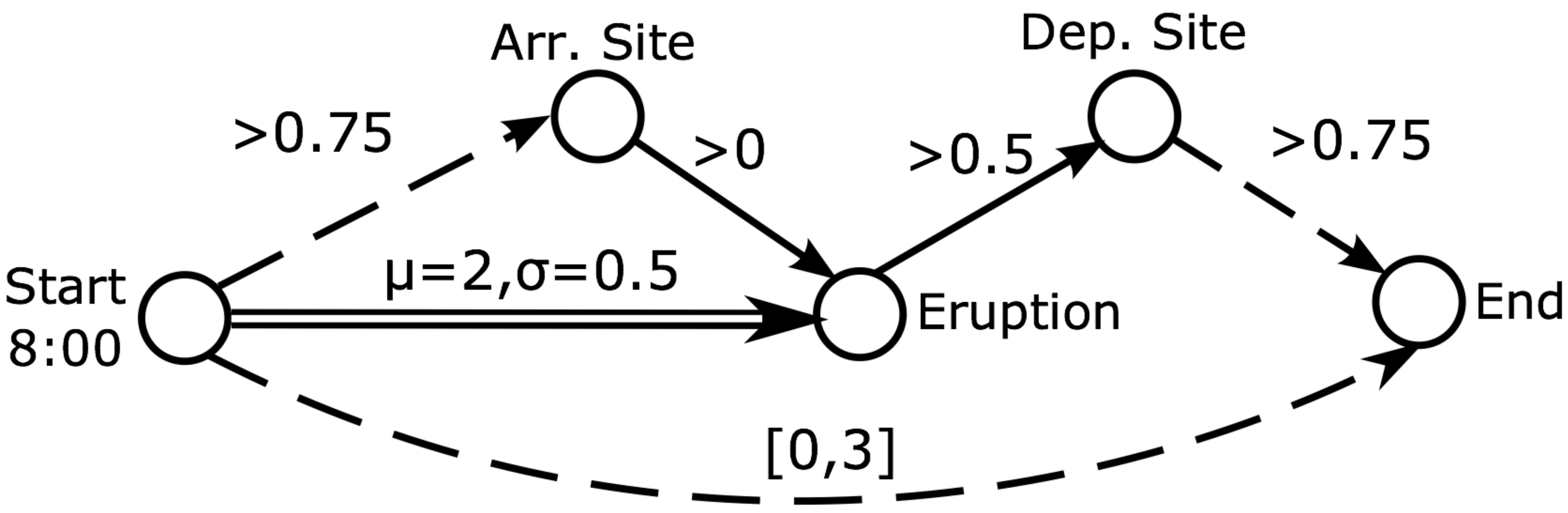
Intelligent Manufacturing Testbed

- Kirk + Uhura + Pike for flexible assembly tasks.
- http://youtu.be/kjaEJvJnbXY

TATA Smart Grid Project

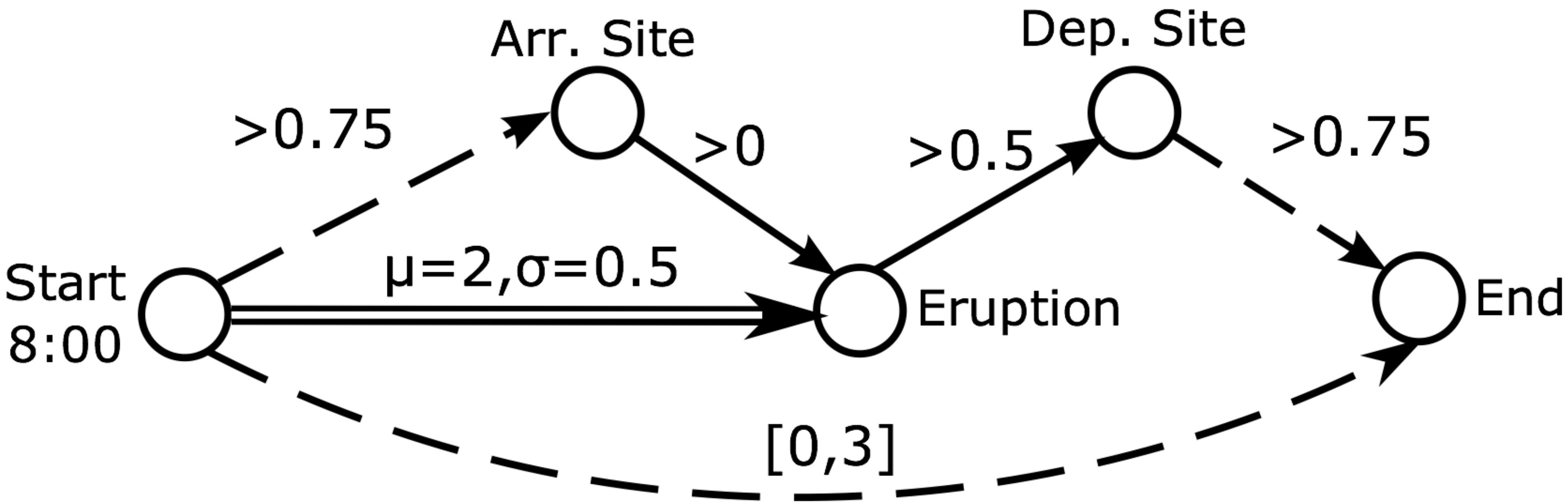
- (New) Household activity advisor for smart grid users.
  - A centralized controller determines how much power to supply to each household, given limited power supply and uncertainty.
  - Uhura communicates with residents of each household and manages their daily activities to meet the demand constraints while maximizing their convenience.

Risk Management in A Deep-sea Mission



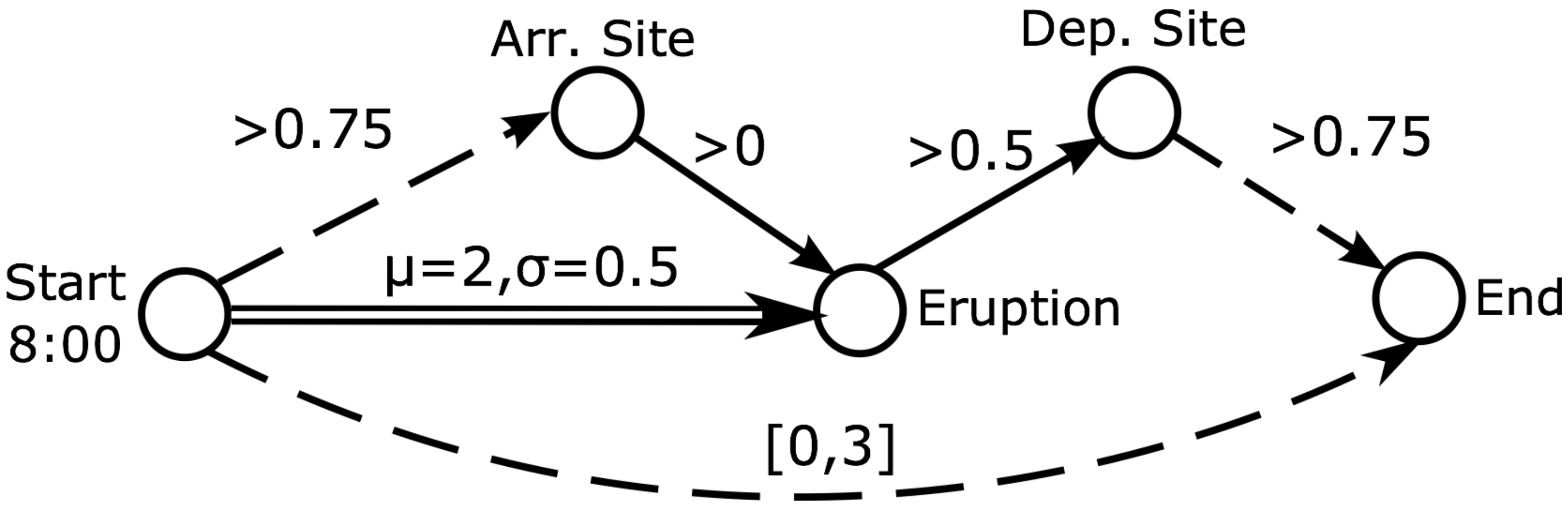
- A scientist (ST) is planning to deploy an under-water robot to survey a volcano eruption on the sea floor.
- After evaluating all the requirements, the Uhura (DA) determines that no solution meets all requirements.

Risk Management in A Deep-sea Mission



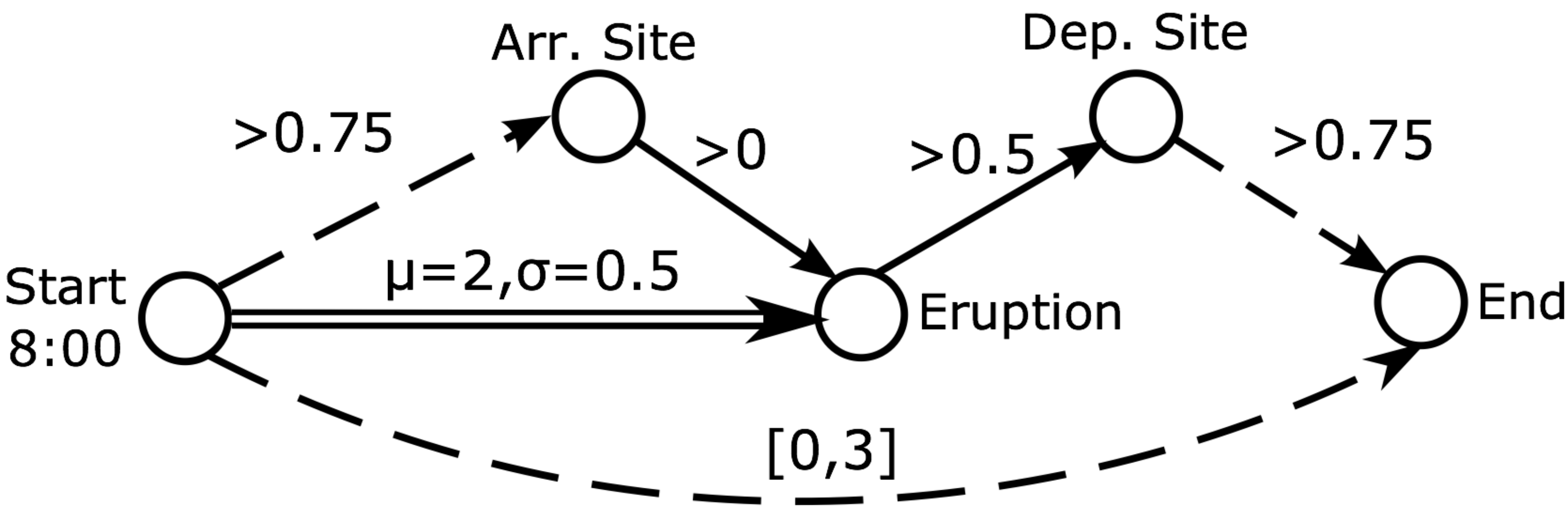
- DA: *I cannot meet all requirements due to the limited mission time and the uncertainty in eruption. Can you extend the mission to 4 hours and 10 minutes.*
- ST: *You can have at most 4 hours for this mission.*

Risk Management in A Deep-sea Mission



- DA: *May I increase the risk bound for this mission to 7.3% in order to meet the duration requirement?*
- ST: *I do not want to take that much risk on this task.*

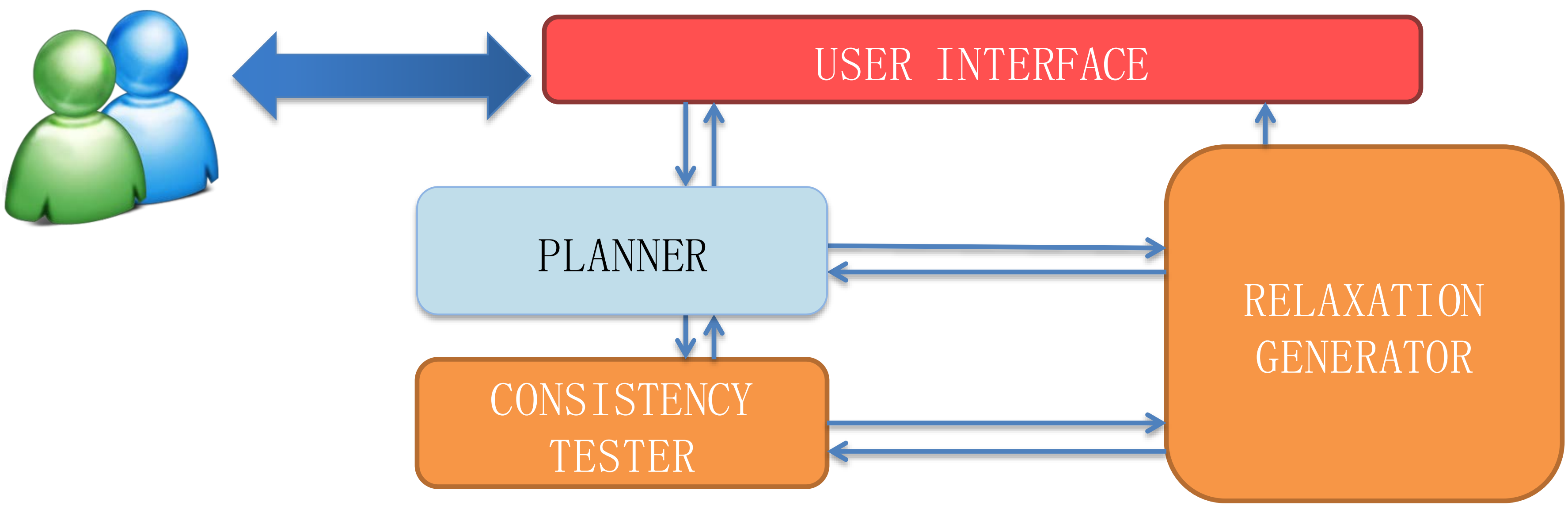
Risk Management in A Deep-sea Mission



- DA: *Ok, can you shorten the traversal time from the site to the ship by 6 minutes? My plan can then cover 95% of the possible eruption time, between 8:45 and 10:51.*
- ST: *That's fine. Thanks.*

Summary for Uhura's Development Plan

- These questions are the keys for making Uhura an effective decision aid on collaborative plan diagnosis.
  - **Intelligent**.
  - **Reliable**.
  - **Conversational**.



References

- [1] Peng Yu and Brian Williams, Continuously Relaxing Over-constrained Conditional Temporal Problems through Generalized Conflict Learning and Resolution, *Proceedings of the Twenty- third International Joint Conference on Artificial Intelligence (IJCAI-13)*, Beijing, 2013.
- [2] Peng Yu and Cheng Fang and Brian Williams, Resolving Uncontrollable Conditional Temporal Problems using Continuous Relaxations, *Proceedings of the Twenty-fourth International Conference on Automated Planning and Scheduling (ICAPS-14)*, Portsmouth, 2014 (**Honorable Mention for Best Paper Award**).
- [3] Cheng Fang and Peng Yu and Brian Williams, Chance-constrained Probabilistic Simple Temporal Problems, *In Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence (AAAI-14)*, Quebec City, 2014.
- [4] Peng Yu and Cheng Fang and Brian Williams, Resolving Over-constrained Probabilistic Temporal Problems through Chance Constraint Relaxation, **submitted to** *the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15)*, Austin, 2015.

References

- [5] Amanda Coles, Andrew Coles, Maria Fox, and Derek Long. Colin: Planning with continuous linear numeric change. *Journal of Artificial Intelligence Research (JAIR)*, 44:1–96, 2012.
- [6] I-hsiang Shu, Robert Effinger, and Brian C. Williams. Enabling fast flexible planning through incremental temporal reasoning with conflict extraction. In *Proceedings of the 15th International Conference on Automated Planning and Scheduling (ICAPS-05)*, pages 252– 261, 2005.
- [7] Richard Socher, Jeffrey Pennington, Eric H. Huang, Andrew Y. Ng, and Christopher D. Manning. Semi-supervised recursive autoencoders for predicting sentiment distributions. *In Proceedings of the Conference on Empirical Methods on Natural Language Processing*, 2011.
- [8] Traum David, Approaches to Dialogue Systems and Dialogue Management - Lecture Notes and Bibliography, *http://people.ict.usc.edu/~traum/ESSLLI08/,* accessed Sep 23, 2014.