Acting is the Purpose of Planning

or

The Actor's view of Deliberation

Malik Ghallab

LAAS-CNRS

Rome, June 2013

A change of focus

A change of focus

- Not a novel research path

A change of focus

- Not a novel research path
- Focus Automated Planning research on the ***actor's perspective***

A change of focus

- Not a novel research path
- Focus Automated Planning research on the ***actor's perspective***

- Ideas in this presentation stem from

A change of focus

- Not a novel research path
- Focus Automated Planning research on the ***actor's perspective***

- Ideas in this presentation stem from
  - Collaboration with Felix Ingrand

    A Survey on Deliberation Functions for Autonomous Robots

A change of focus

- Not a novel research path
- Focus Automated Planning research on the ***actor's perspective***

- Ideas in this presentation stem from
  - Collaboration with Felix Ingrand

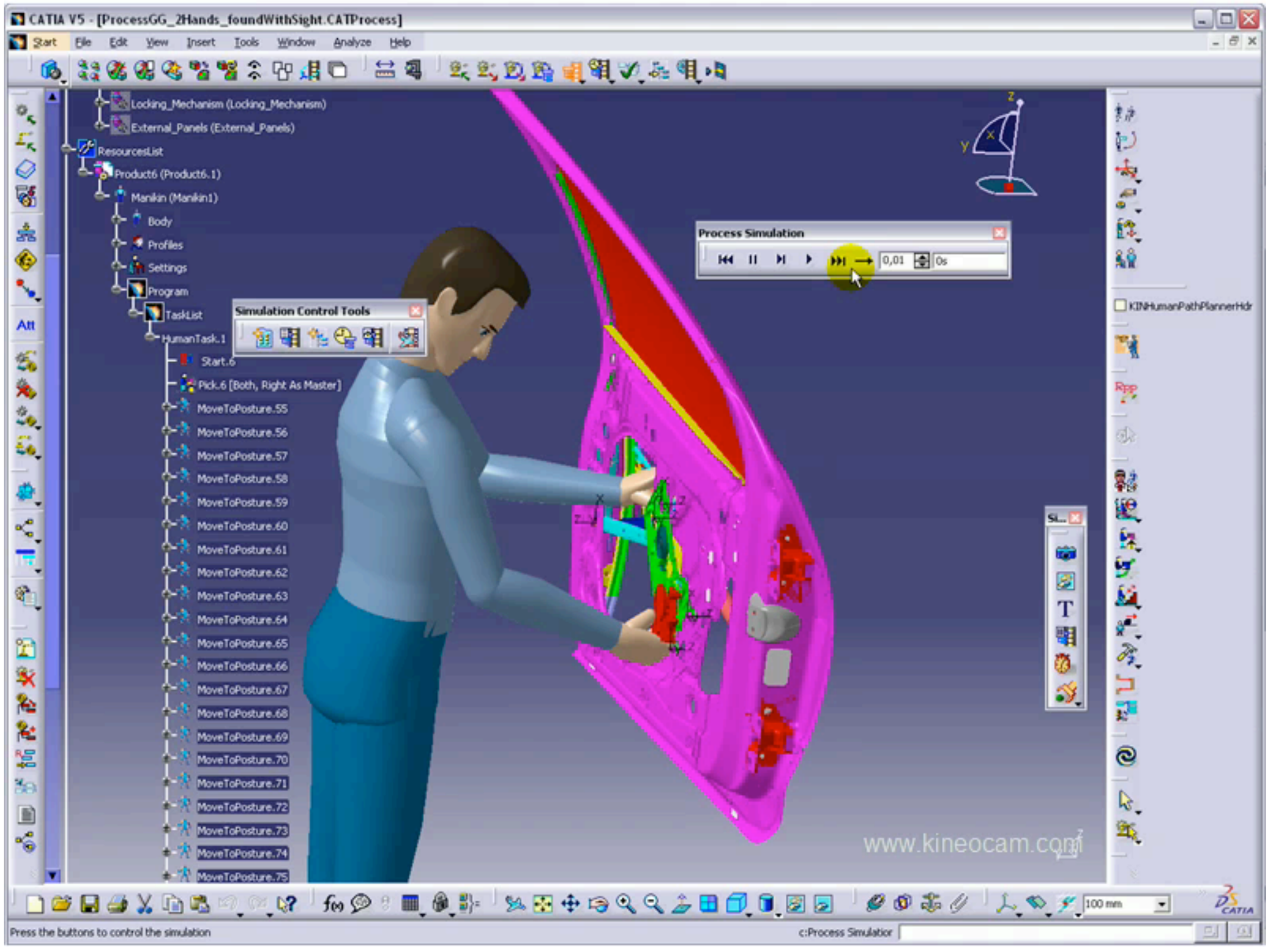
    A Survey on Deliberation Functions for Autonomous Robots
  - Collaboration with Dana Nau and Paolo Traverso

    A Position Paper on Automated Planning and Acting

Premises

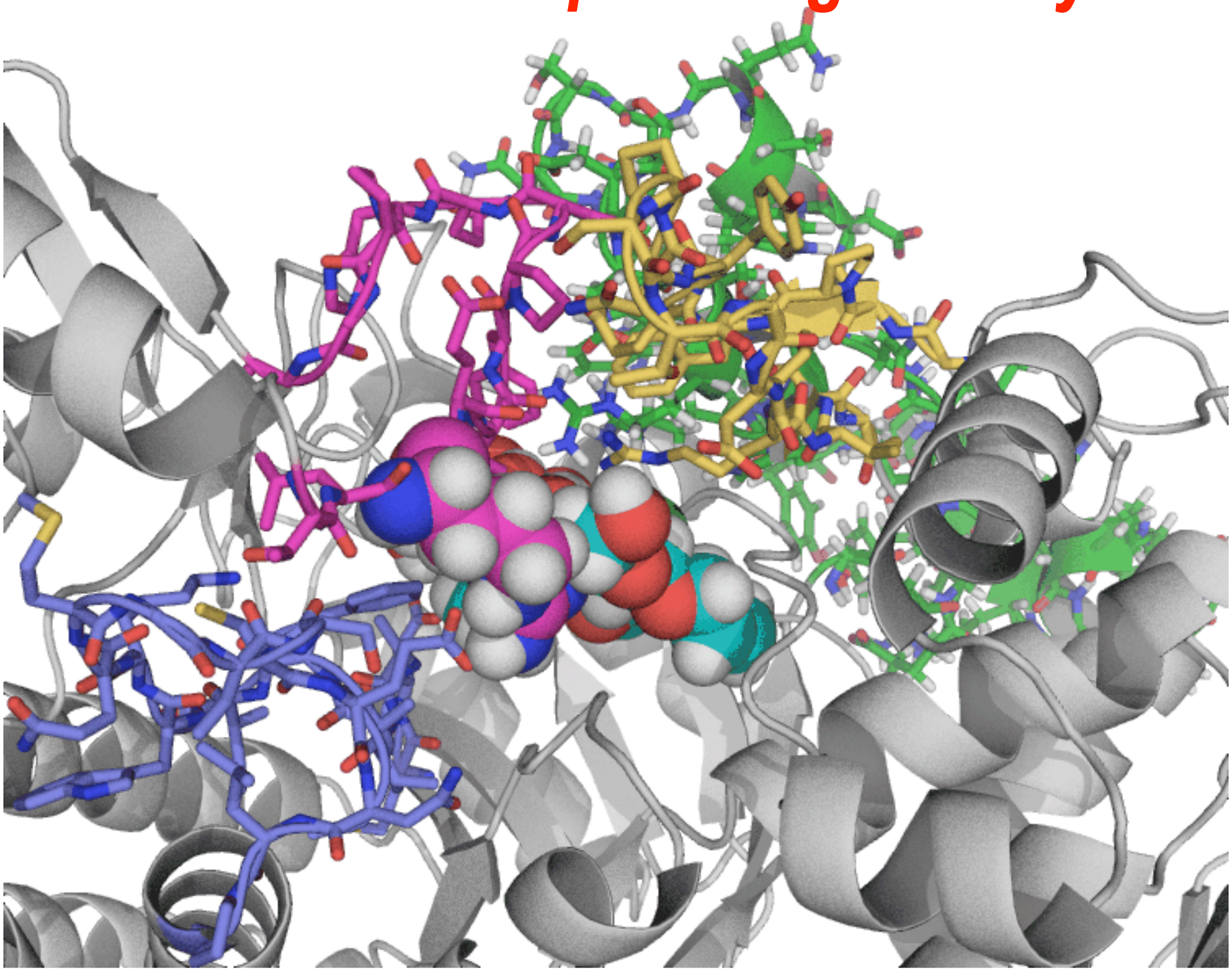
- Planning is a deliberation function
- Main purpose of planning is acting
  - Planning is valuable for other uses than acting
    - *Design* of tools, assembly, molecules permitting actions
    - Video games

Motion planning in car design



[Kineo]

Motion planning in enzyme design



[J.Cortes, LAAS]

Premises

- Planning is a deliberation function
- Main purpose of planning is acting
  - Planning is valuable for other uses

Premises

- Planning is a deliberation function
- Main purpose of planning is acting
  - Planning is valuable for other uses
  - Planning is always motivated by acting

Premises

- Planning is a deliberation function
- Main purpose of planning is acting
  - Planning is valuable for other uses
  - Planning is always motivated by acting
- Acting requires other deliberation functions

Deliberate action:

- Intentionally chosen and planned to achieve some objective
- Pursued for the accomplishment of the robot's task

Premises

- Planning is a deliberation function
- Main purpose of planning is acting
  - Planning is valuable for other uses
  - Planning is always motivated by acting
- Acting requires other deliberation functions

  Deliberate action:
  - Intentionally chosen and planned to achieve some objective
  - Pursued for the accomplishment of the robot's task
- Automated Planning research *does not*

  focus on actual challenges of deliberate action

Critical view of current main stream

Critical view of current main stream

- Automated Planning literature

Critical view of current main stream

- Automated Planning literature
  - Numerous "*Planners*"

Critical view of current main stream

- Automated Planning literature
  - Numerous "*Planners*"
  - Very few "*Actors*"

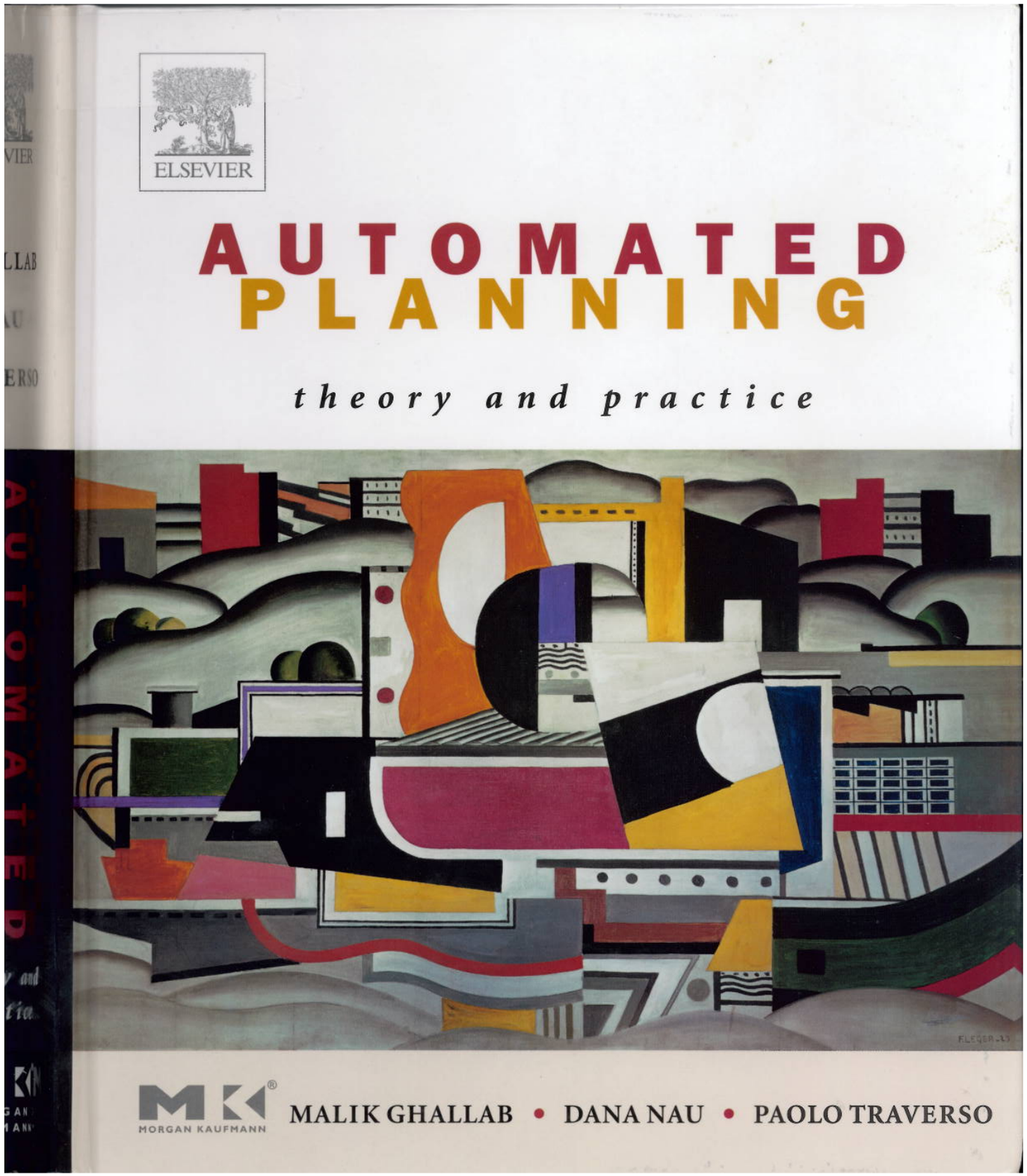
Critical view of current main stream

- Automated Planning literature
  - Numerous "*Planners*"
  - Very few "*Actors*"
  - Search problems: over-explored

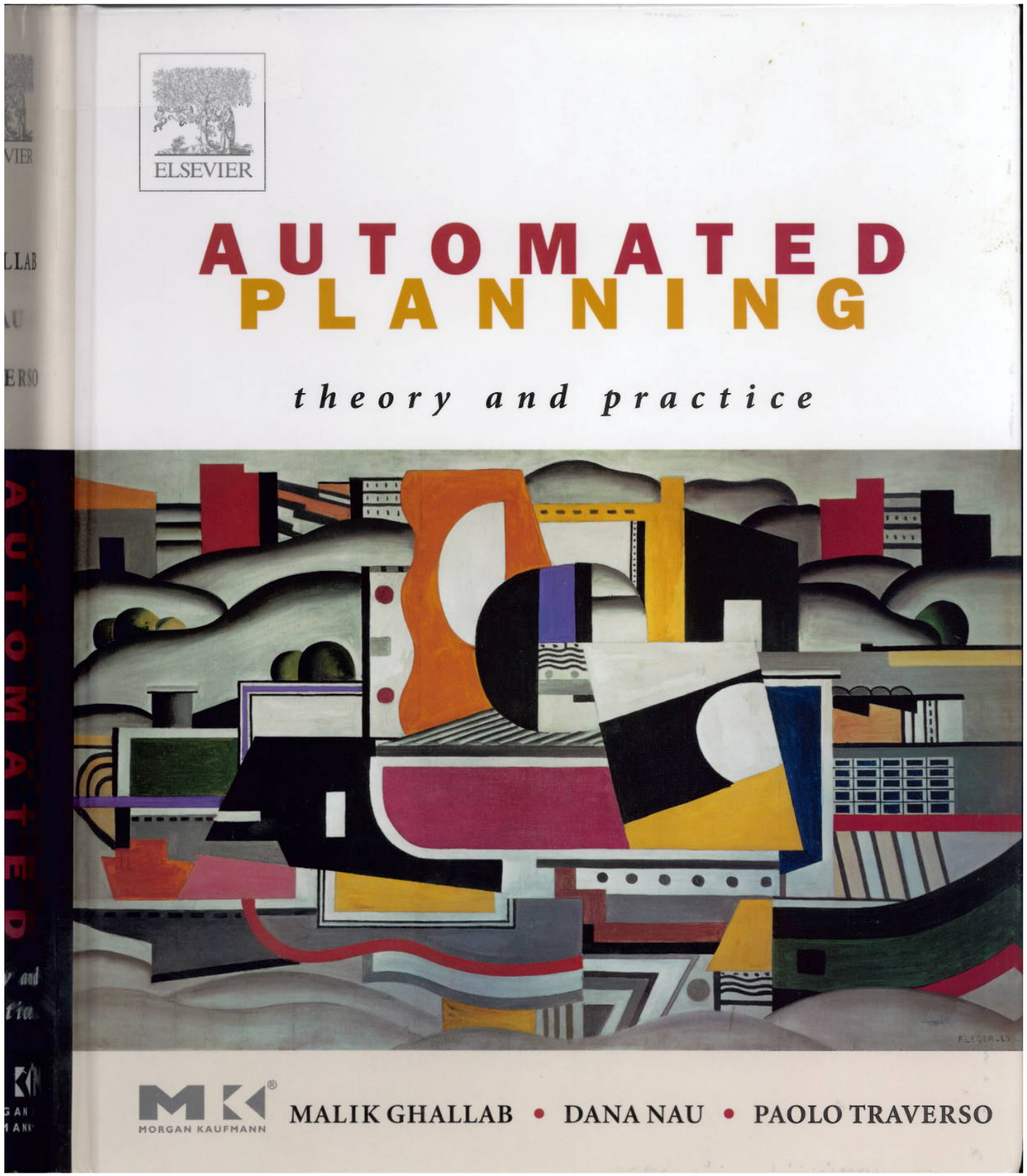
Critical view of current main stream

- Automated Planning literature
  - Numerous "*Planners*"
  - Very few "*Actors*"
  - Search problems: over-explored
  - Acting problems: insufficiently investigated

A typical textbook of the field

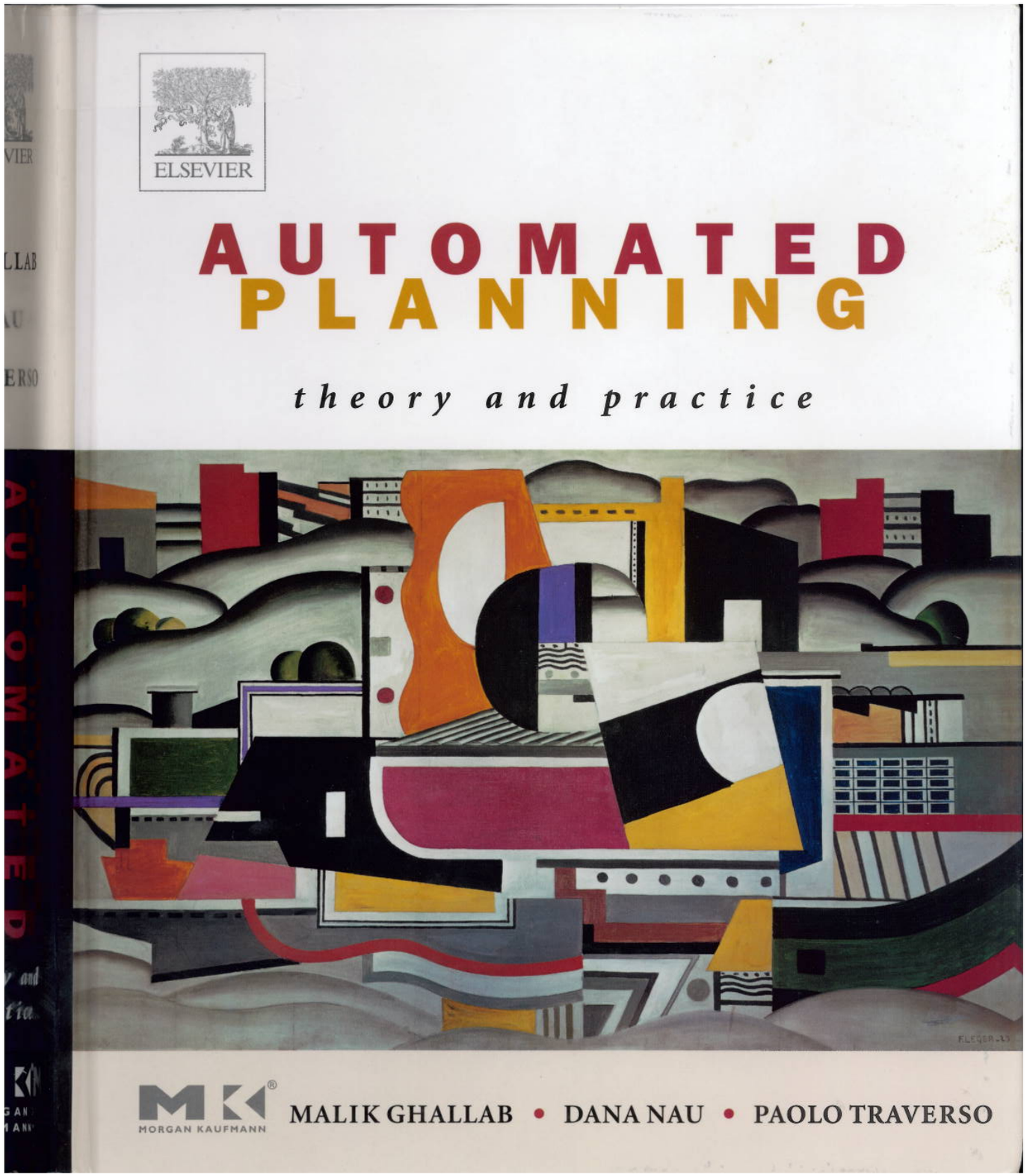


A typical textbook of the field



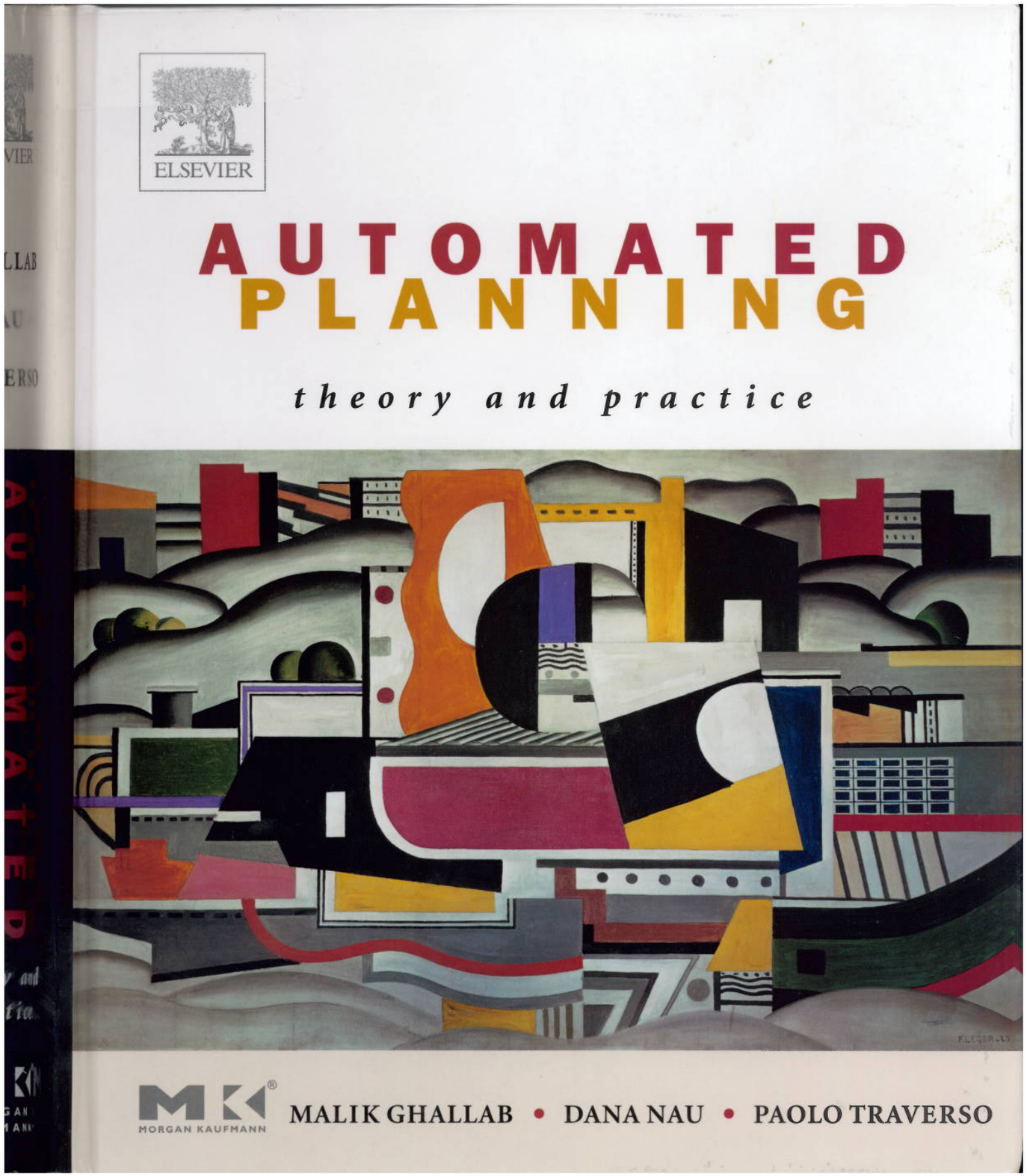
- Planing and Acting:

A typical textbook of the field



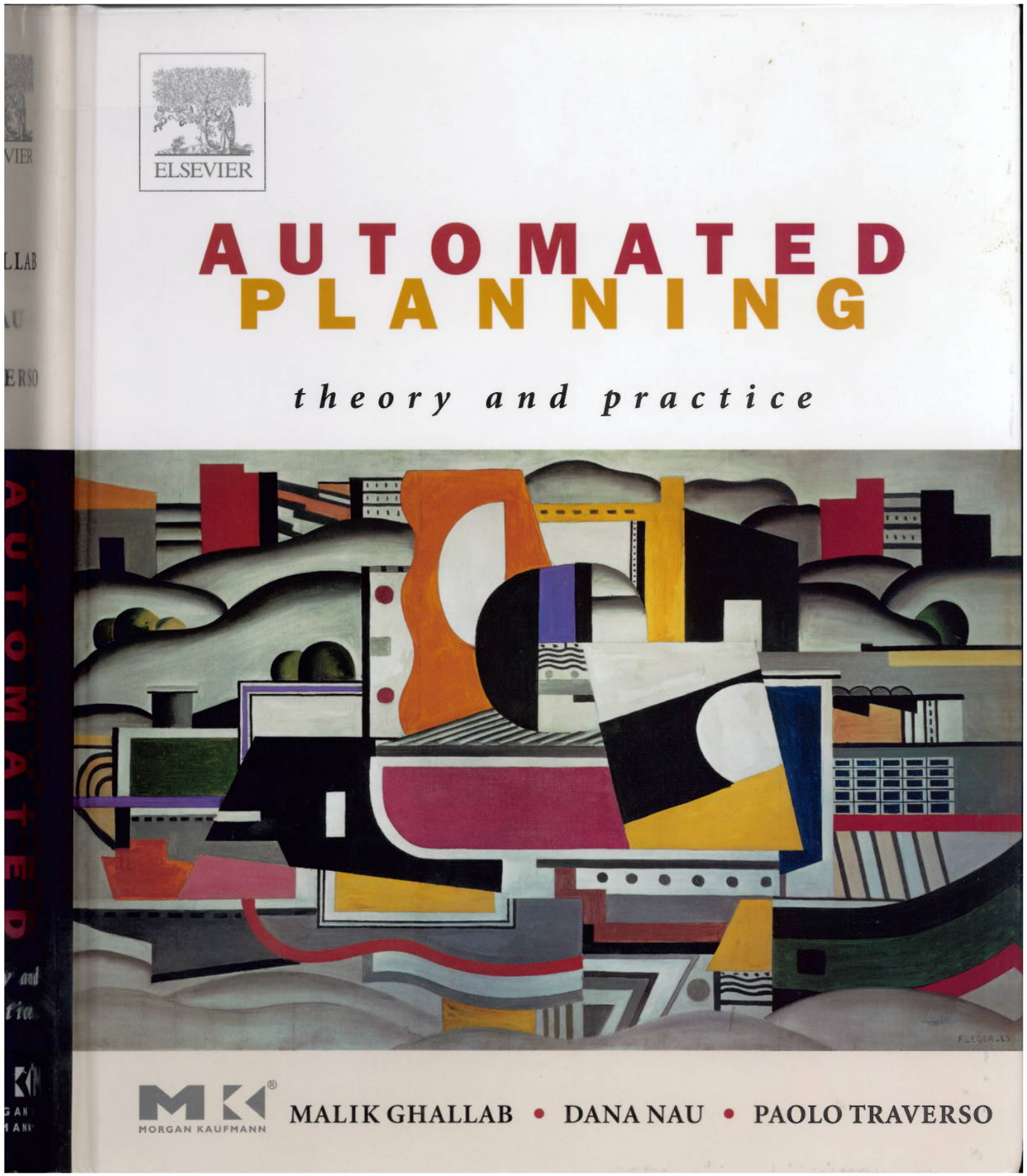
- Planing and Acting:
  - Section 24.8

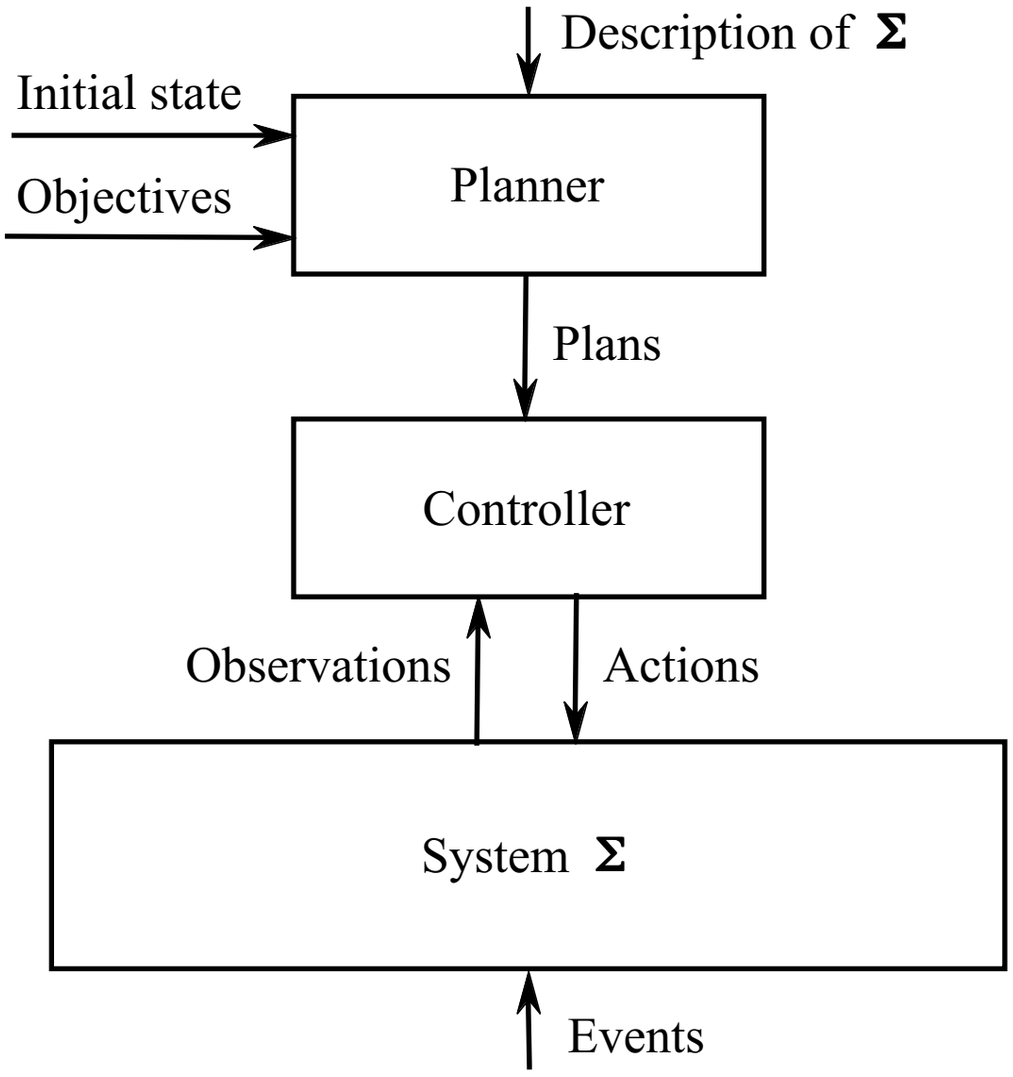
A typical textbook of the field



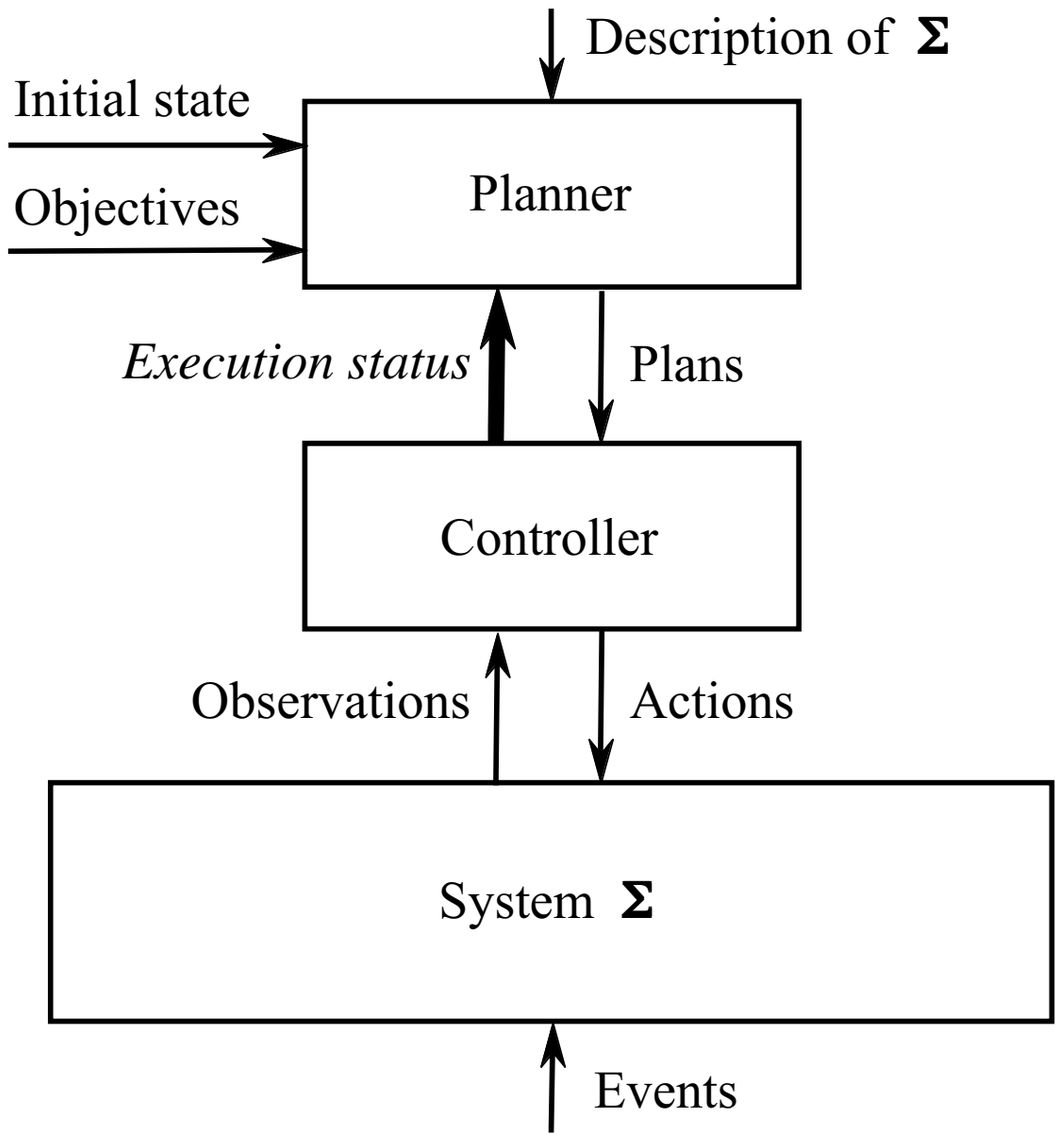
- Planing and Acting:
  - Section 24.8
  - 1.2 page / 630p.

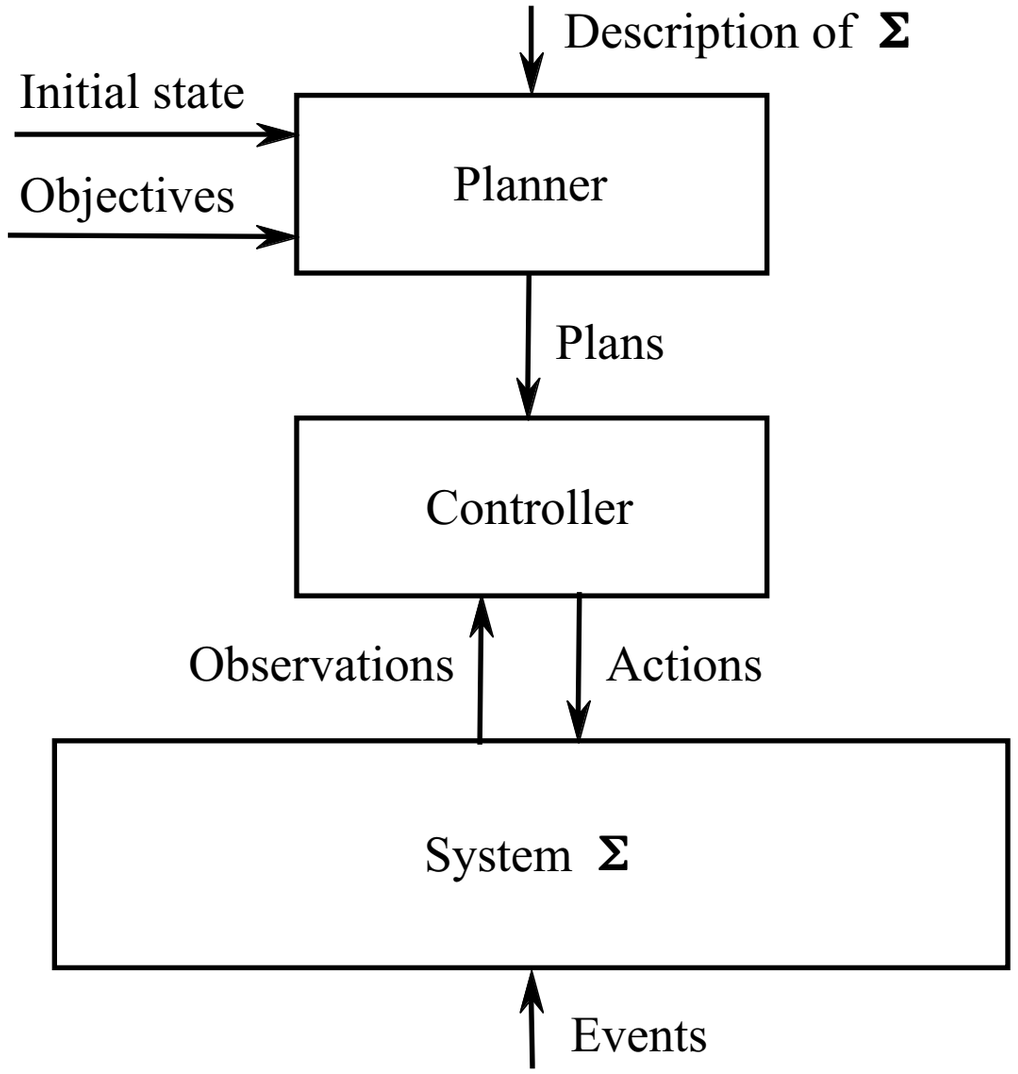
A typical textbook of the field





A typical textbook of the field





Critical view of current main stream

- Automated Planning literature

Critical view of current main stream

- Automated Planning literature
- Task planning in robotics: *not the main deliberation bottleneck*

Critical view of current main stream

- Automated Planning literature

- Task planning in robotics: *not the main deliberation bottleneck*
  - "*Precond-Effects*" actions: too abstract
    too far from sensory-motor commands

Critical view of current main stream

- Automated Planning literature

- Task planning in robotics: *not the main deliberation bottleneck*
  - "*Precond-Effects*" actions: too abstract
    too far from sensory-motor commands
  - In robot platforms "*Precond-Effects*" actions

Critical view of current main stream

- Automated Planning literature

- Task planning in robotics: *not the main deliberation bottleneck*
  - "*Precond-Effects*" actions: too abstract
    too far from sensory-motor commands
  - In robot platforms "*Precond-Effects*" actions
    - Not available as plug-&-trigger executable primitives

Critical view of current main stream

- Automated Planning literature

- Task planning in robotics: *not the main deliberation bottleneck*
  - "*Precond-Effects*" actions: too abstract
    too far from sensory-motor commands
  - In robot platforms "*Precond-Effects*" actions
    - Not available as plug-&-trigger executable primitives
    - Will they be available in future robots ?

Critical view of current main stream

- Automated Planning literature

- Task planning in robotics: *not the main deliberation bottleneck*
  - "*Precond-Effects*" actions: too abstract
    too far from sensory-motor commands
  - In robot platforms "*Precond-Effects*" actions
    - Not available as plug-&-trigger executable primitives
    - Will they be available in future robots ?
  - Robot manufacturers focus on

Critical view of current main stream

- Automated Planning literature

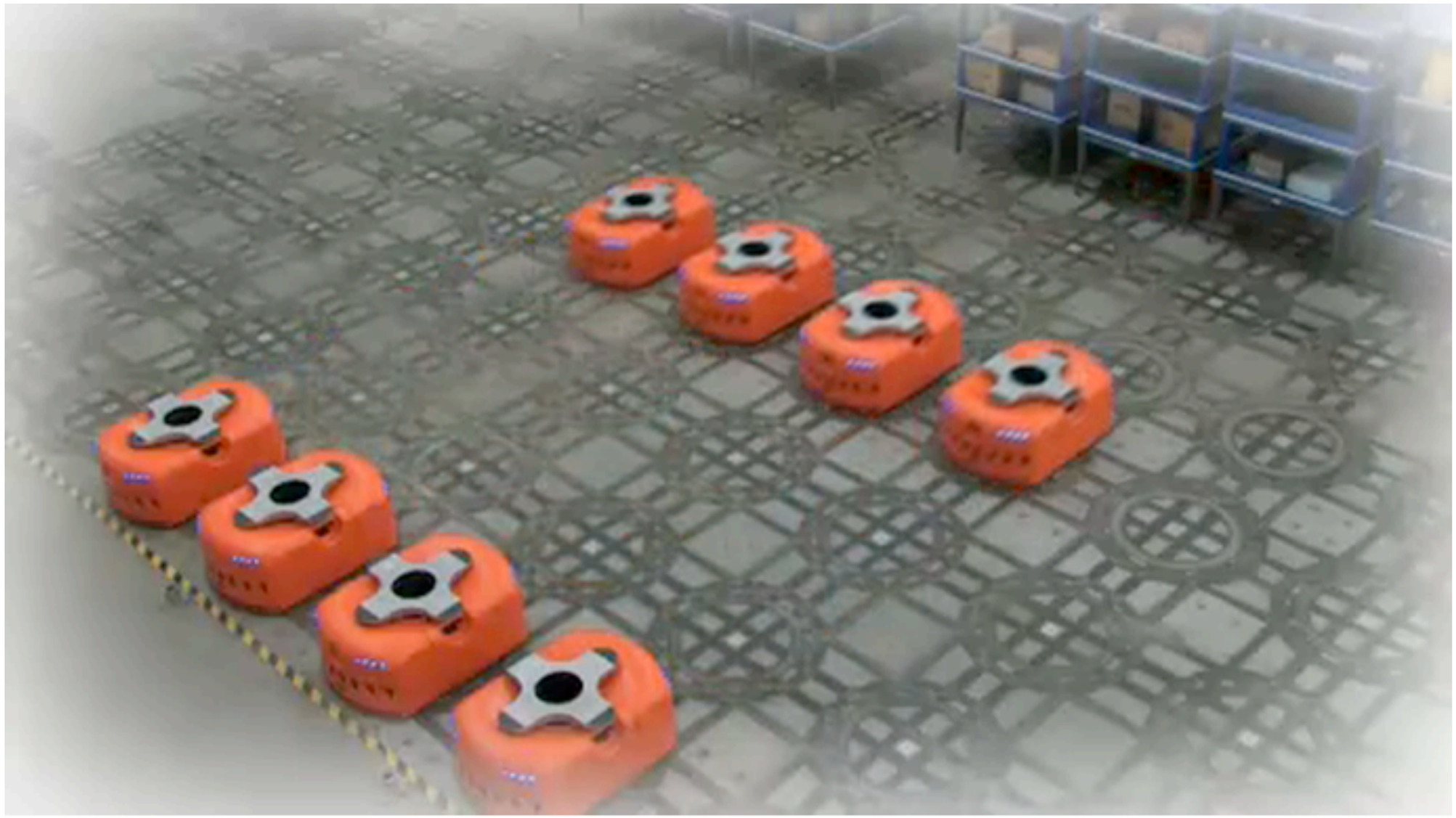
- Task planning in robotics: *not the main deliberation bottleneck*
  - "*Precond-Effects*" actions: too abstract
    too far from sensory-motor commands
  - In robot platforms "*Precond-Effects*" actions
    - Not available as plug-&-trigger executable primitives
    - Will they be available in future robots ?
  - Robot manufacturers focus on
    - Programmable systems and ease of programming

Critical view of current main stream

- Automated Planning literature

- Task planning in robotics: *not the main deliberation bottleneck*
  - "*Precond-Effects*" actions: too abstract
    too far from sensory-motor commands
  - In robot platforms "*Precond-Effects*" actions
    - Not available as plug-&-trigger executable primitives
    - Will they be available in future robots ?
  - Robot manufacturers focus on
    - Programmable systems and ease of programming
    - Not much on autonomous deliberation through planning

Example of Warehouse automation



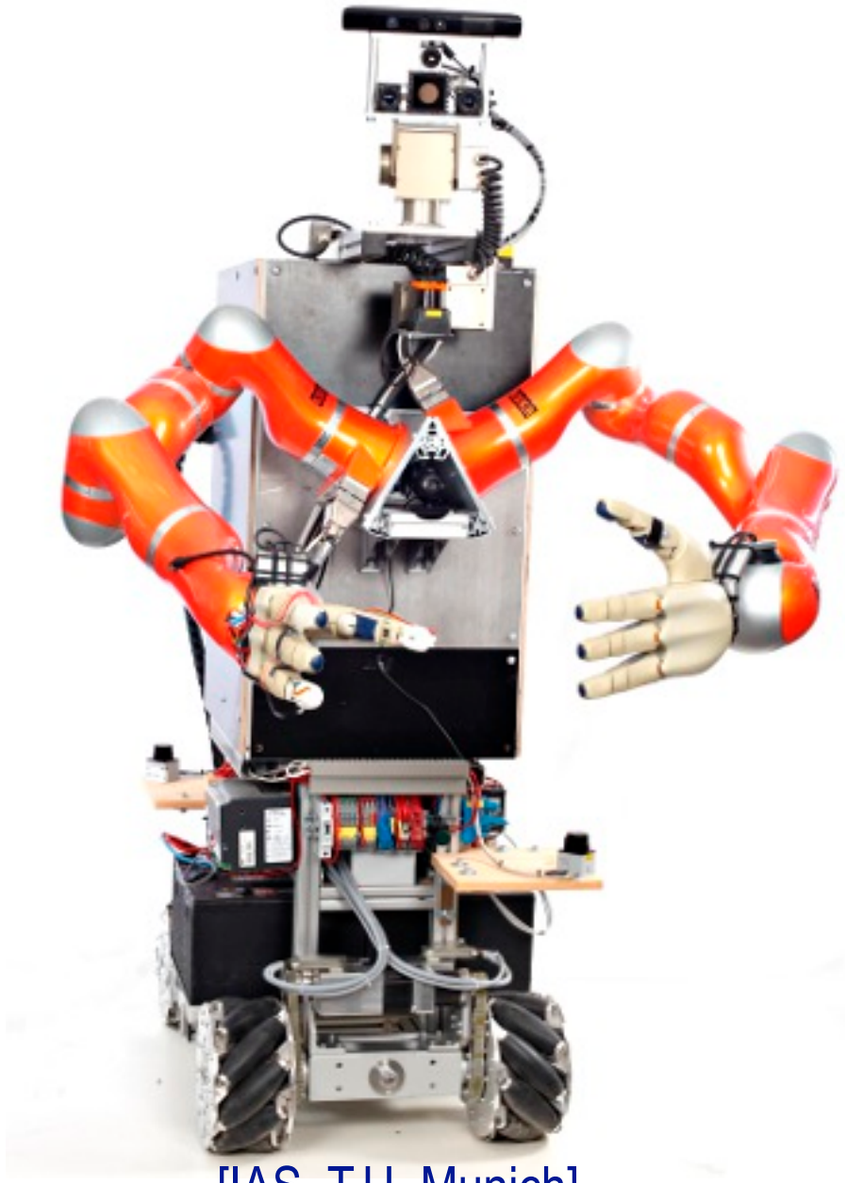
Kiva Systems

Household robot factotum

[LAAS, Toulouse]

Household robot factotum

[LAAS, Toulouse]

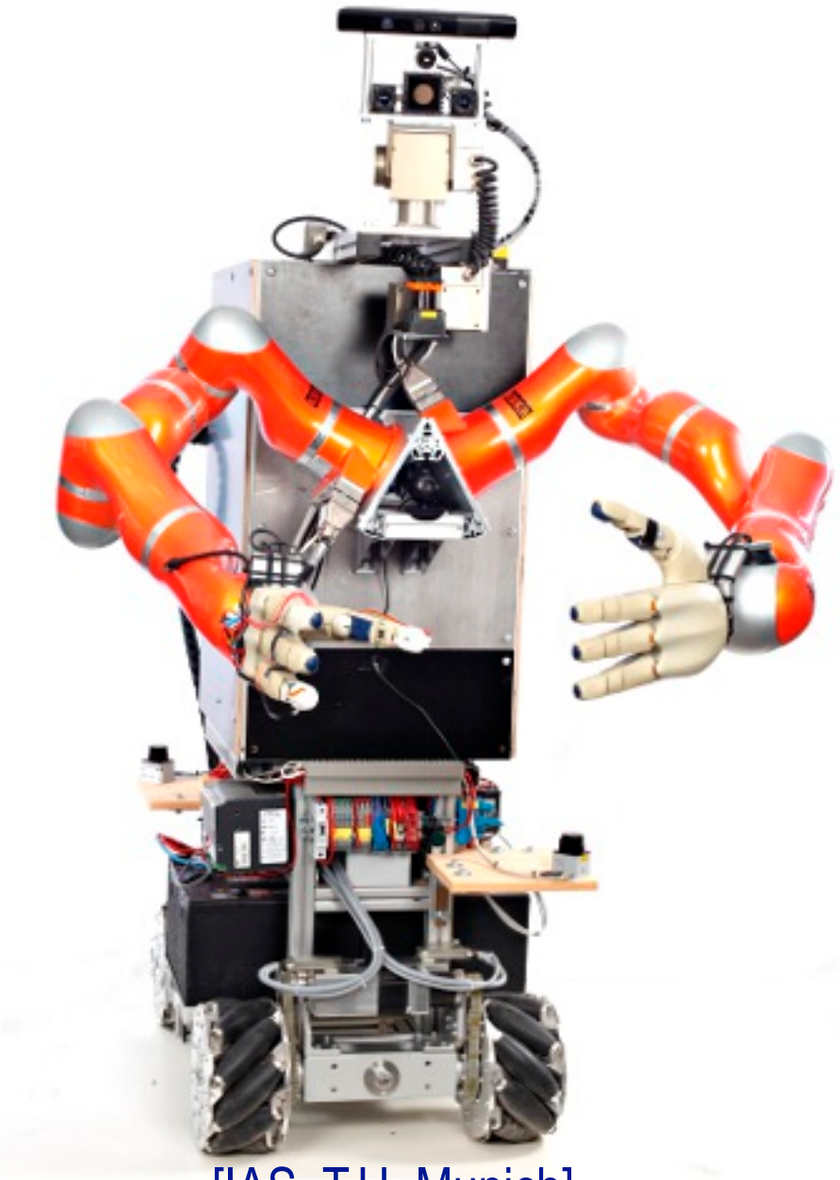


[IAS, T.U. Munich]

Household robot factotum

- Mission
  - Household services: pick, bring, maintain, sort, clean, cook
  - Support users: help, remind, monitor
- Main features
  - Wide *variety* of environments and tasks
  - Complex interactions with users

[LAAS, Toulouse]



[IAS, T.U. Munich]

Basic skills for a factotum robot

- ***Localize*** itself in the environment map
- ***Map*** the environment: extend or update the map
- ***Move*** to a target position
- ***Detect*** and ***avoid*** obstacle
- ***Identify*** and ***locate*** an item
- ***Grasp*** and ***ungrasp*** an item
- ***Push*** an item

Basic skills for a factotum robot

- ***Localize*** itself in the environment map
- ***Map*** the environment: extend or update the map
- ***Move*** to a target position
- ***Detect*** and ***avoid*** obstacle
- ***Identify*** and ***locate*** an item
- ***Grasp*** and ***ungrasp*** an item
- ***Push*** an item

Move: {environment features} ⟶ {navigation strategies}

- plan a path
- localize
- detect and avoid
- update map

Basic skills for a factotum robot

- ***Localize*** itself in the environment map
- ***Map*** the environment: extend or update the map
- ***Move*** to a target position
- ***Detect*** and ***avoid*** obstacle
- ***Identify*** and ***locate*** an item
- ***Grasp*** and ***ungrasp*** an item
- ***Push*** an item

Move: {environment features} ⟶ {navigation strategies}

- plan a path
- localize
- detect and avoid
- update map

Mapping: dynamic & context-dependent

Basic skills for a factotum robot

- ***Localize*** itself in the environment map
- ***Map*** the environment: extend or update the map
- ***Move*** to a target position
- ***Detect*** and ***avoid*** obstacle
- ***Identify*** and ***locate*** an item
- ***Grasp*** and ***ungrasp*** an item
- ***Push*** an item

Can these skills be circumscribed as "*Precond-Effects*" primitives?

Basic skills for a factotum robot

- ***Localize*** itself in the environment map
- ***Map*** the environment: extend or update the map
- ***Move*** to a target position
- ***Detect*** and ***avoid*** obstacle
- ***Identify*** and ***locate*** an item
- ***Grasp*** and ***ungrasp*** an item
- ***Push*** an item

Can these skills be circumscribed as "*Precond-Effects*" primitives?

```
(:action move
    :parameters (?r - robot ?from ?to - location)
    :precondition (and (adjacent ?from ?to)
            (at ?r ?from) (not (occupied ?to)))
    :effect (and (at ?r ?to) (not (occupied ?from))
            (occupied ?to) (not (at ?r ?from)) ))
```

Basic skills for a factotum robot

- ***Localize*** itself in the environment map
- ***Map*** the environment: extend or update the map
- ***Move*** to a target position
- ***Detect*** and ***avoid*** obstacle
- ***Identify*** and ***locate*** an item
- ***Grasp*** and ***ungrasp*** an item
- ***Push*** an item

Can these skills be circumscribed as "*Precond-Effects*" primitives?

$$\mathsf{move}(r, l, l')@[t_s, t_e)$$

$$\begin{aligned}
\text{precond:}\quad & \mathsf{at}(r, l)@[t_1, t_s),\ \mathsf{free}(l')@[t_2, t_e) \\
\text{effects:}\quad & \mathsf{at}(r,\ \mathsf{routes})@[t_s, t_e) \\
& \mathsf{at}(r, l')@[t_e, t_3) \\
& \mathsf{free}(l)@[t_4, t_5) \\
\text{const:}\quad & t_s < t_4 < t_2,\ \mathsf{adjacent}(l, l')
\end{aligned}$$

Basic skills for a factotum robot

- ***Localize*** itself in the environment map
- ***Map*** the environment: extend or update the map
- ***Move*** to a target position
- ***Detect*** and ***avoid*** obstacle
- ***Identify*** and ***locate*** an item
- ***Grasp*** and ***ungrasp*** an item
- ***Push*** an item

Can these skills be circumscribed as "*Precond-Effects*" primitives?

Not without much refinement!

Basic skills for a factotum robot

- ***Localize*** itself in the environment map
- ***Map*** the environment: extend or update the map
- ***Move*** to a target position
- ***Detect*** and ***avoid*** obstacle
- ***Identify*** and ***locate*** an item
- ***Grasp*** and ***ungrasp*** an item
- ***Push*** an item

Can these skills be circumscribed as "*Precond-Effects*" primitives?

Not without much refinement!

These skills require significant deliberation in various forms

Basic skills for a factotum robot

- ***Localize*** itself in the environment map
- ***Map*** the environment: extend or update the map
- ***Move*** to a target position
- ***Detect*** and ***avoid*** obstacle
- ***Identify*** and ***locate*** an item
- ***Grasp*** and ***ungrasp*** an item
- ***Push*** an item

Can these skills be circumscribed as "*Precond-Effects*" primitives?

Not without much refinement!

These skills require significant deliberation in various forms

→ *the actor's view of deliberation*

Basic skills for a factotum robot

- ***Localize*** itself in the environment map
- ***Map*** the environment: extend or update the map
- ***Move*** to a target position
- ***Detect*** and ***avoid*** obstacle
- ***Identify*** and ***locate*** an item
- ***Grasp*** and ***ungrasp*** an item
- ***Push*** an item

Can these skills be circumscribed as "*Precond-Effects*" primitives?

Not without much refinement!

These skills require significant deliberation in various forms

→ *the actor's view of deliberation*

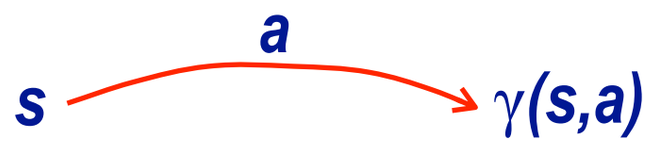
not enough addressed in planning

Why actor's view not enough addressed in planning

- Emphasis bias issue
  - Planning: Choice and organization of actions
  - Ingredients: Prediction + Search
  - *Precond-Effects* models
    - Focus on search
    - Trivialized prediction

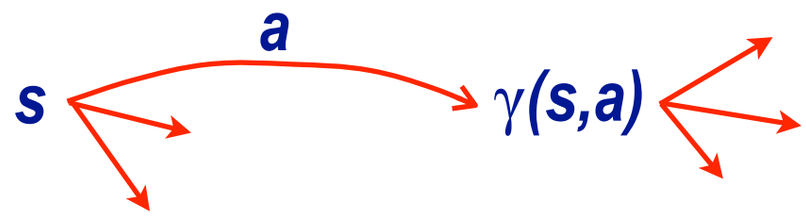
Why actor's view not enough addressed in planning

- Emphasis bias issue
  - Planning: Choice and organization of actions
  - Ingredients: Prediction + Search
  - *Precond-Effects* models
    - Focus on search
    - Trivialized prediction



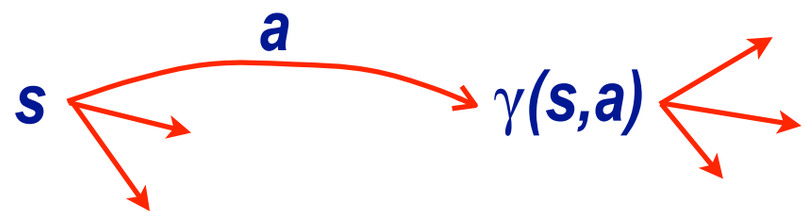
Why actor's view not enough addressed in planning

- Emphasis bias issue
  - Planning: Choice and organization of actions
  - Ingredients: Prediction + Search
  - *Precond-Effects* models
    - Focus on search
    - Trivialized prediction



Why actor's view not enough addressed in planning

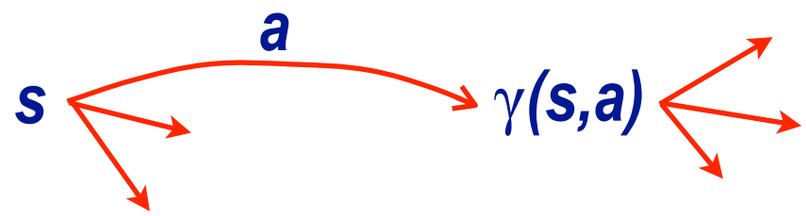
- Emphasis bias issue
  - Planning: Choice and organization of actions
  - Ingredients: Prediction + Search
  - *Precond-Effects* models
    - Focus on search
    - Trivialized prediction



- Formalization issue
  - Planning nicely formalized as mapping Domains → Plans
  - Acting difficult to formalize

Why actor's view not enough addressed in planning

- Emphasis bias issue
  - Planning: Choice and organization of actions
  - Ingredients: Prediction + Search
  - *Precond-Effects* models
    - Focus on search
    - Trivialized prediction



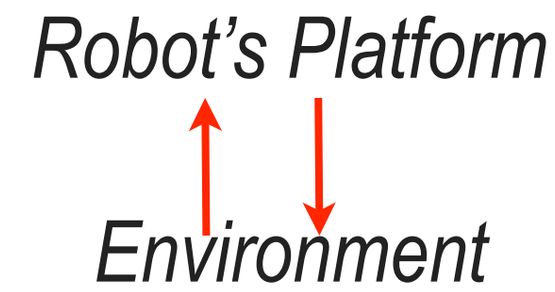
- Formalization issue
  - Planning nicely formalized as mapping Domains → Plans
  - Acting difficult to formalize
- Frontier issue
  - Planning easily separated from sensory-motor functions
  - "*Acting" vs "Executing"* more blurred frontier

Actor's view of deliberation

Actor's view of deliberation

Robot's Platform

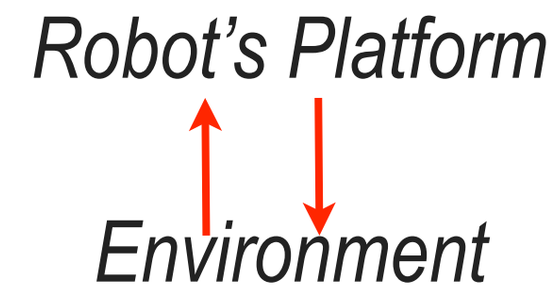
Actor's view of deliberation



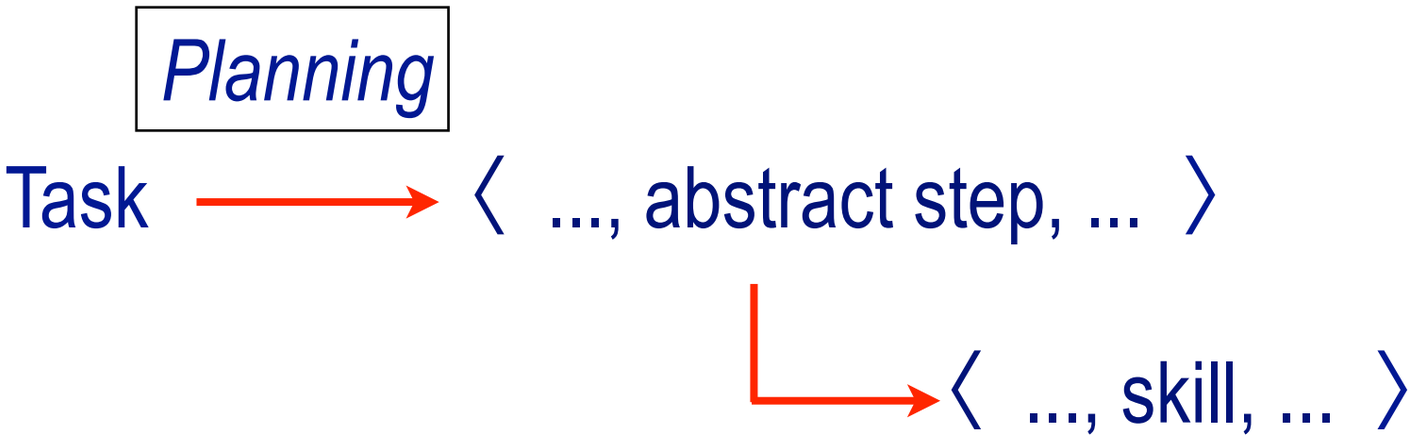
Actor's view of deliberation

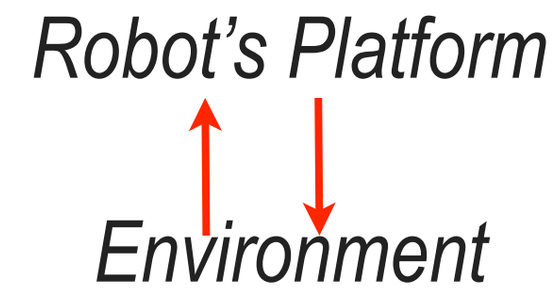
Planning

Task ⟶ ⟨ ..., abstract step, ... ⟩

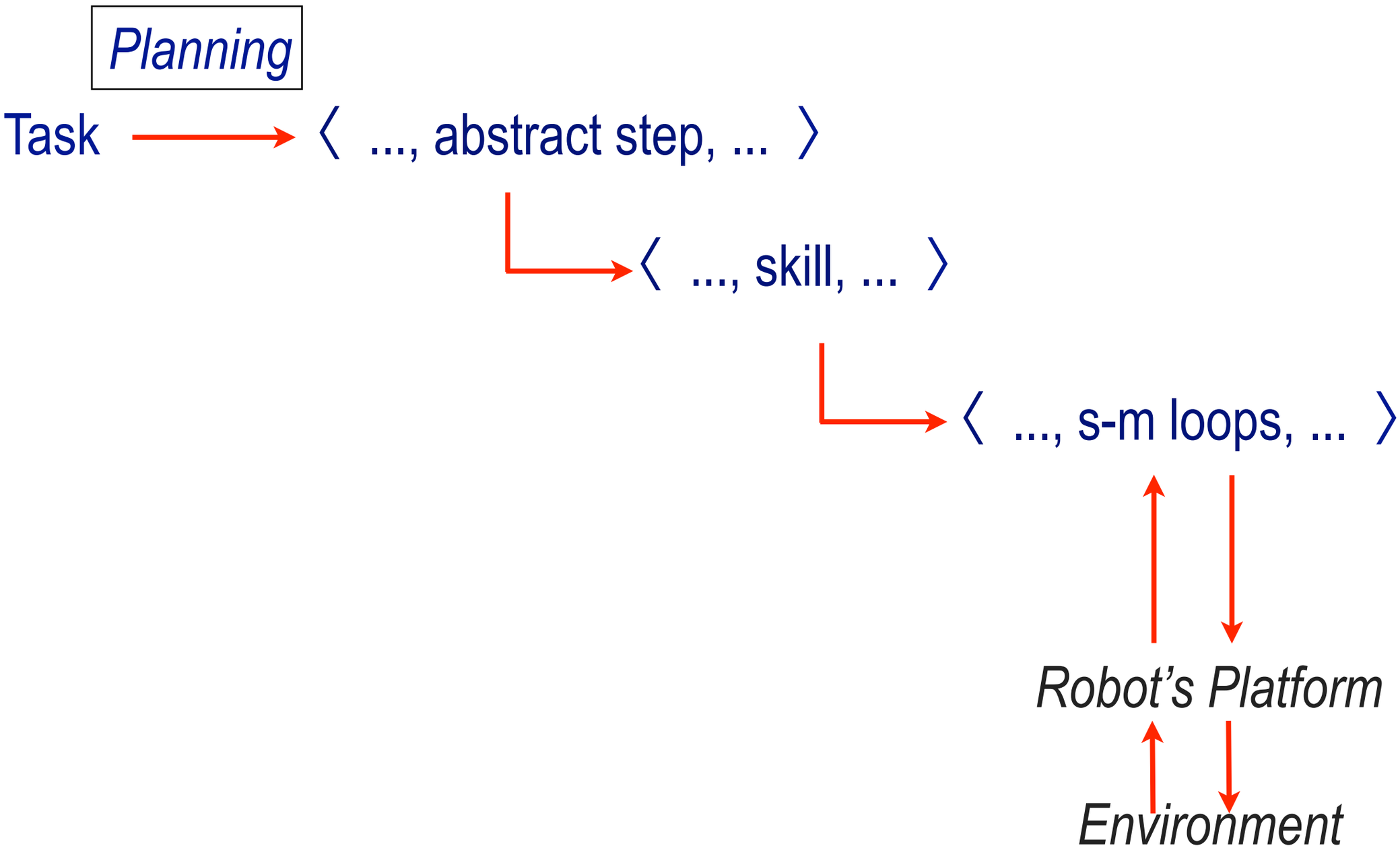


Actor's view of deliberation

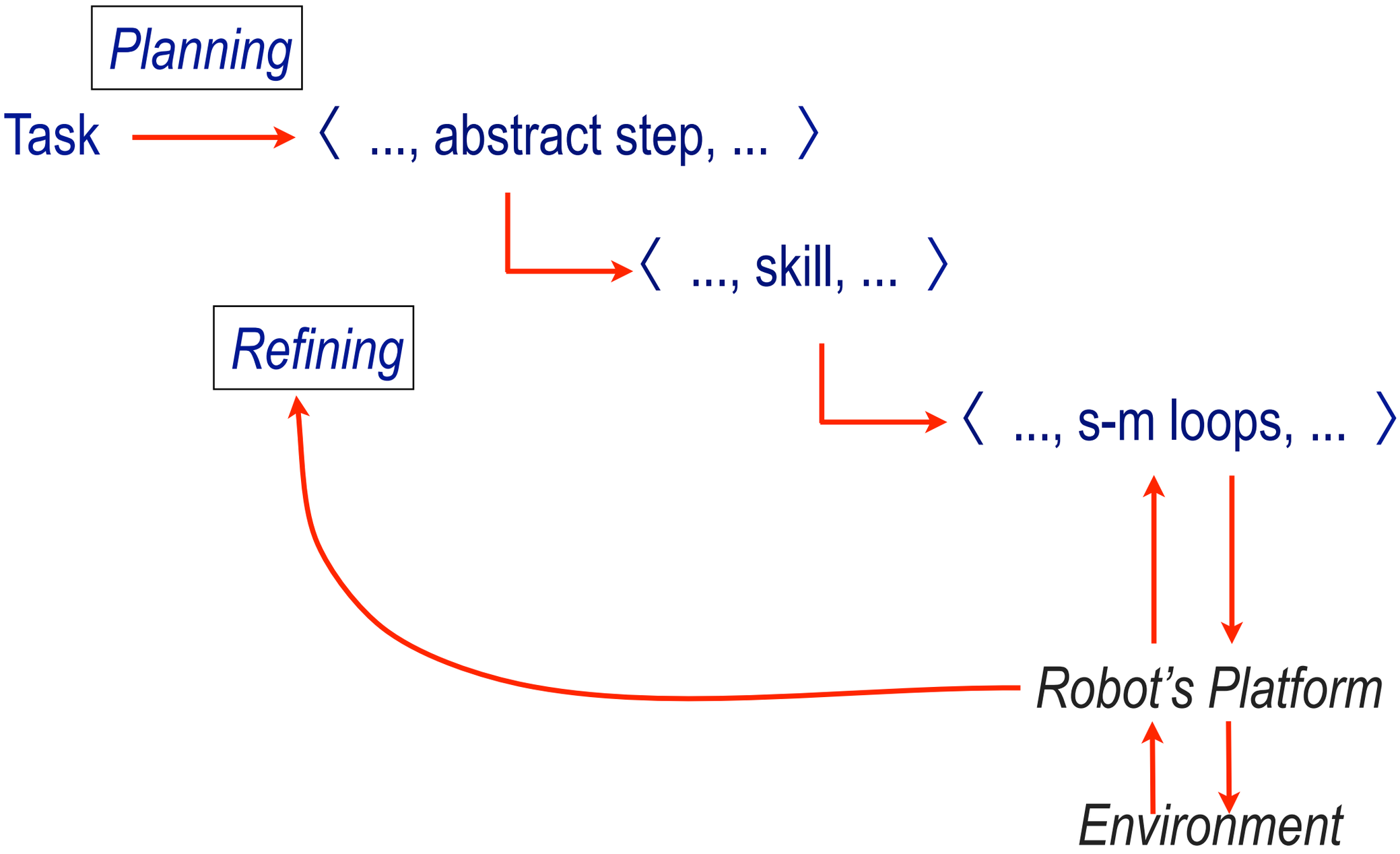




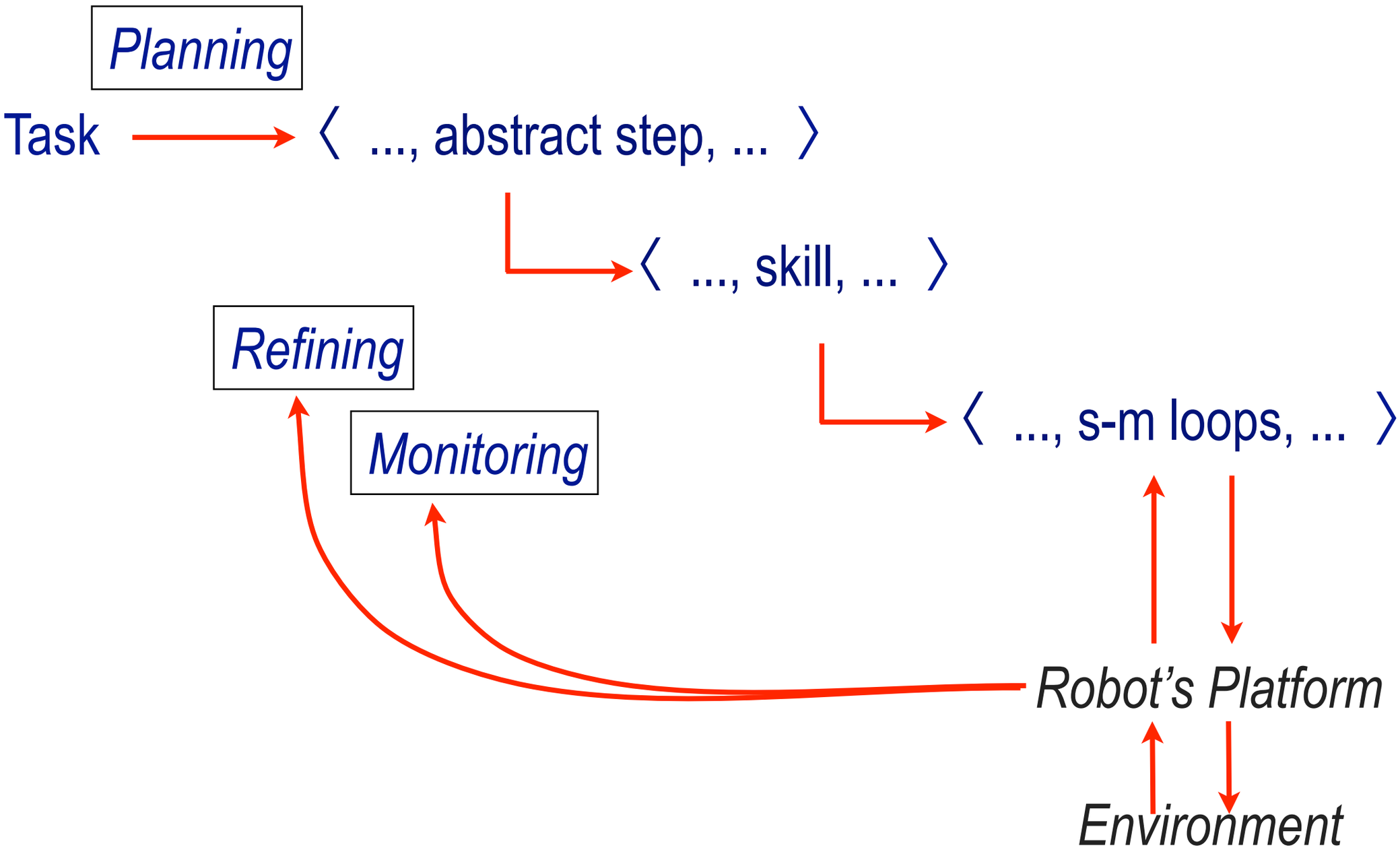
Actor's view of deliberation



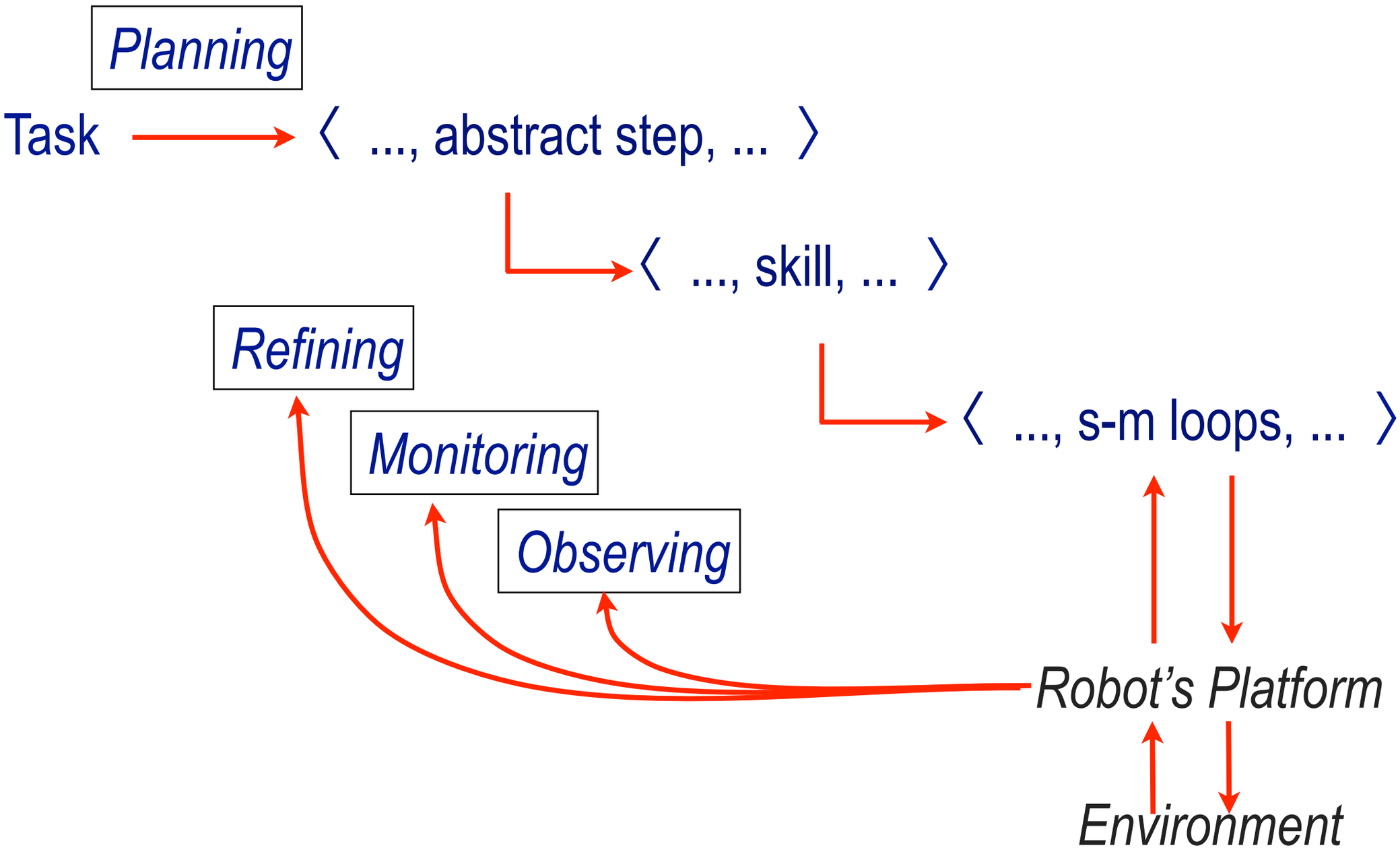
Actor's view of deliberation



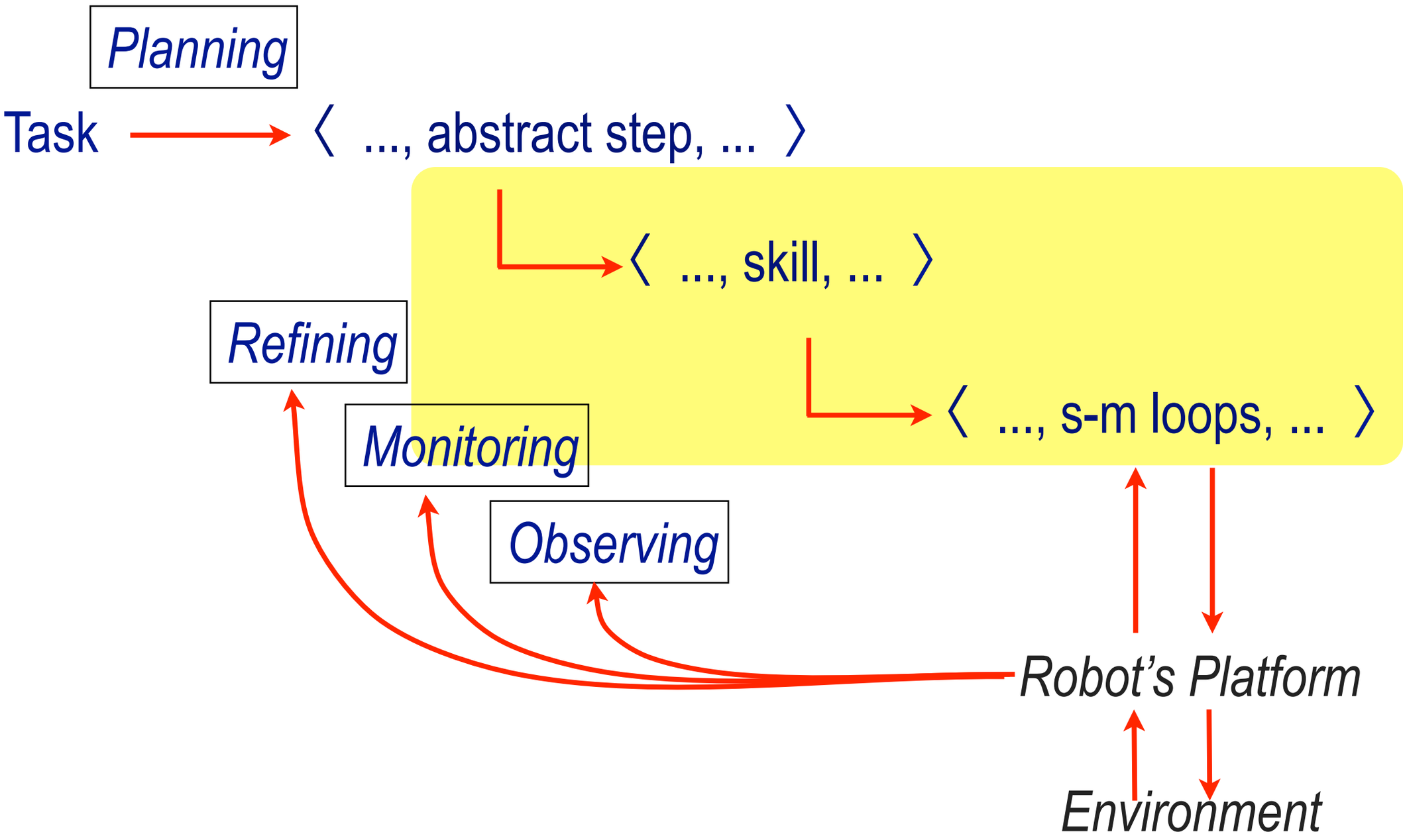
Actor's view of deliberation



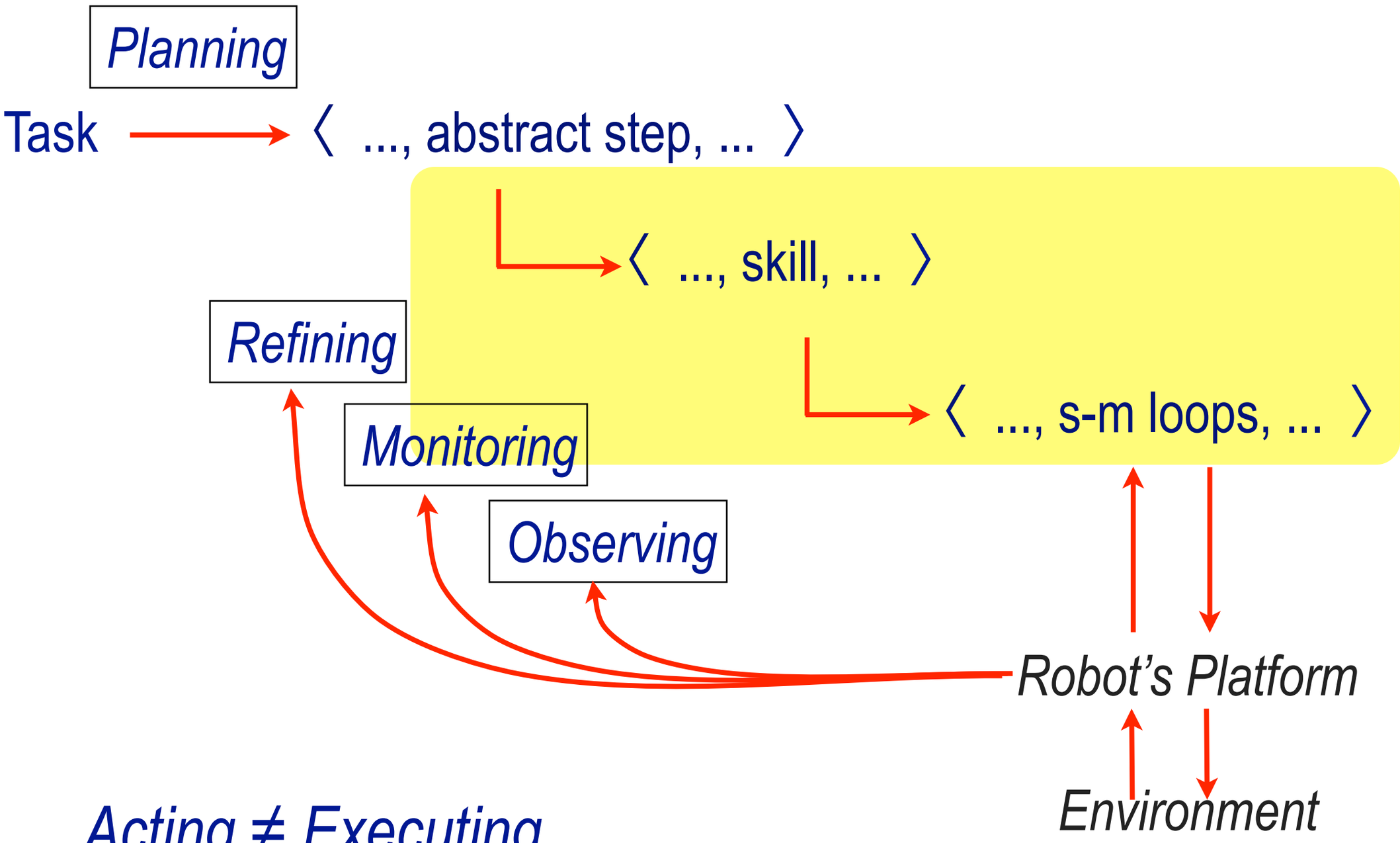
Actor's view of deliberation



Actor's view of deliberation

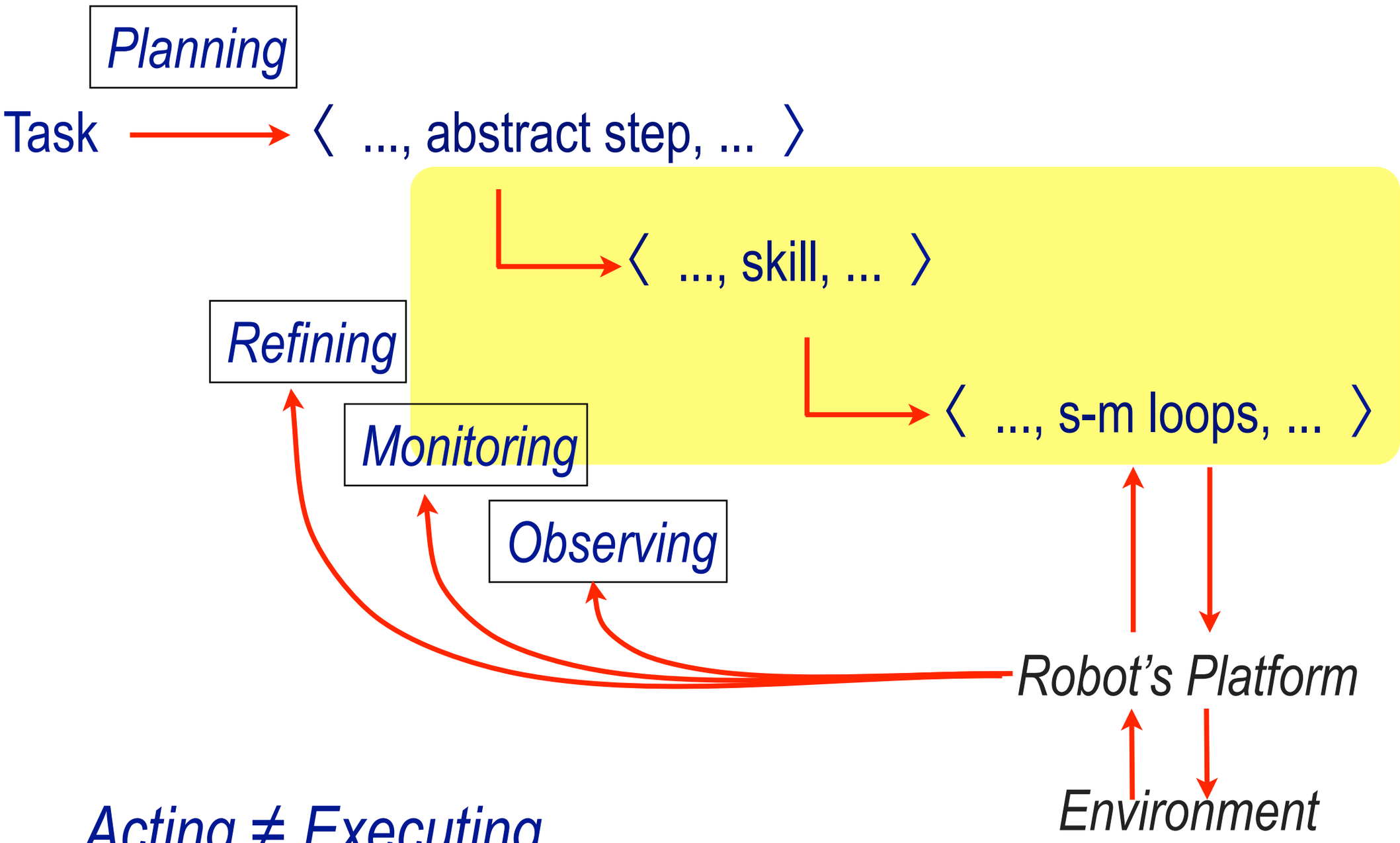


Actor's view of deliberation



Acting ≠ Executing

Actor's view of deliberation



Acting ≠ Executing

Actor ≠ Executor

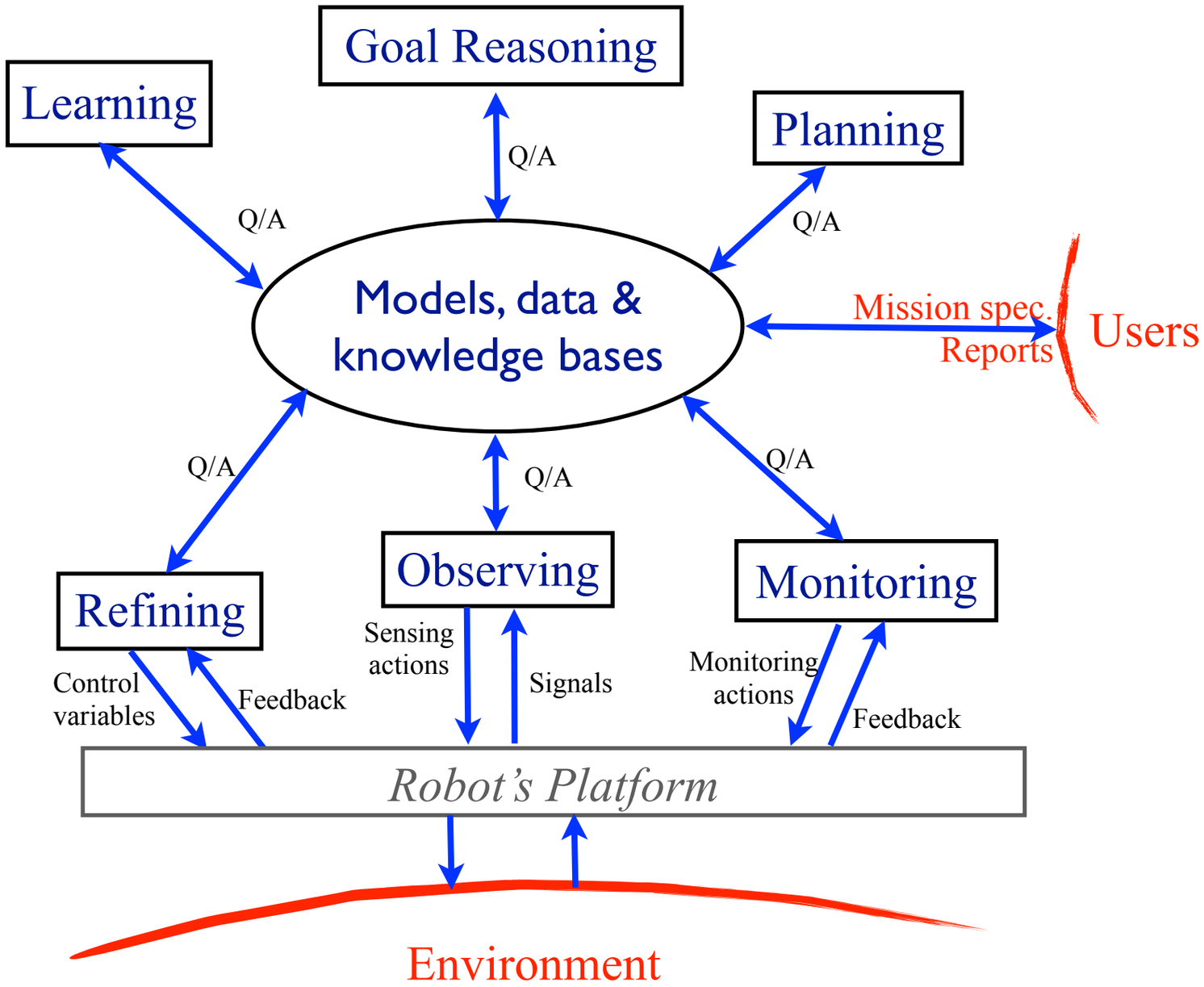
Outline

✓Motivations

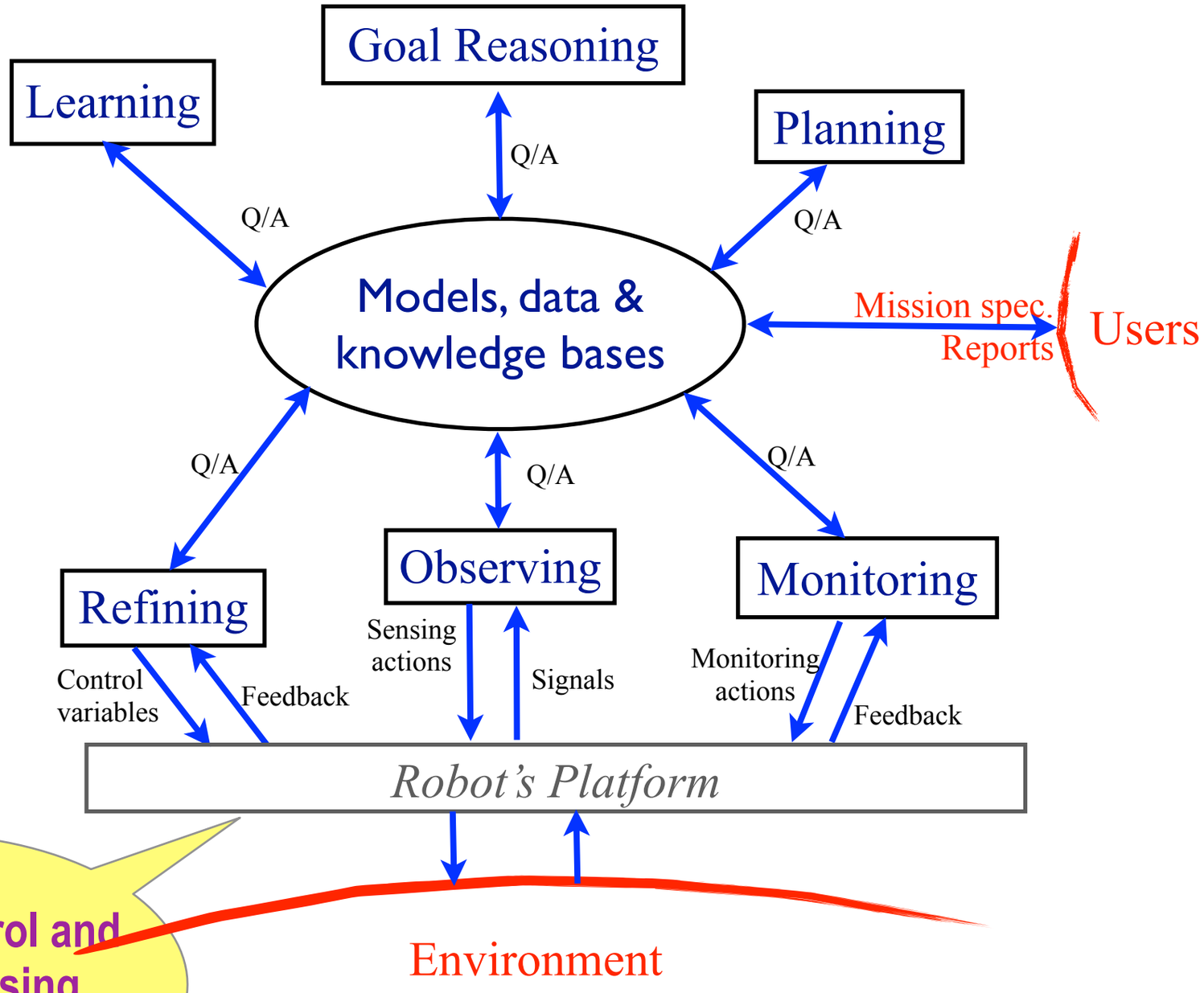
- Deliberation functions
  - Planning
  - Refining
  - Monitoring
  - Perceiving
  - Goal reasoning
  - Learning
  - Integration

- Research Challenges
  - Representation
  - Model acquisition & Verification
  - Synthesis
  - Monitoring and Goal reasoning
  - Integration

Actor's deliberation functions

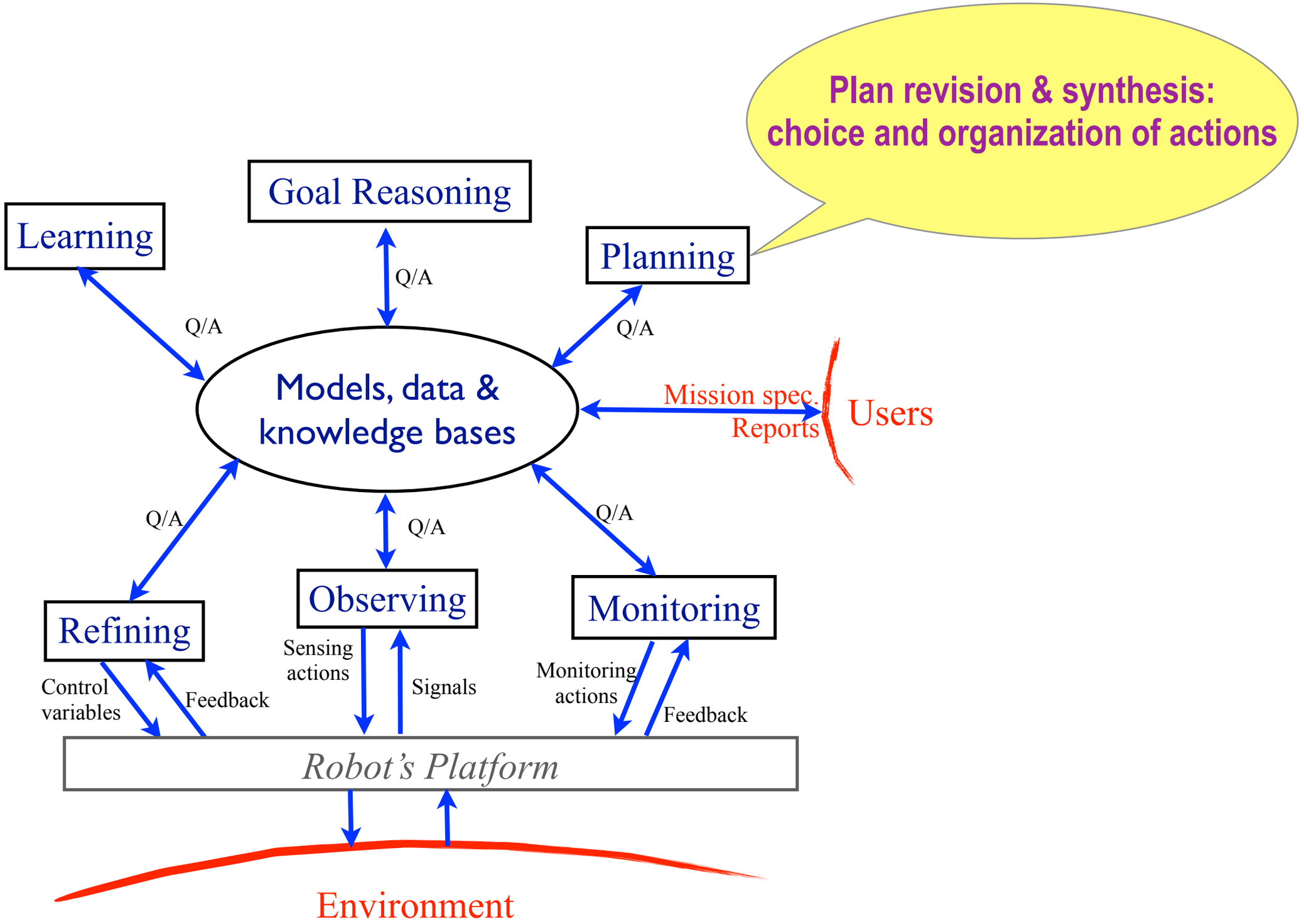


Actor's deliberation functions

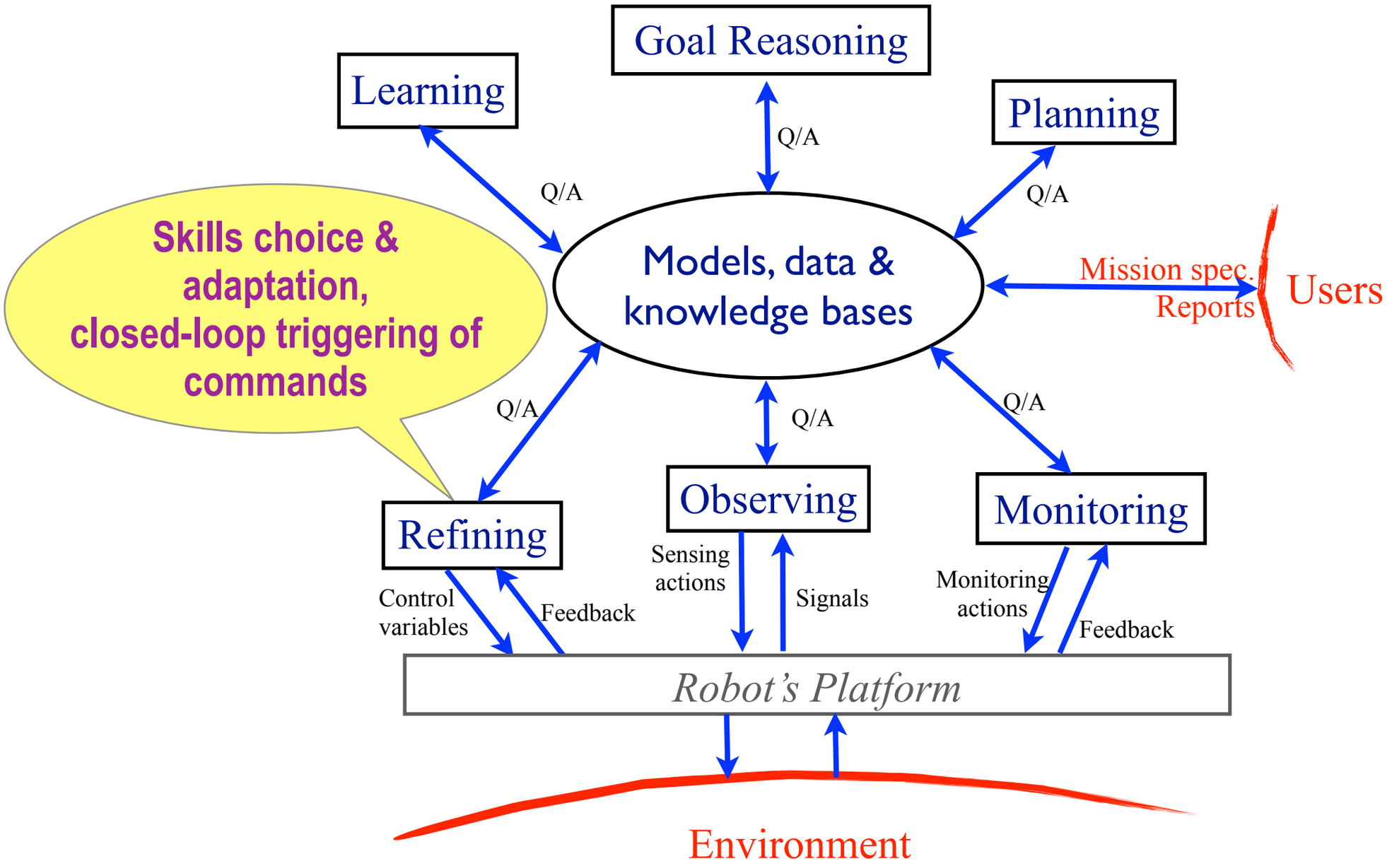


Low-level control and signal processing, *no deliberation*

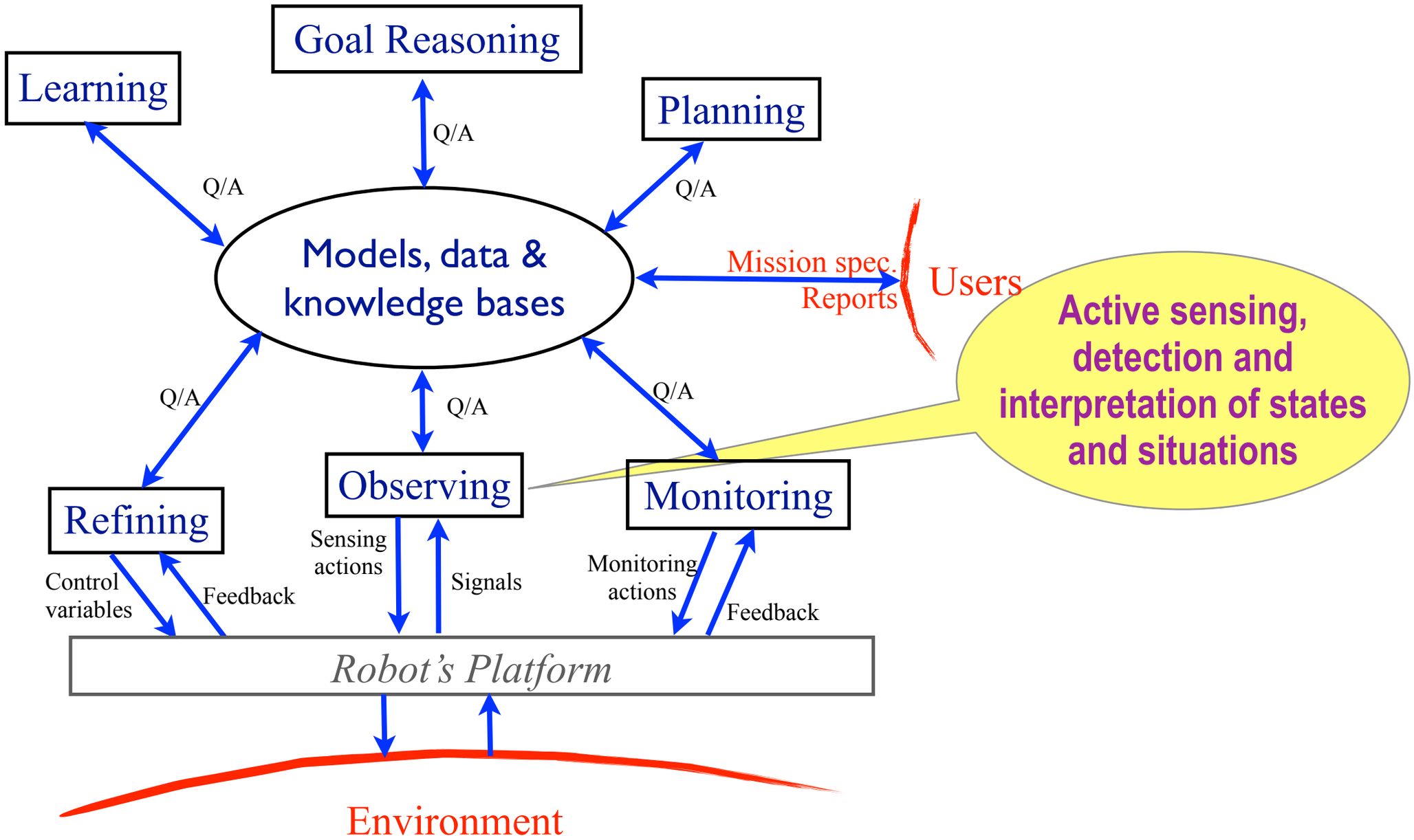
Actor's deliberation functions



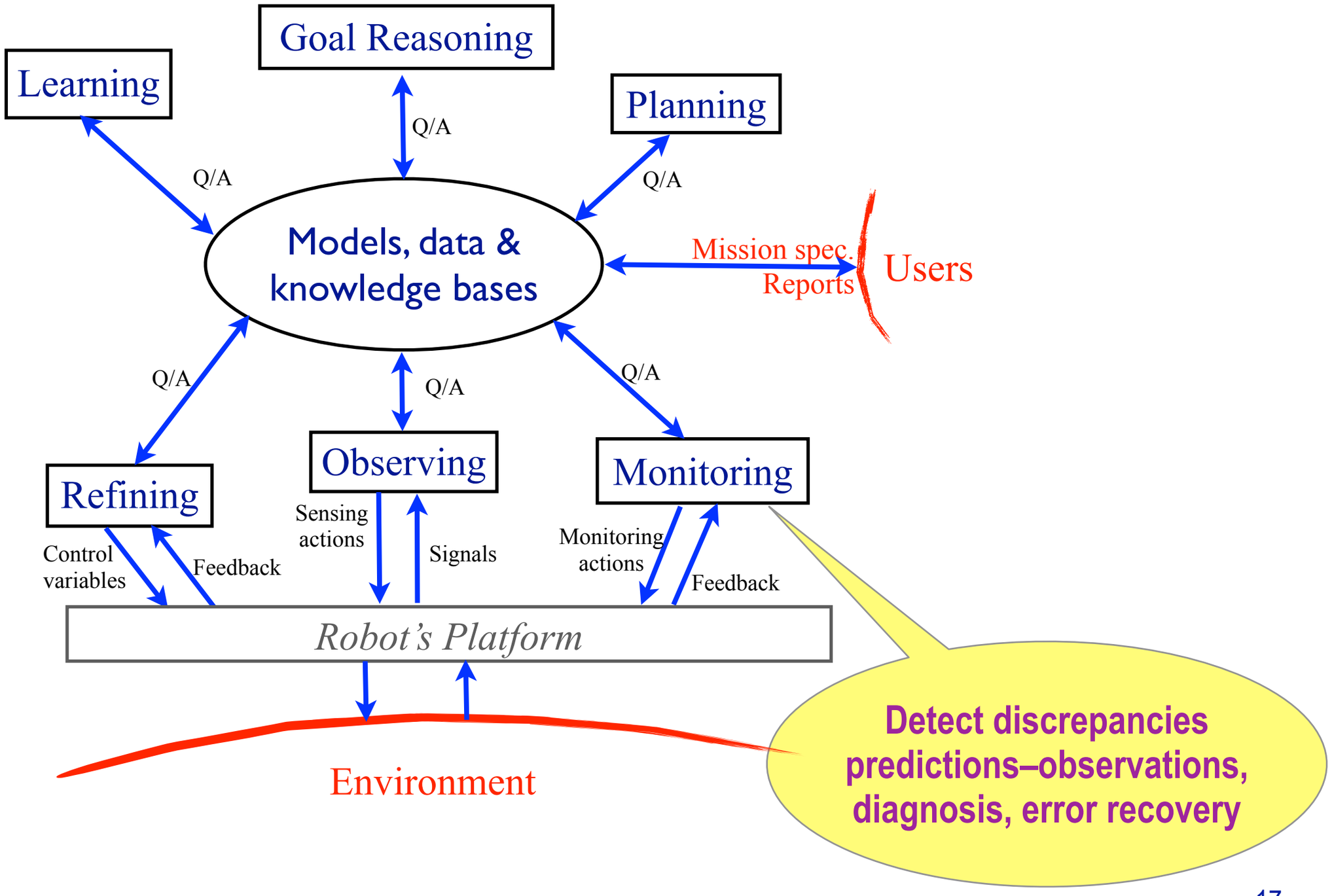
Actor's deliberation functions



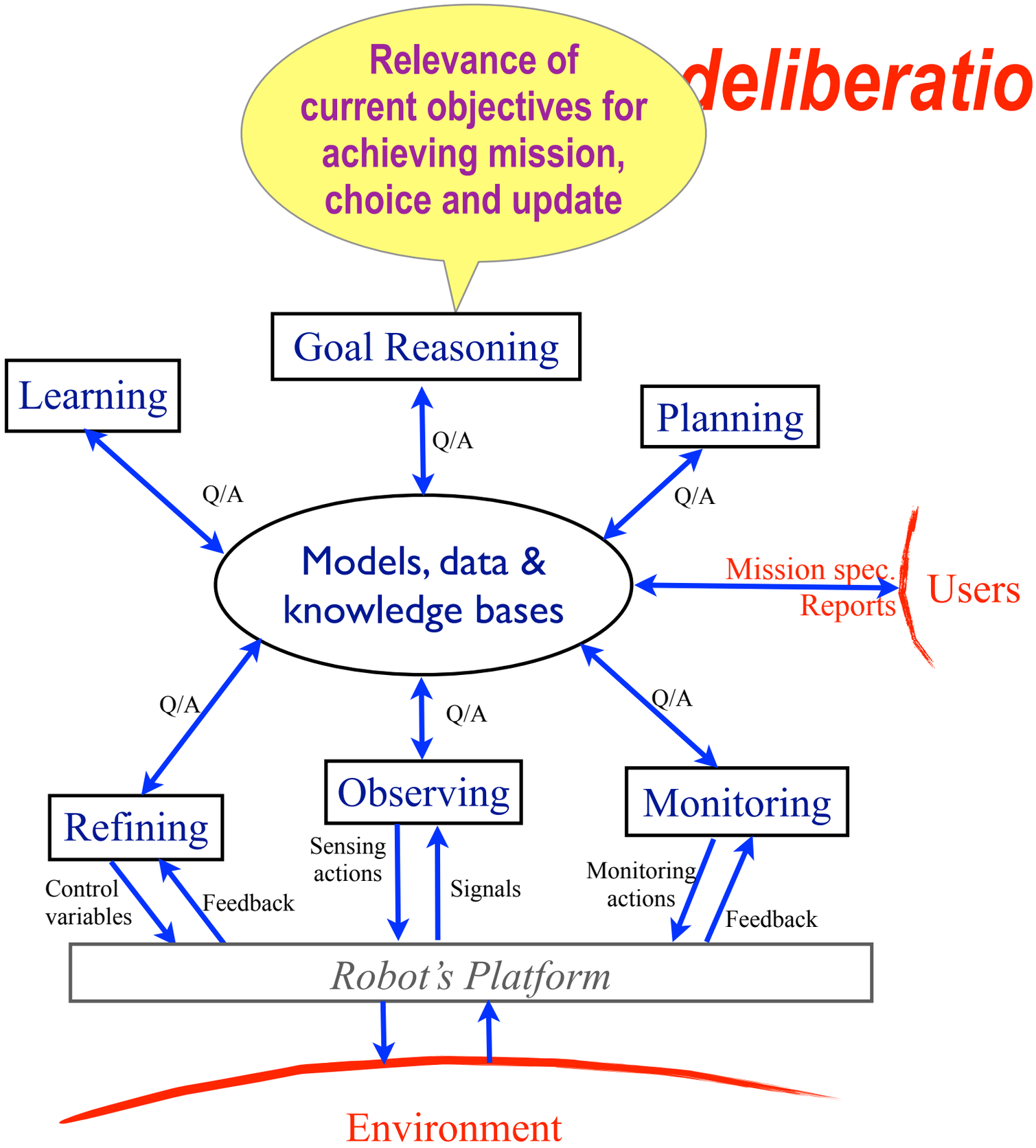
Actor's deliberation functions



Actor's deliberation functions

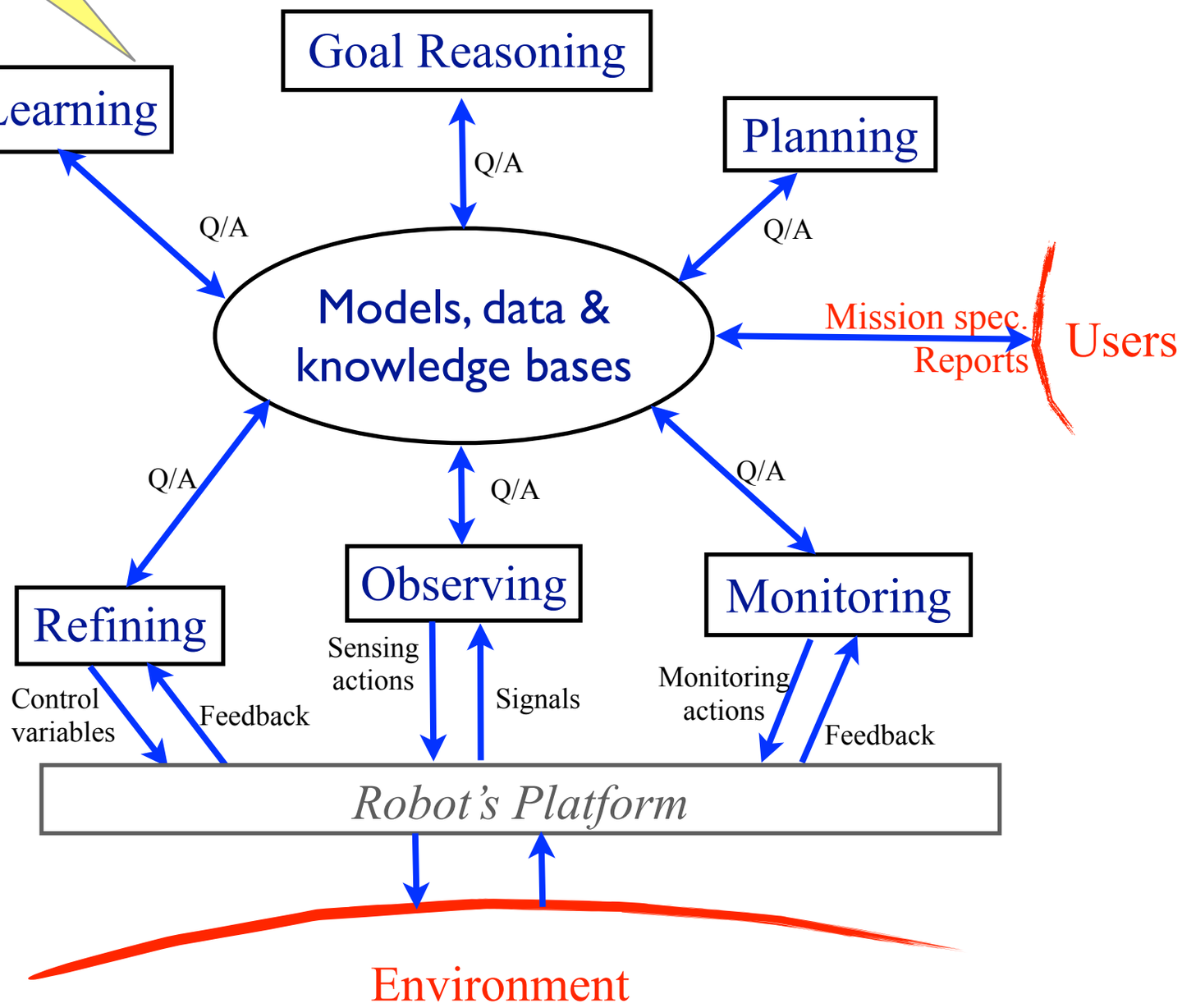


deliberation functions

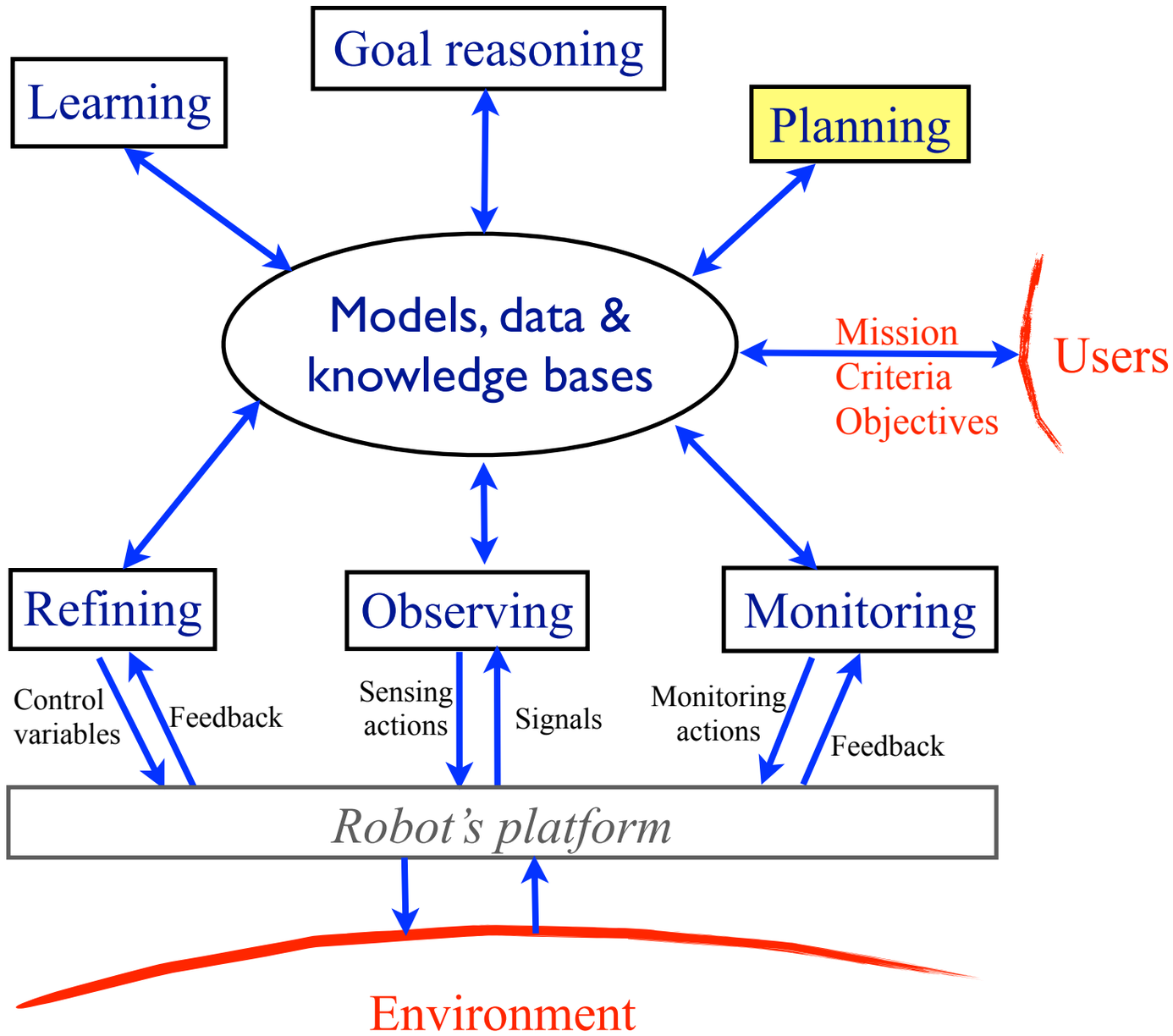


Actor's deliberation functions

Active learning to acquire and improve models and behaviors



Actor's deliberation functions



Planning in Robotics

- ***Different types of actions***

  => Different predictive models

  => Different planning problems and techniques

  - Motion and manipulation planning
  - Perception planning
  - Navigation planning
  - Communication planning
  - Task planning

Planning in Robotics

- ***Different types of actions***

  => Different predictive models

  => Different planning problems and techniques

  - Motion and manipulation planning
  - Perception planning
  - Navigation planning
  - Communication planning
  - Task planning

  ➡ Integration of planners with distinct state and action spaces

Planning in Robotics

- *Usual context*
  - Open, dynamic environment
  - Interaction with users
  - Online, interleaved with acting
- *Desirable features*
  - Access to external domain knowledge
  - Manage concurrency
  - Reason on uncertainty
- *Approaches*
  - Hierarchical
  - Temporal
  - Probabilistic

Planning in Robotics

- *Usual context*
  - Open, dynamic environment
  - Interaction with users
  - Online, interleaved with acting
- *Desirable features*
  - Access to external domain knowledge
  - Manage concurrency
  - Reason on uncertainty
- *Approaches*
  - Hierarchical
  - Temporal
  - Probabilistic

} ANML

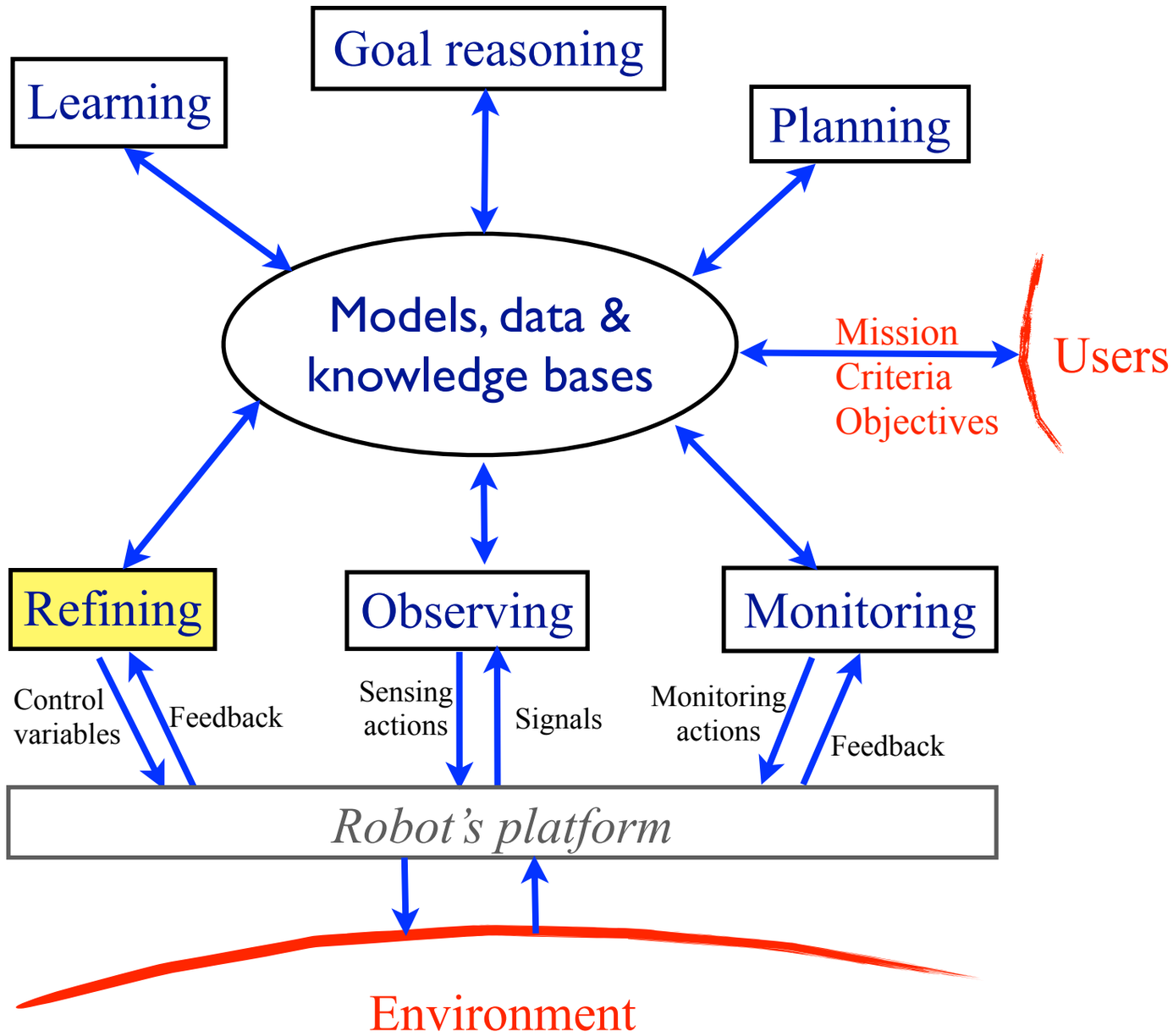
Planning in Robotics

- *Usual context*
  - Open, dynamic environment
  - Interaction with users
  - Online, interleaved with acting
- *Desirable features*
  - Access to external domain knowledge
  - Manage concurrency
  - Reason on uncertainty
- *Approaches*
  - Hierarchical
  - Temporal
  - Probabilistic

} ANML

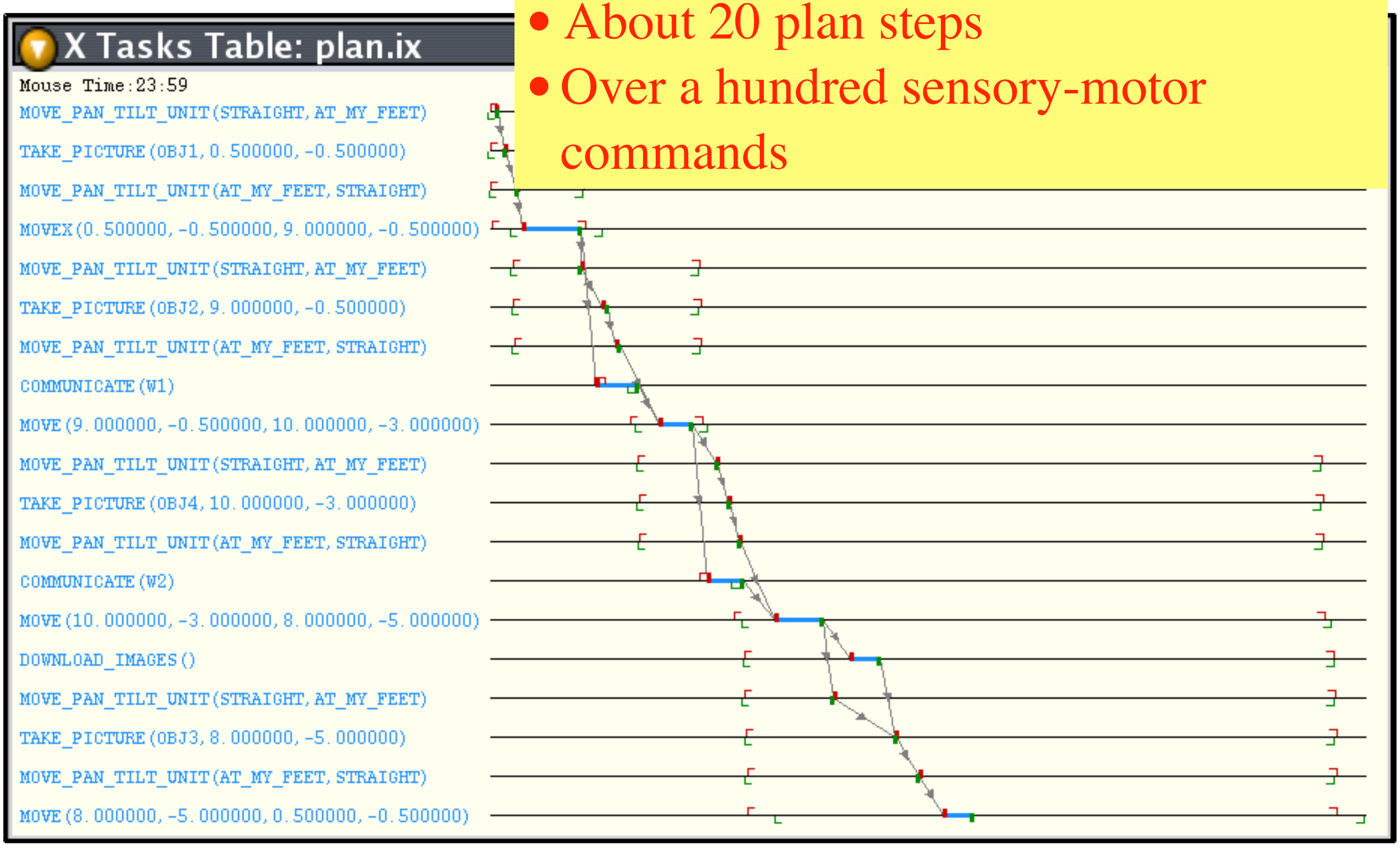
=> Sparse probabilistic models

Actor's deliberation functions



Refining abstract steps into skills

- About 20 plan steps
- Over a hundred sensory-motor commands



Refining abstract steps into skills

20 abstract steps:
Descriptive models
What, When, Why

Refinement

100 s-m commands:
Operational models
Modus operandi
How

Refining abstract steps into skills

20 abstract steps:
Descriptive models
What, When, Why

Refinement

100 s-m commands:
Operational models
Modus operandi
How

How to get the operational models ?

Refining abstract steps into skills

20 abstract steps:
Descriptive models
What, When, Why

Refinement

100 s-m commands:
Operational models
Modus operandi
How

How to get the operational models ?

- Specification

Refining abstract steps into skills

20 abstract steps:
Descriptive models
What, When, Why

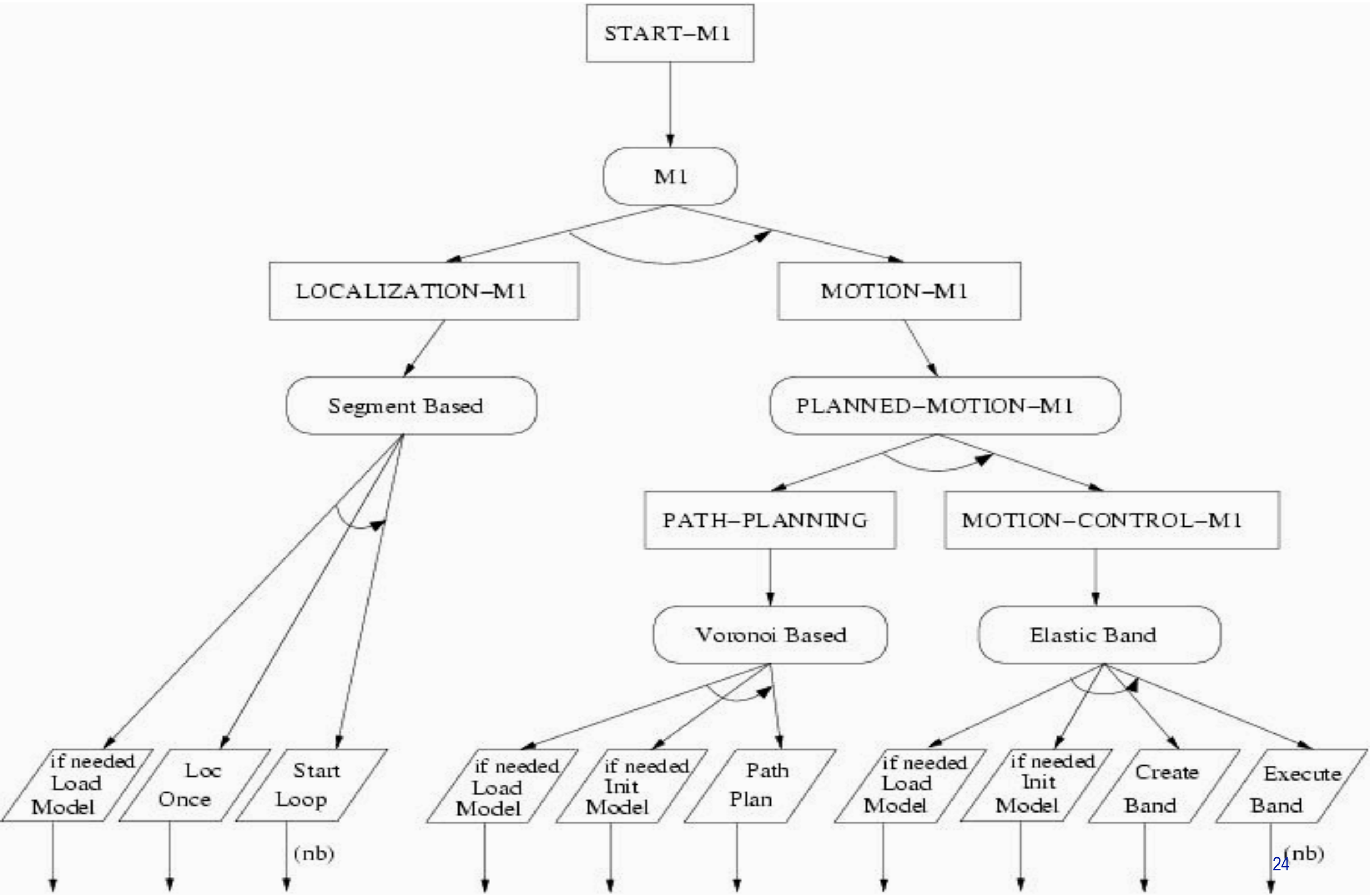
Refinement

100 s-m commands:
Operational models
Modus operandi
How

How to get the operational models ?

- Specification
- Synthesis

Synthesis of skills achieving an action



Approaches

Approaches

- Informal specification
  PRS, RAP, TDL, XFRM
  - Flexible
  - Hand-written

Approaches

- Informal specification
  PRS, RAP, TDL, XFRM
  - Flexible
  - Hand-written
- State machine approaches
  PLEXIL, BPEL, Smach
  - More open to V&V
  - Hand-written

Approaches

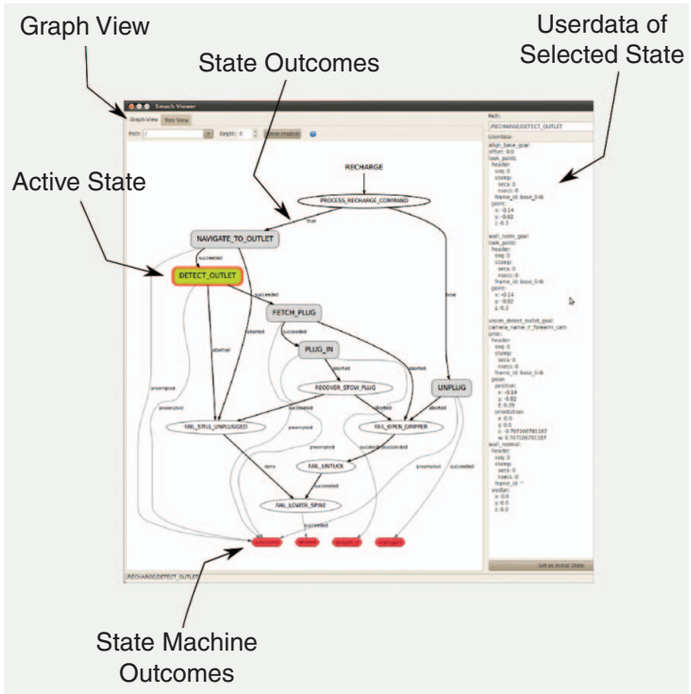
- Informal specification
  PRS, RAP, TDL, XFRM
  - Flexible
  - Hand-written
- State machine approaches
  PLEXIL, BPEL, Smach
  - More open to V&V
  - Hand-written
- CSP-based approaches
  IxTeT-eXeC, IDEA, T-ReX
  - Handle time
  - Dispatching, plan extension,revision

Approaches

- Informal specification
  PRS, RAP, TDL, XFRM
  - Flexible
  - Hand-written
- State machine approaches
  PLEXIL, BPEL, Smach
  - More open to V&V
  - Hand-written
- CSP-based approaches
  IxTeT-eXeC, IDEA, T-ReX
  - Handle time
  - Dispatching, plan extension,revision
- Stochastic approaches (MDP, DBN/DDN,)
  - Learned from experiences or from teaching

Approaches

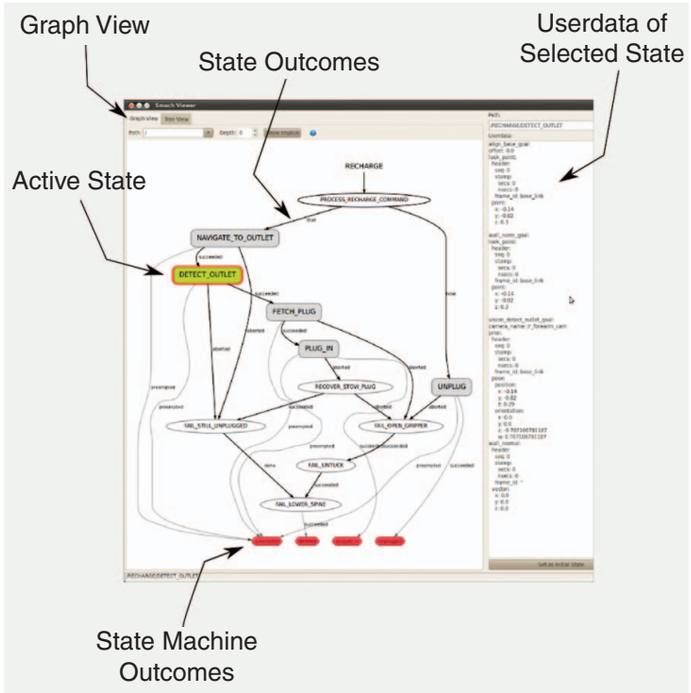
- Informal specification
  PRS, RAP, TDL, XFRM
  - Flexible
  - Hand-written
- State machine approaches
  PLEXIL, BPEL, Smach
  - More open to V&V
  - Hand-written
- CSP-based approaches
  IxTeT-eXeC, IDEA, T-ReX
  - Handle time
  - Dispatching, plan extension,revision
- Stochastic approaches (MDP, DBN/DDN,)
  - Learned from experiences or from teaching



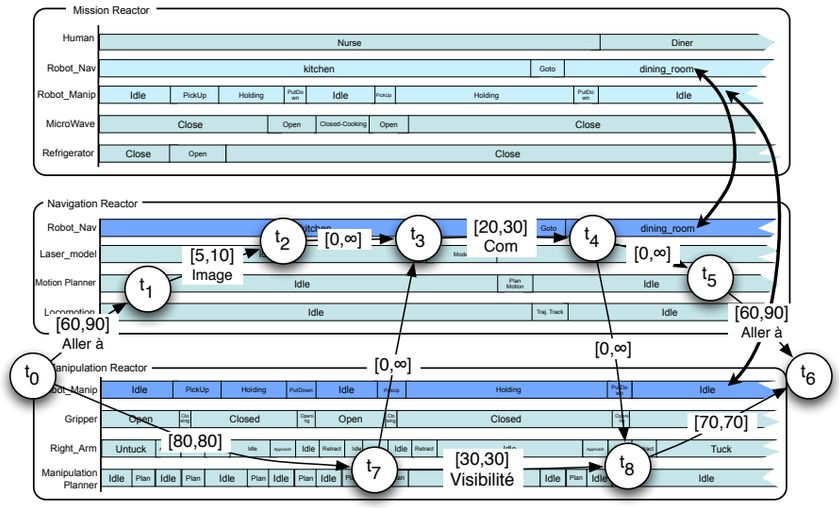
[Smach (ROS)]

Approaches

- Informal specification
  PRS, RAP, TDL, XFRM
  - Flexible
  - Hand-written
- State machine approaches
  PLEXIL, BPEL, Smach
  - More open to V&V
  - Hand-written
- CSP-based approaches
  IxTeT-eXeC, IDEA, T-ReX
  - Handle time
  - Dispatching, plan extension,revision
- Stochastic approaches (MDP, DBN/DDN,)
  - Learned from experiences or from teaching



[Smach (ROS)]



[T-Rex / IDEA]

Approaches

Approaches	Systems
Direct	STRIPS/Planex
Procedure	RAP
	PRS
	Cypress/CPEF
	TCA/TDL
	XFRM/RPL/SRP
Petri-Net	IMRS
	Procosa
	Petri-Net Plans
Automata Graph	FSA
	PLEXIL
	RMPL
	SMACH
Logic	Golex
	ReadyLog
CSP	IxTeT
	RMPL
	IDEA/T-ReX
	Casper
MDP	PEARL
	K9 MDP
	Robel
	Ressac

Approaches

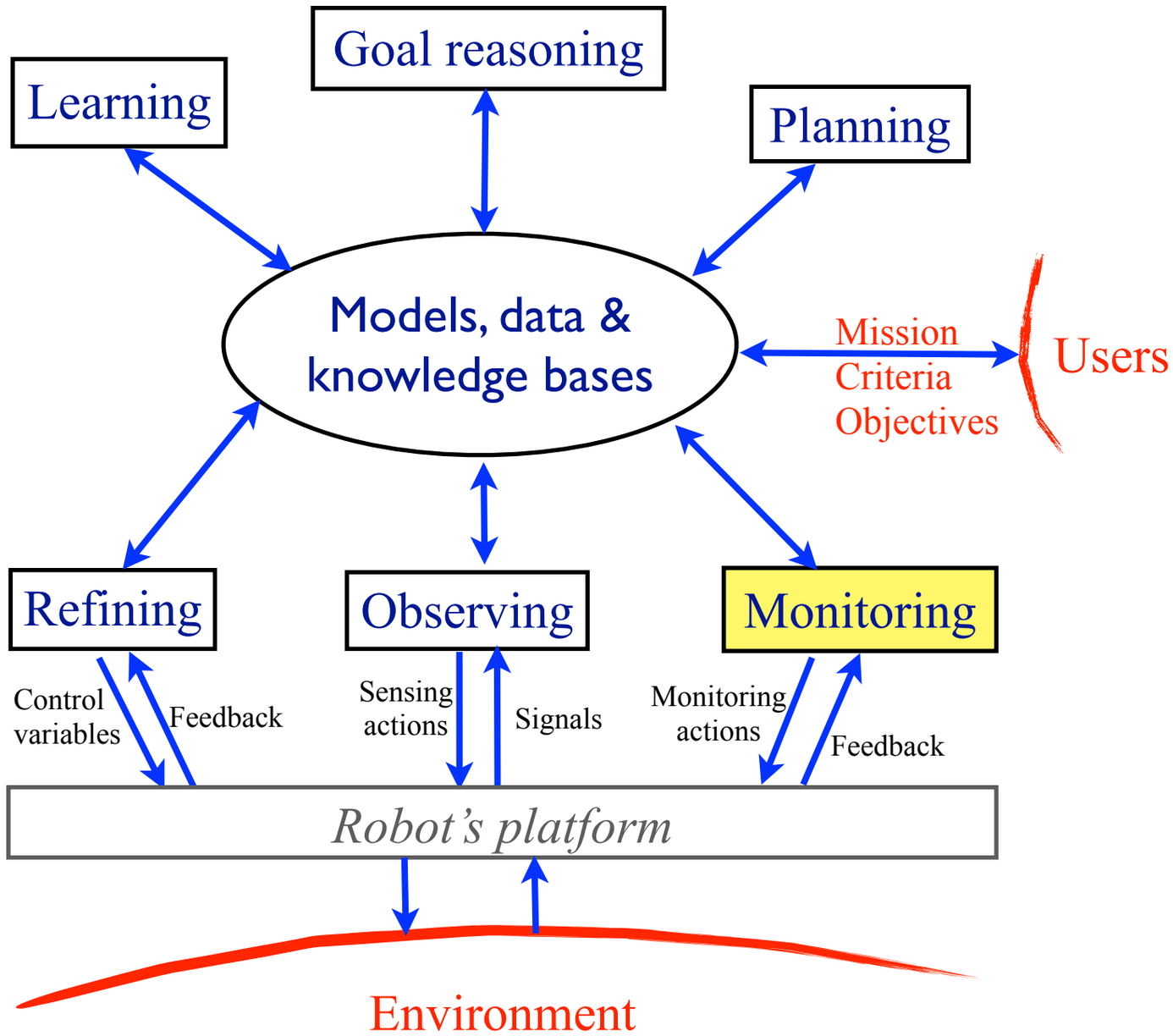
		Functions					Knowledge Representation			
Approaches	**Systems**	***Refinement***	***Instantiation***	***Time handling***	***Nondeterminism***	***Repair***	***Hand-Written***	***Model Based***	***V&V***	***Similar to Planning KR***
Direct	STRIPS/Planex					X	X			
Procedure	RAP	X	X				X			
	PRS	X	X				X			
	Cypress/CPEF	X	X				X			ACT
	TCA/TDL	X	X	X			X			
	XFRM/RPL/SRP	X	X			X	X			X
Petri-Net	IMRS	X		coordination			X	X	X	
	Procosa	X		coordination			X	X	X	
	Petri-Net Plans	X		coordination			X	X	X	
Automata Graph	FSA	X	X	X			X	X		Link
	PLEXIL	X	X	X			X	X		
	RMPL	X	X	X			X	X		
	SMACH	X	X	X			X	X		
Logic	Golex	X	X				X		X	Same lang. not same model
	ReadyLog	X	X				X		X	
CSP	IxTeT	X	X	X		X		X		X
	RMPL	X	X	X	X	X		X		X
	IDEA/T-ReX	X	X	X		X		X		X
	Casper	X	X	X		X		X		X
MDP	PEARL				X					
	K9 MDP				X					
	Robel				X					
	Ressac				X					

Refining abstract steps into skills

Descriptive models
Know-What

Operational models
Know-How

Actor's deliberation functions



Monitoring

- Functions
  - Survey actor's predictions

    in plans, skills and environment models
  - Detect discrepancies = predictions - observations
  - Explain and diagnose discrepancies
  - Recover: trigger first reactions and repair actions

Approaches

Approaches

- Hand-written procedures to monitor applicability/maintenance conditions

Approaches

- Hand-written procedures to monitor applicability/maintenance conditions
- Model-based supervision and diagnosis, e.g., DS1
  more proprioceptive then exteroceptive

Approaches

- Hand-written procedures to monitor applicability/maintenance conditions
- Model-based supervision and diagnosis, e.g., DS1
  more proprioceptive then exteroceptive
- Monitor plan invariants

Approaches

- Hand-written procedures to monitor applicability/maintenance conditions
- Model-based supervision and diagnosis, e.g., DS1
  more proprioceptive then exteroceptive
- Monitor plan invariants
- Synthesize additional invariants from extended planning problems which guarantees the plan execution, e.g., TALPlan

Approaches

- Hand-written procedures to monitor applicability/maintenance conditions
- Model-based supervision and diagnosis, e.g., DS1
  more proprioceptive then exteroceptive
- Monitor plan invariants
- Synthesize additional invariants from extended planning problems which guarantees the plan execution, e.g., TALPlan
- Model checking execution traces with LTL

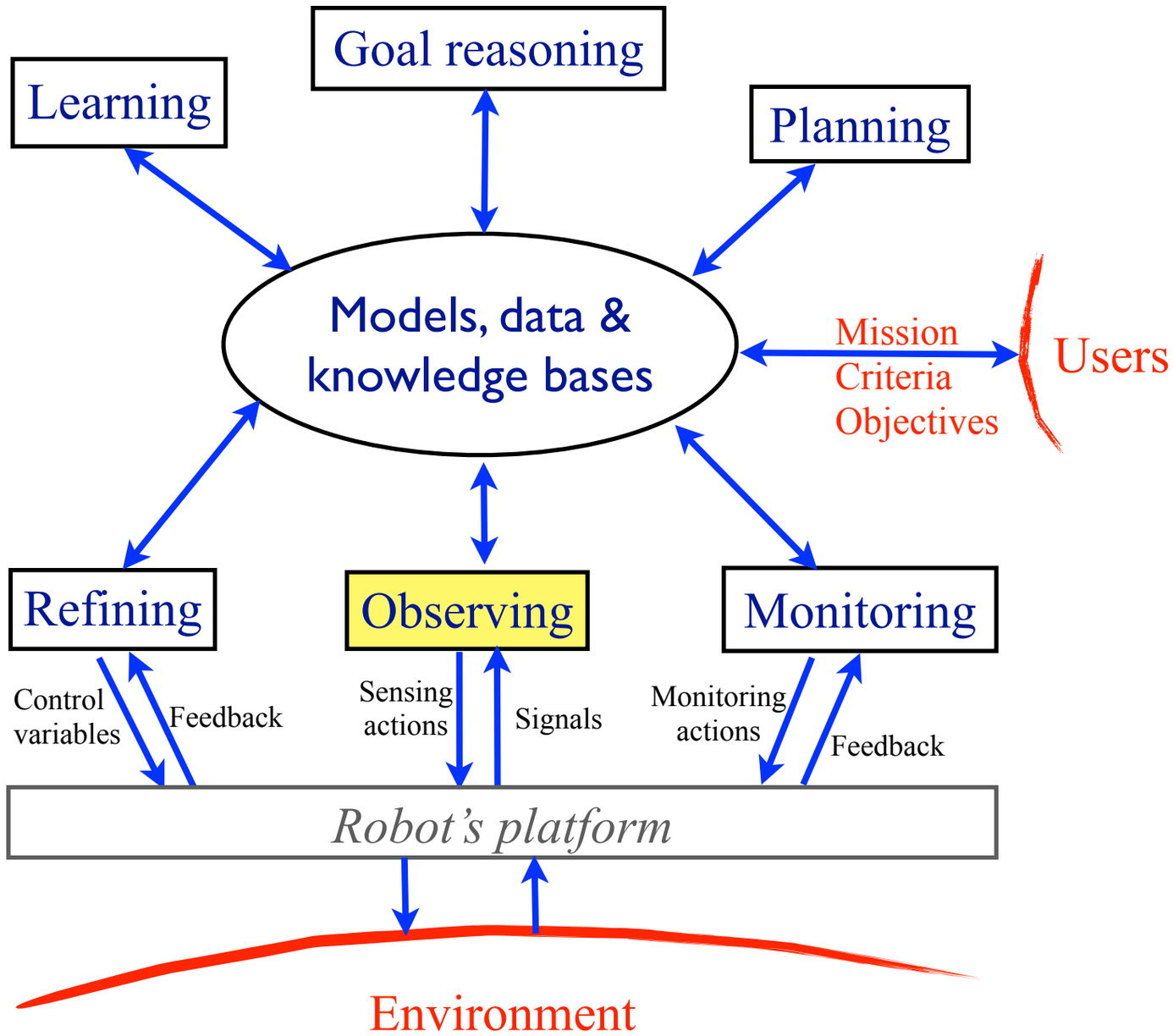
Approaches

- Hand-written procedures to monitor applicability/maintenance conditions
- Model-based supervision and diagnosis, e.g., DS1
  more proprioceptive then exteroceptive
- Monitor plan invariants
- Synthesize additional invariants from extended planning problems which guarantees the plan execution, e.g., TALPlan
- Model checking execution traces with LTL
- Constraint Based Automata + Control programs, e.g., RMPL

Approaches

	Knowledge Representation	*Search & Algorithm*	*Link with other Deliberative Functions*
Planex	Triangular table		with planning and exec
Livingstone	Qualitative model Transition automata with probabilities and costs	ITMS with conflict directed best-first search	
RMPL	Hierarchical constraint-based automata	Reactive programming	with planning
[Fraser et al., 2005]	Logical invariant	Logical satisfiability	with planning
[Fichtner et al., 2003]	Fluent Calculus	Prioritized non monotonic default logic	
[Lamine and Kabanza, 2002]	Linear Temporal Logic	Delayed formula progression	same technique used for planning
[Petterssonetal.,2003]	Neural net		
SKEMon	Description Logic & Bayesian Belief		
TALplanner	Temporal Action Logics	Formula progression	with planning and observing

Actor's deliberation functions



Observing

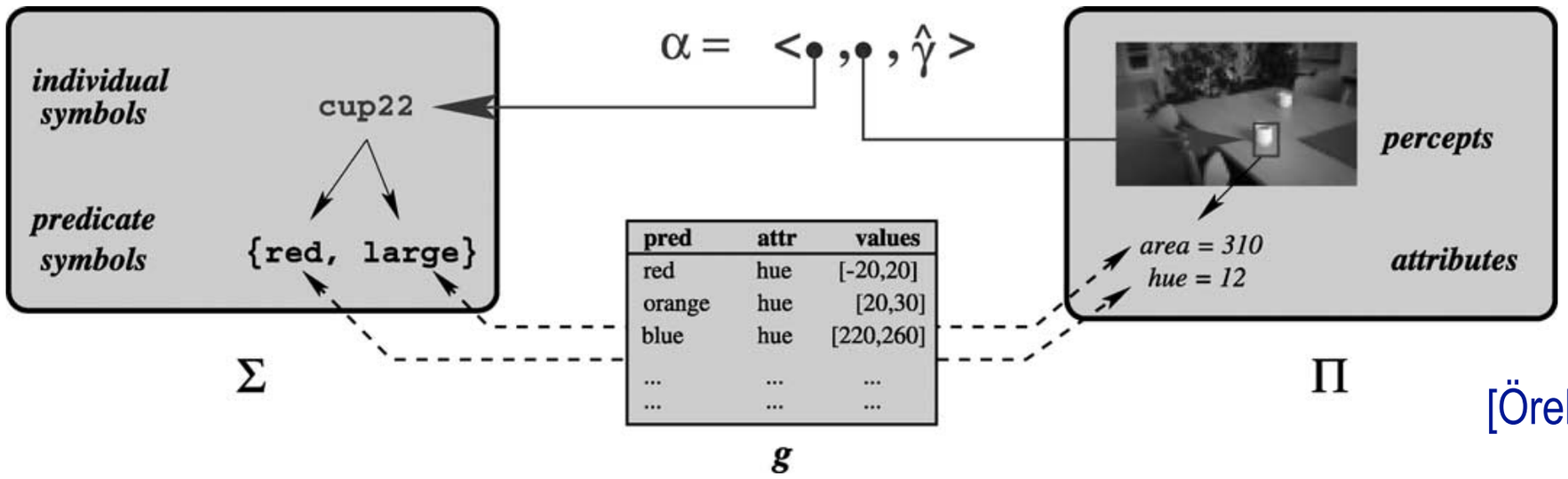
- Role
  - Process signals needed in closed loop servoing
  - Detect and structure environment features

    Recognize, categorize,

    Link signals to symbols: anchoring
    - Recognize situations and plans in observed sequences of events
- Bottom-up to from signal to symbols
- Top-down to focus attention and trigger observation actions

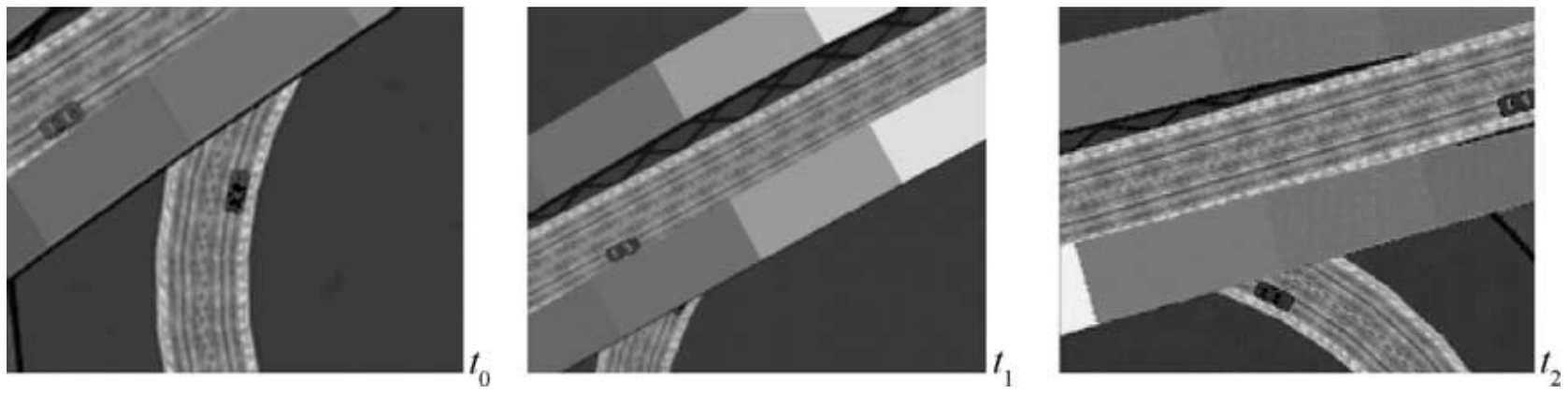
Observing

- Anchoring problem
  - Relate perceptual data and symbolic attribute corresponding to the same physical object



[Örebro Univ.]

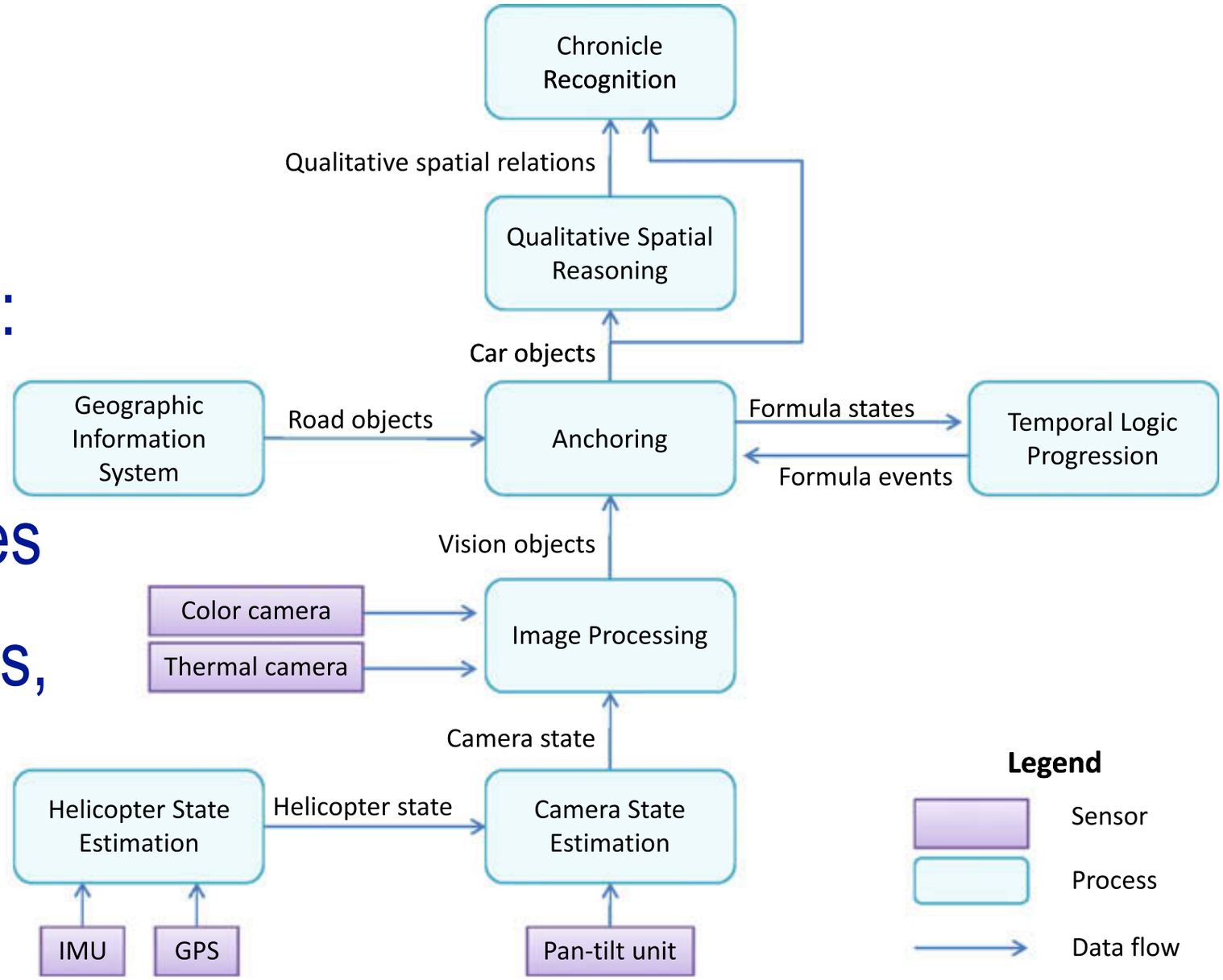
- Track anchors overtime and refine/revise hypothesis



[Linköping Univ.]

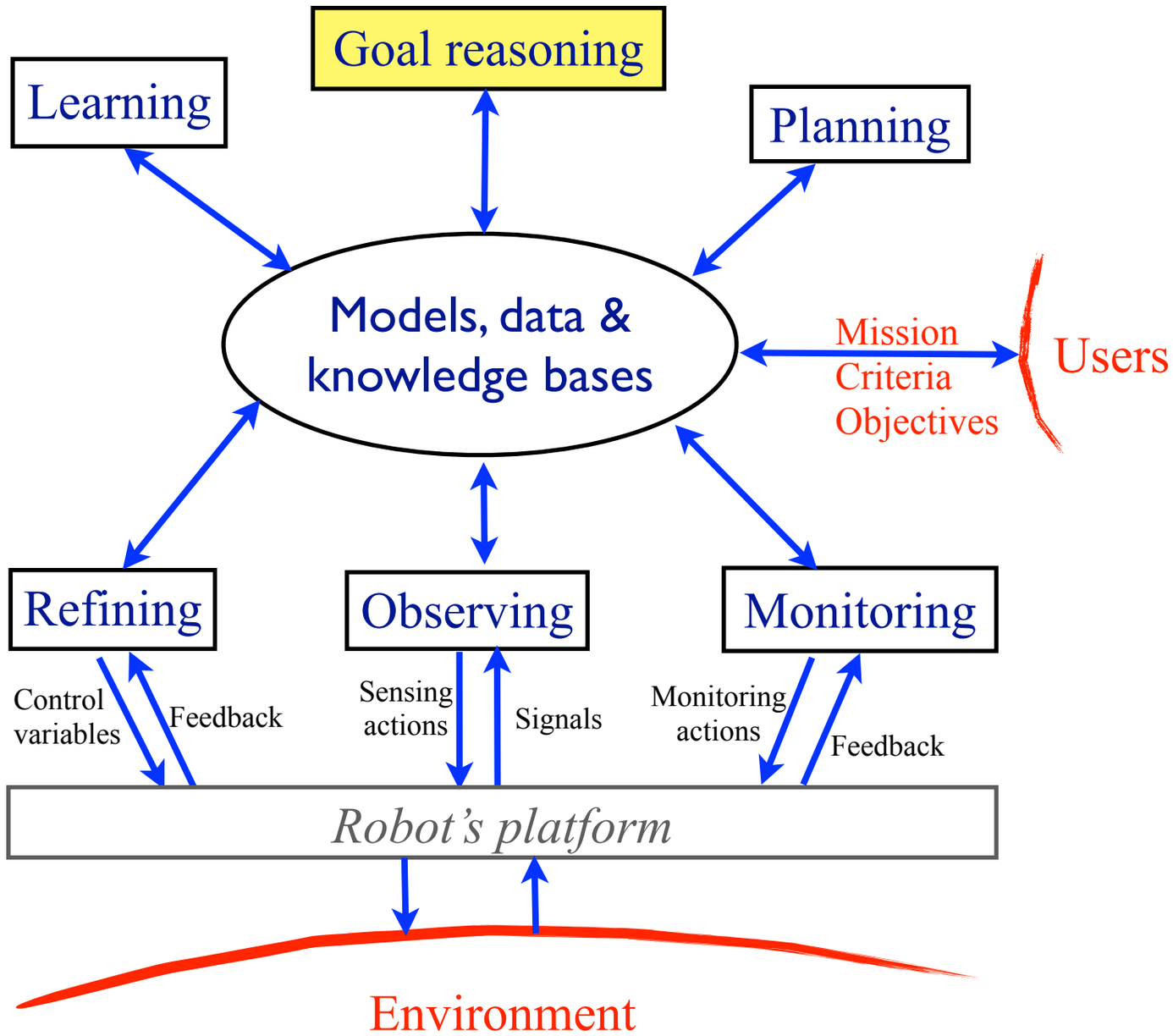
Perception Engine

- Comprehensive and coherent approach for observing
- Data flow architecture
- Stream-based formalism on perception processes:
  - primitive, refinement, configuration processes
  - policies over processes, temporal constraints
- Integration to planning and monitoring,
- Opens V&V perspectives



[Dyknow, Linköping U.]

Actor's deliberation functions



Goal Reasoning

Goal Reasoning

- Higher level monitoring *wrt* objectives, criteria and constraints

Goal Reasoning

- Higher level monitoring *wrt* objectives, criteria and constraints
- *Goal Driven Autonomy*: global mission assessment

Goal Reasoning

- Higher level monitoring *wrt* objectives, criteria and constraints
- *Goal Driven Autonomy*: global mission assessment
  - Discrepancy detector

Goal Reasoning

- Higher level monitoring *wrt* objectives, criteria and constraints
- *Goal Driven Autonomy*: global mission assessment
  - Discrepancy detector
  - Explanation generator

Goal Reasoning

- Higher level monitoring *wrt* objectives, criteria and constraints
- *Goal Driven Autonomy*: global mission assessment
  - Discrepancy detector
  - Explanation generator
  - Goal generator to address new conditions

Goal Reasoning

- Higher level monitoring *wrt* objectives, criteria and constraints
- *Goal Driven Autonomy*: global mission assessment
  - Discrepancy detector
  - Explanation generator
  - Goal generator to address new conditions
  - Goal manager

Goal Reasoning

- Higher level monitoring *wrt* objectives, criteria and constraints
- *Goal Driven Autonomy*: global mission assessment
  - Discrepancy detector
  - Explanation generator
  - Goal generator to address new conditions
  - Goal manager
    - Decision theory: tradeoff between conflicting goals

Goal Reasoning

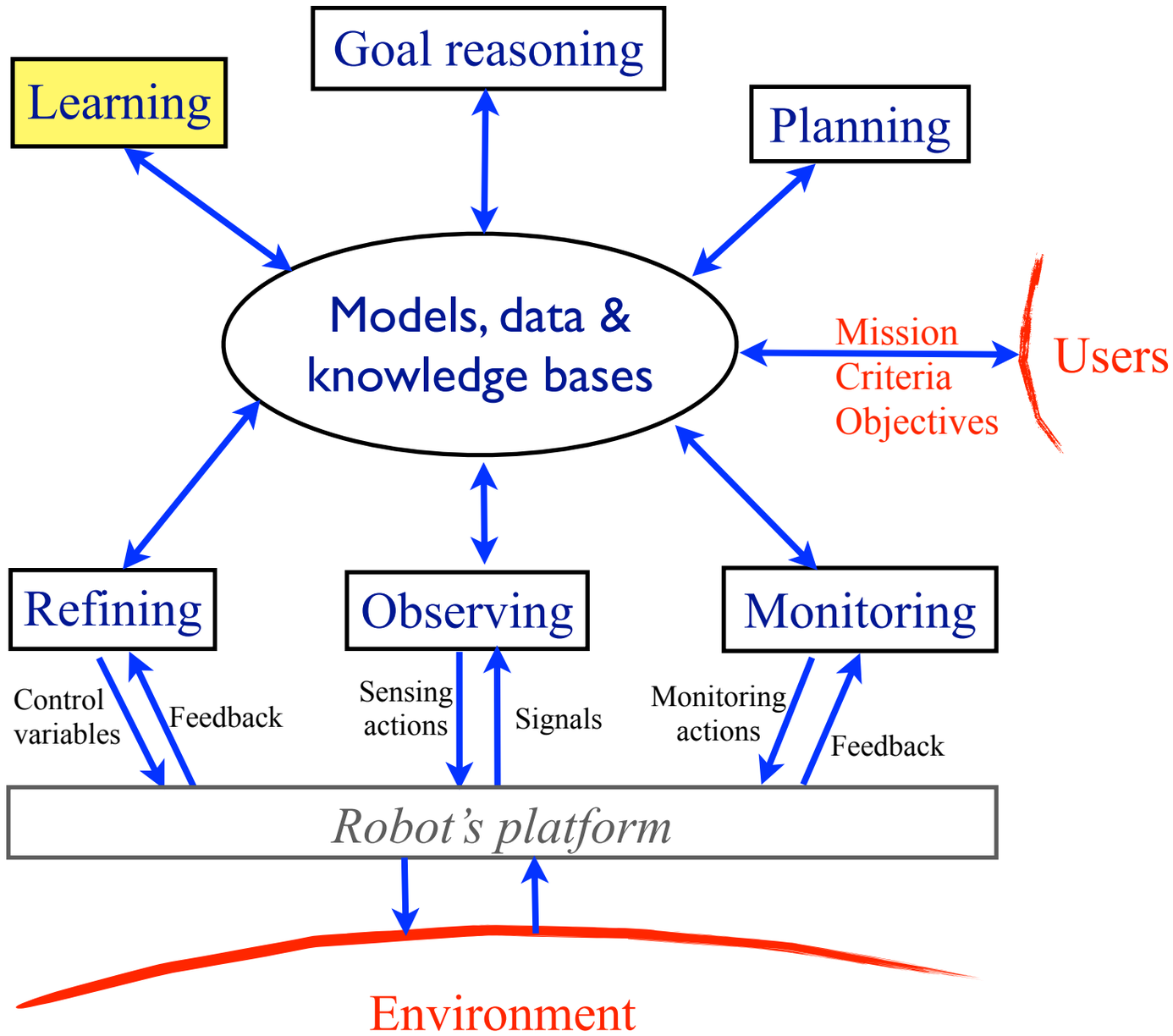
- Higher level monitoring *wrt* objectives, criteria and constraints
- *Goal Driven Autonomy*: global mission assessment
  - Discrepancy detector
  - Explanation generator
  - Goal generator to address new conditions
  - Goal manager
    - Decision theory: tradeoff between conflicting goals
    - Explicit choice

Goal Reasoning

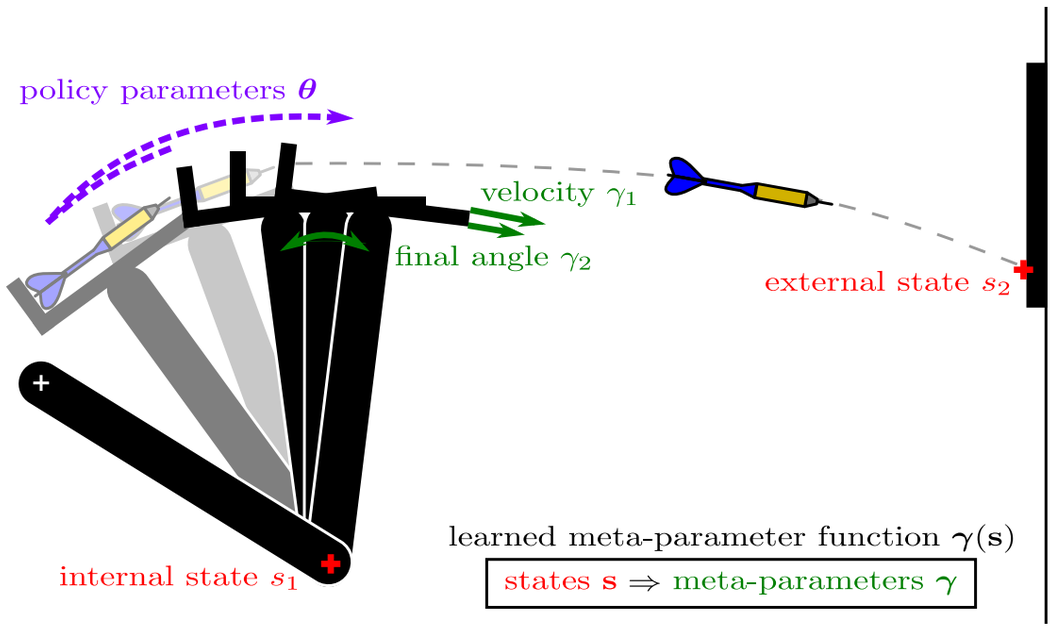
- Higher level monitoring *wrt* objectives, criteria and constraints
- *Goal Driven Autonomy*: global mission assessment
  - Discrepancy detector
  - Explanation generator
  - Goal generator to address new conditions
  - Goal manager
    - Decision theory: tradeoff between conflicting goals
    - Explicit choice
- Found in large systems, e.g., CPEF, DS1

  Function often embedded in acting/monitoring/planning

Actor's deliberation functions

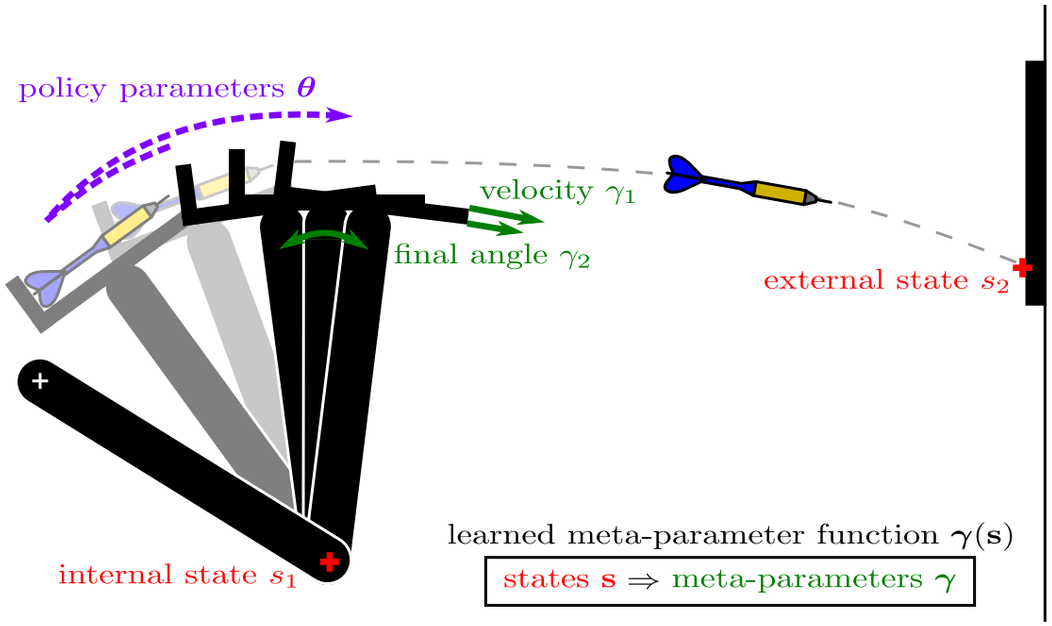


Reinforcement Learning of sensory motor functions

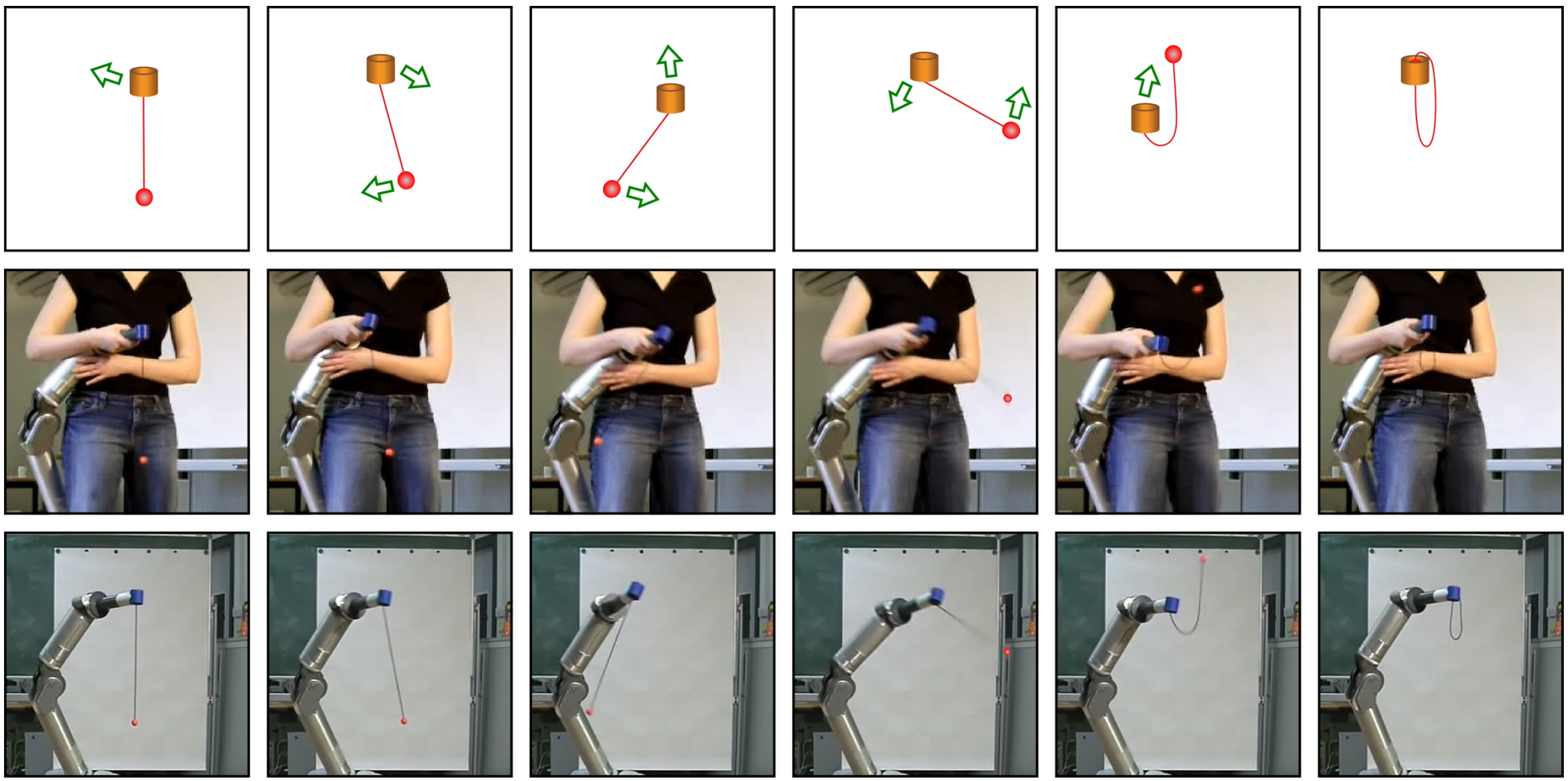


[MPI Bio-Cybernetics, Tübingen]

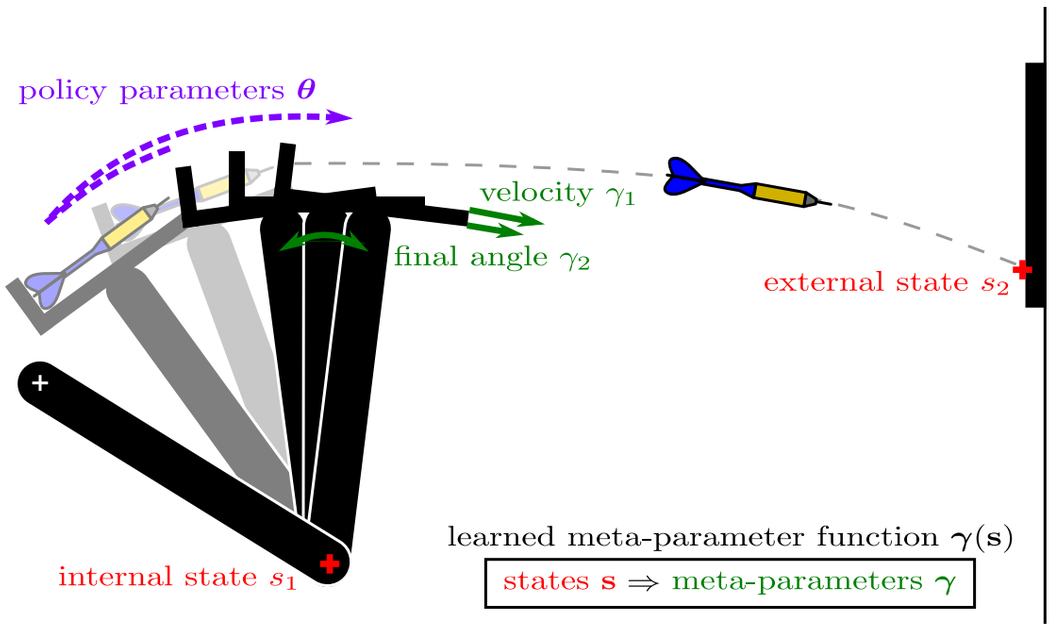
Reinforcement Learning of sensory motor functions



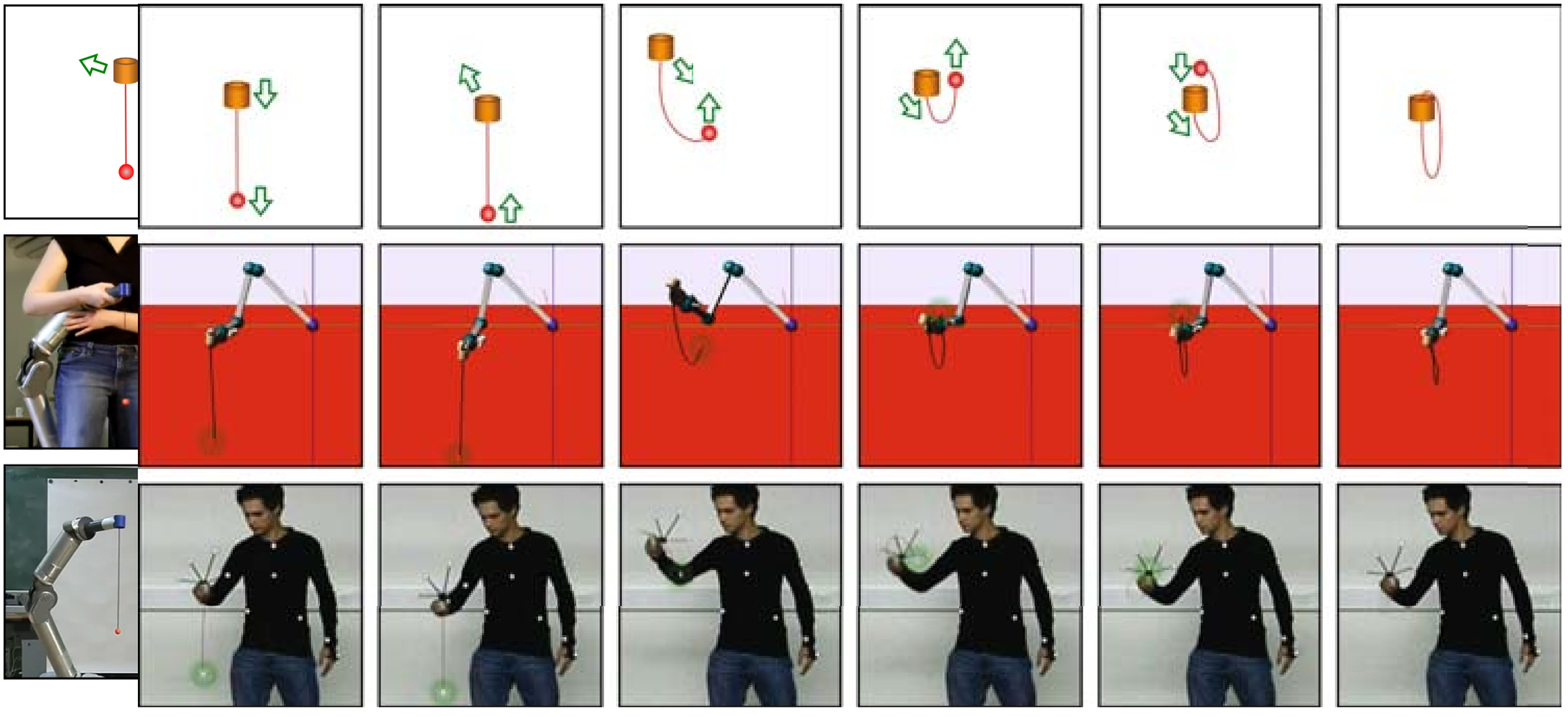
[MPI Bio-Cybernetics, Tübingen]



Reinforcement Learning of sensory motor functions



[MPI Bio-Cybernetics, Tübingen]



Helicopter Aerobatics Apprenticeship Learning

[U. Stanford]

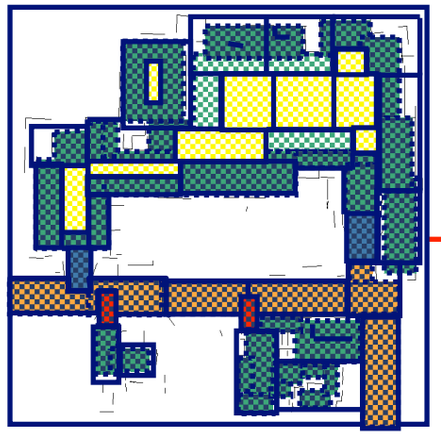
Aerobatics Apprenticeship Learning

Simple linear rigid dynamic models of helicopter

- Learn dynamic models, *one for each type of maneuver*
  - Regression from teacher's demonstrations
  - Improvement by reinforcement learning in autonomous flight
- Learn reference trajectories, one for each aerobatic figure
  - *Expectation-Maximization* on teacher's demonstrations
  - Temporal alignment and optimization
- Learn controllers, one for each aerobatic figure
  - *Differential dynamic programming* continuous MDPs solved by iterative approximation of receding horizon LQR problems

Skill selection learning

Navigation goal



Reports

Markov Decision Process

- Optimal plan
- Receding Horizon Control

S2

Query resource

Resource allocation

Resource Manager

Suspend
Resume

Start

Stop

Extensible set of skills

Skill 1

Skill 2

Skill 3

Skill 4

. . .

action

action

action

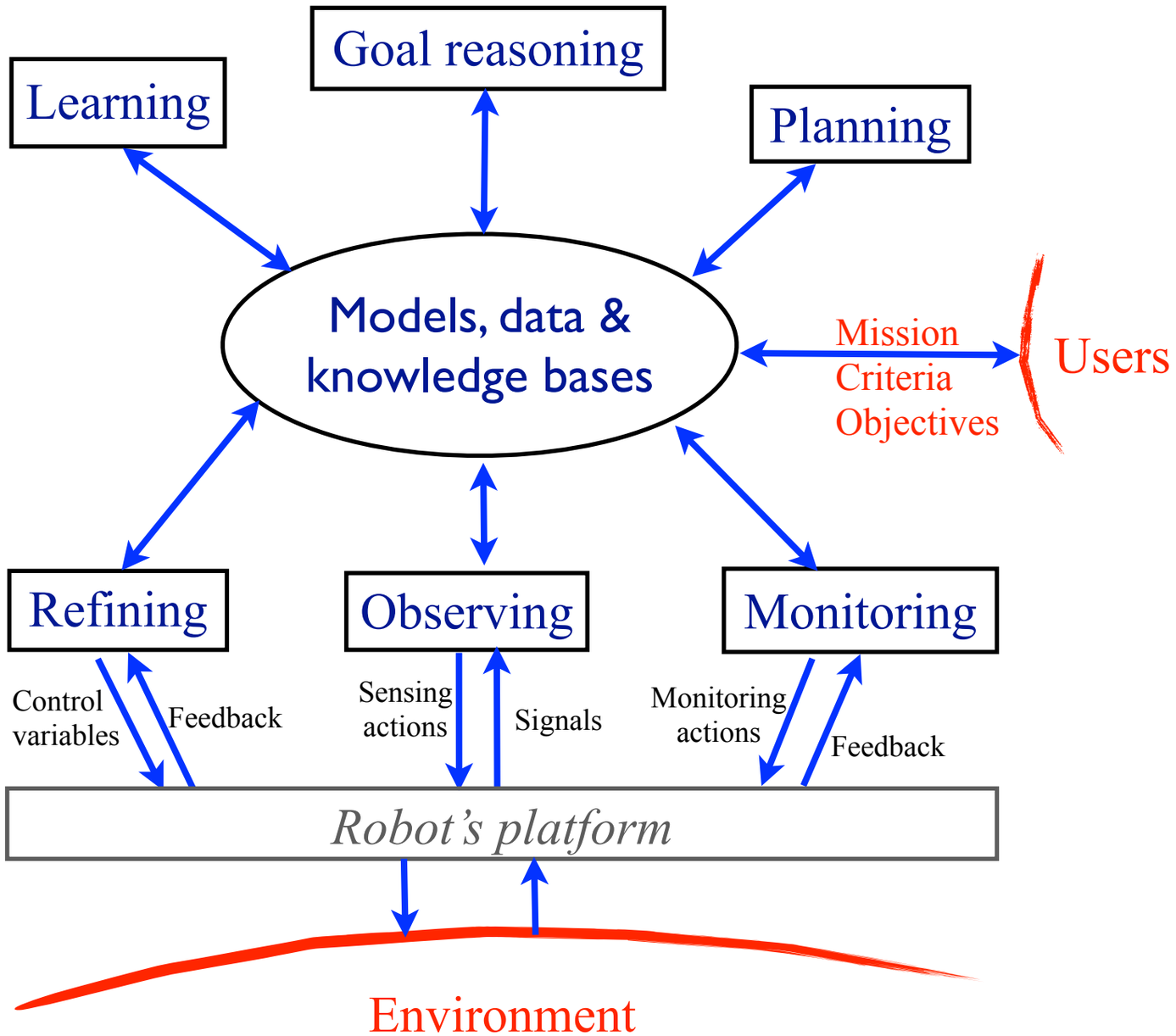
action

Reactive Control System

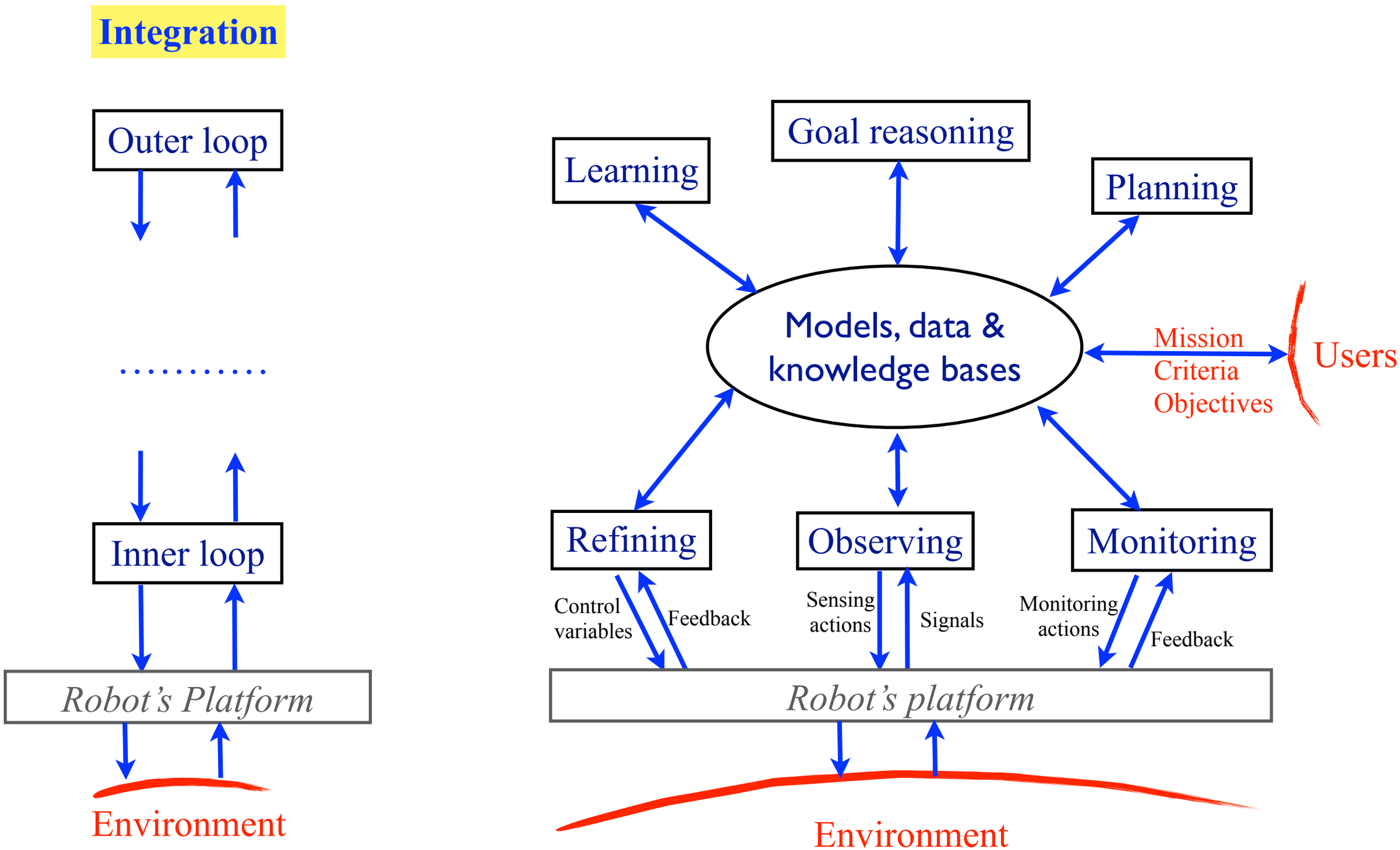
Reports

Sensory-motor commands

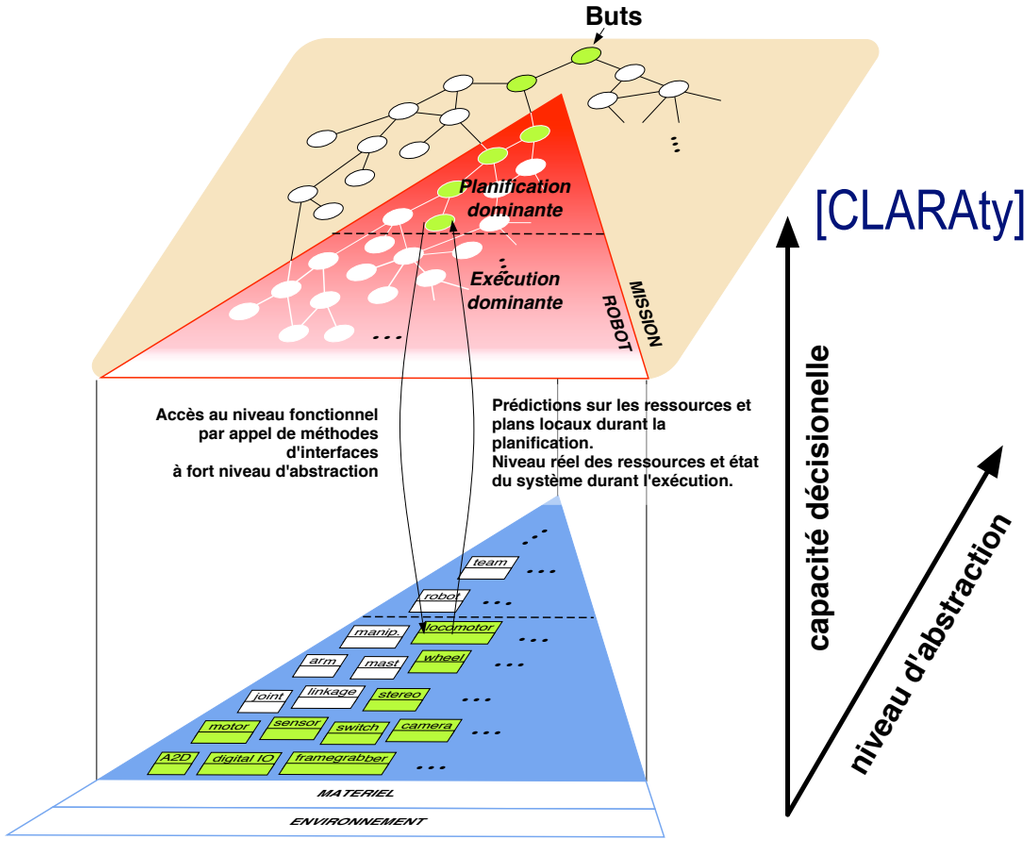
Actor's deliberation functions



Actor's deliberation functions



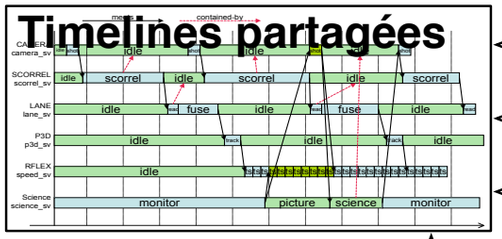
Integration



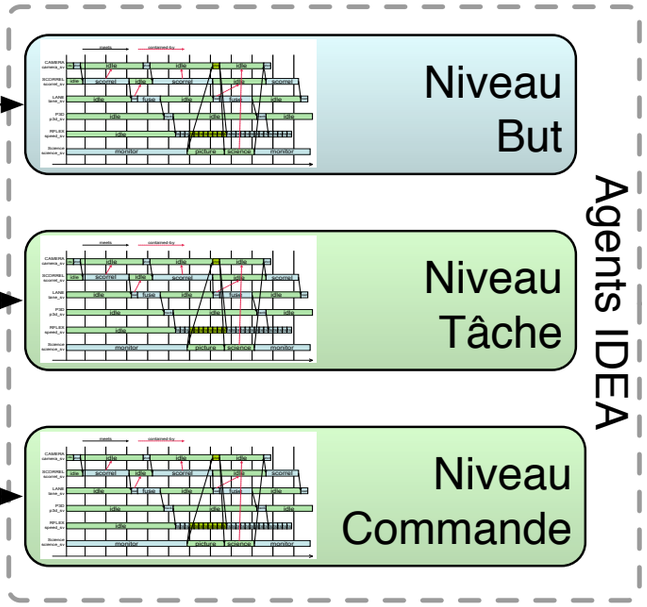
[CLARAty]

capacité décisionelle

niveau d'abstraction

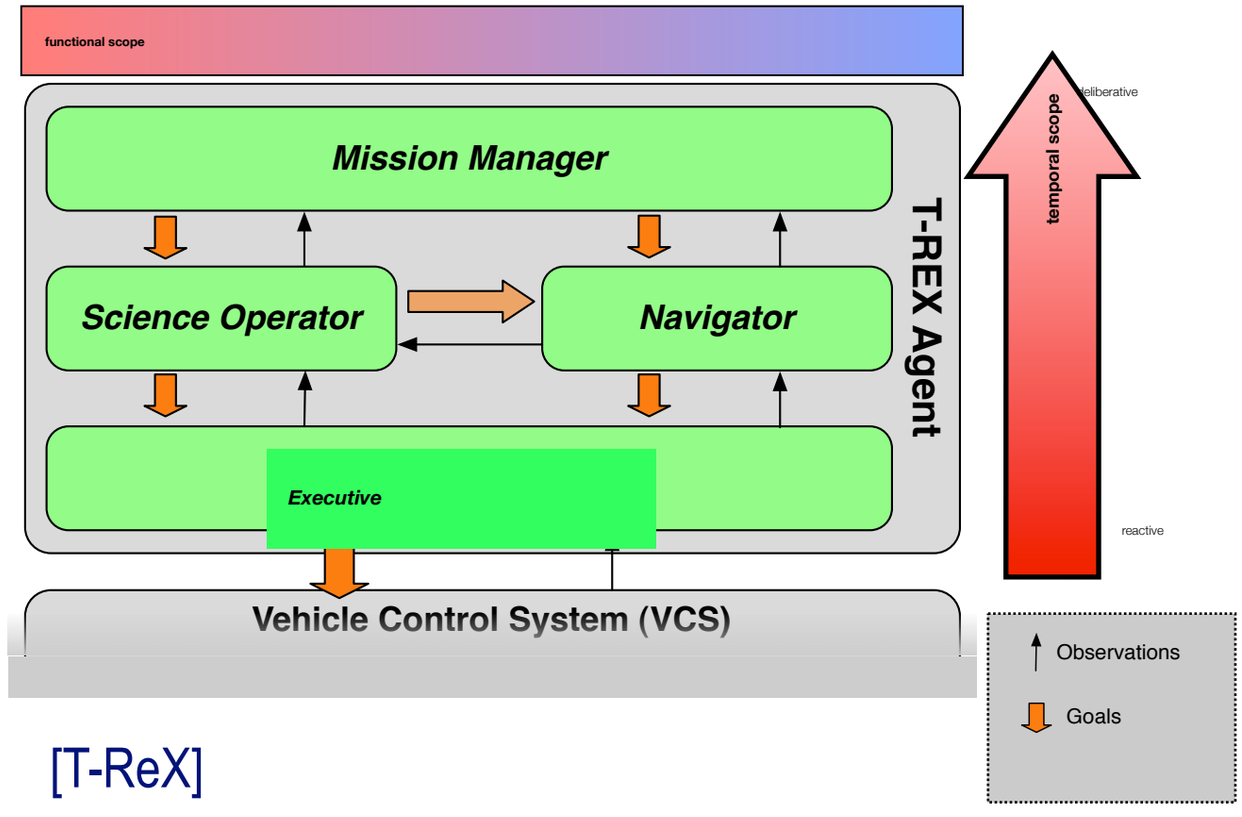


[IDEA]

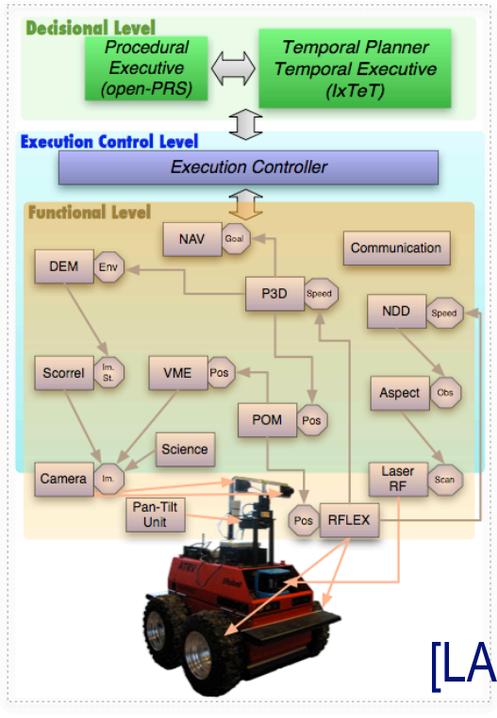


encapsulation IDEA

Flight Software

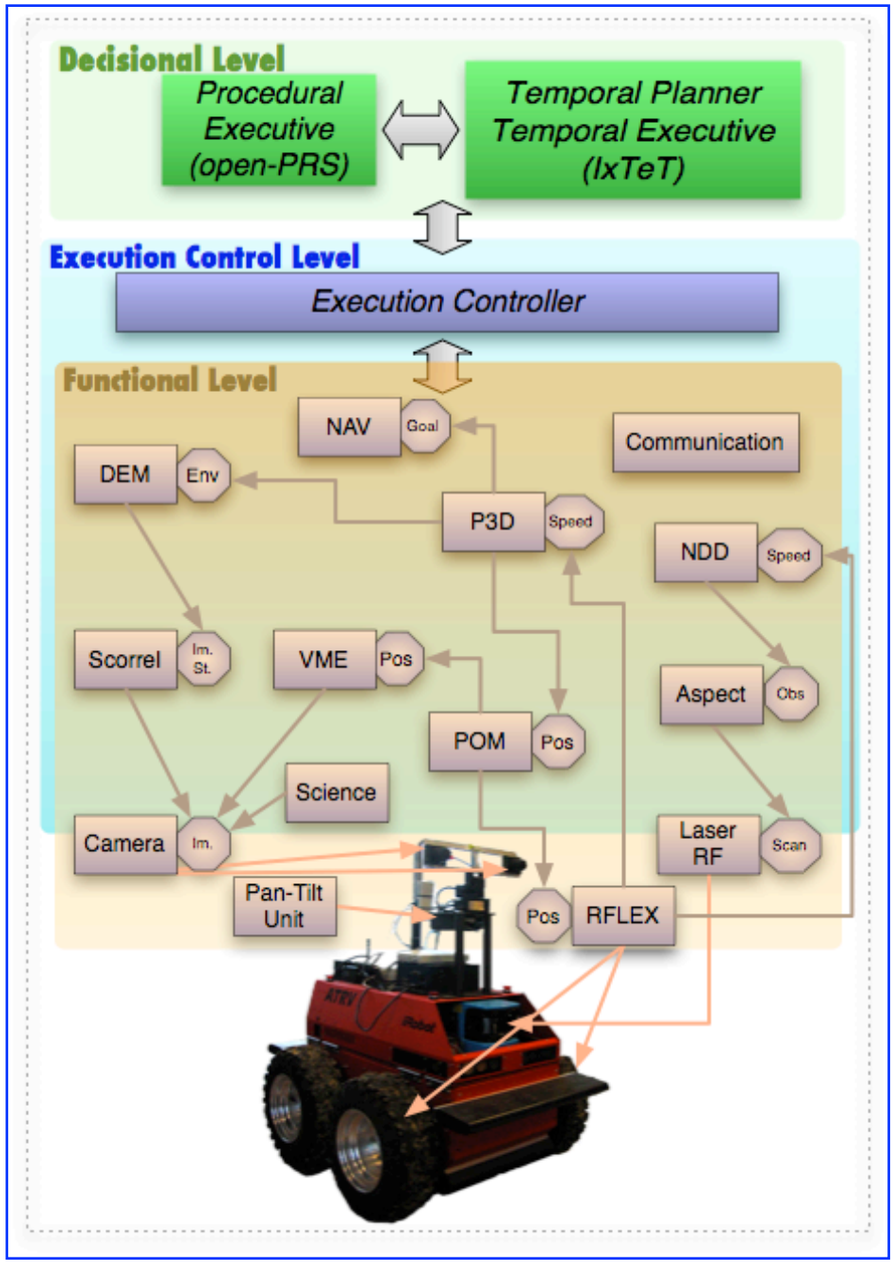


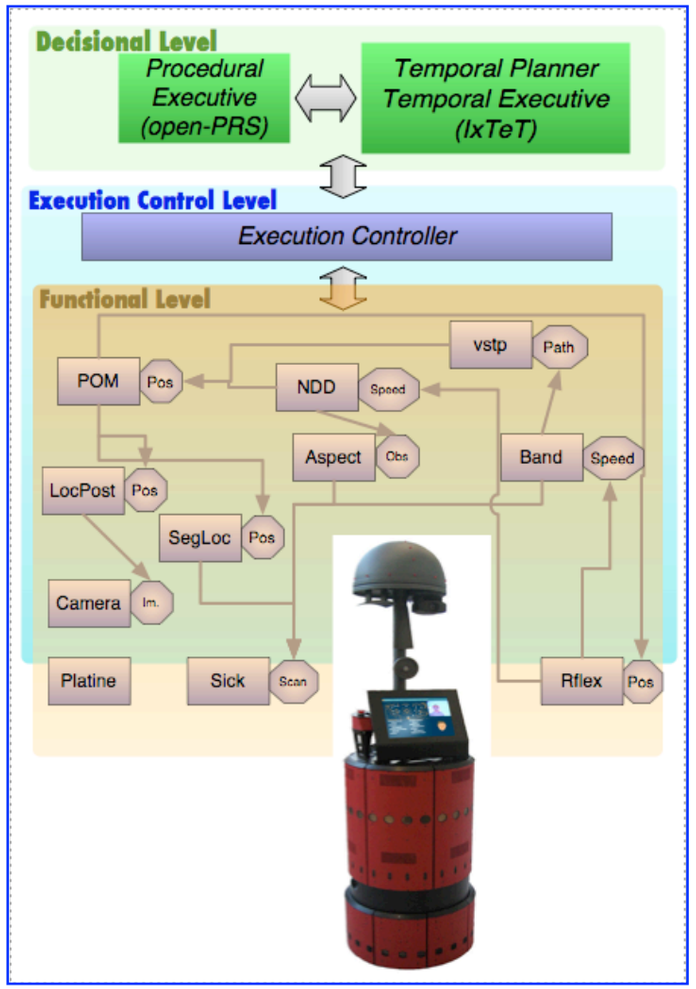
[T-ReX]

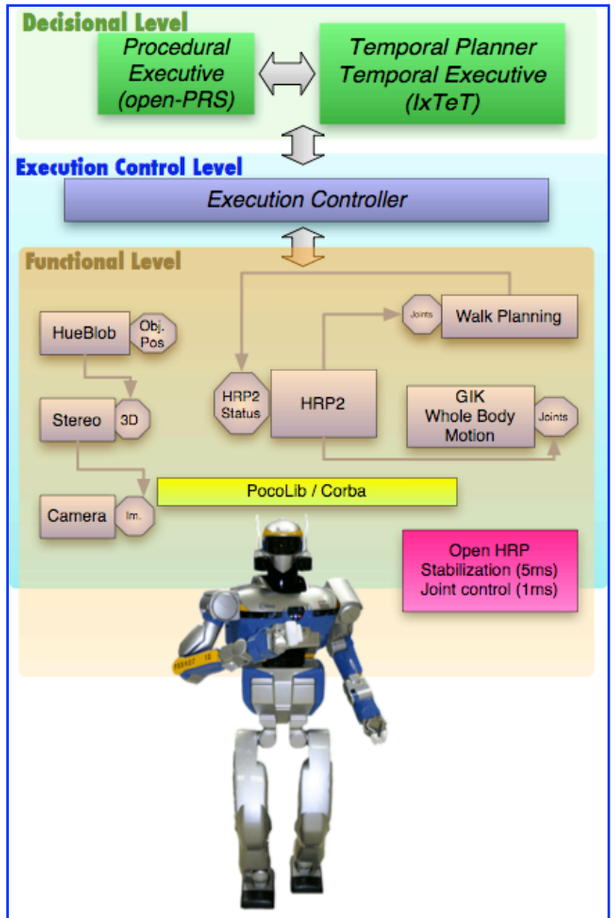


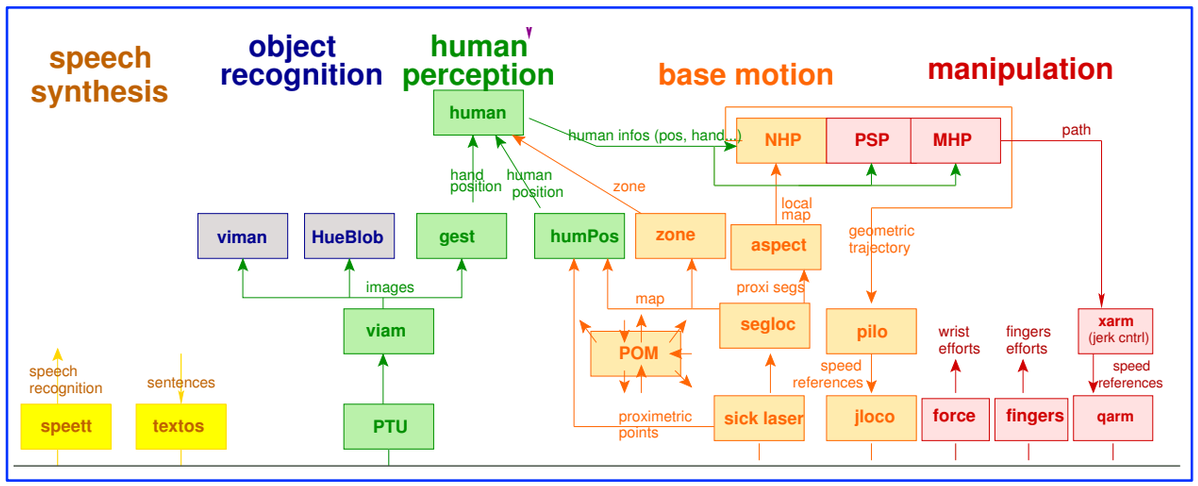
[LAAS]

Integration



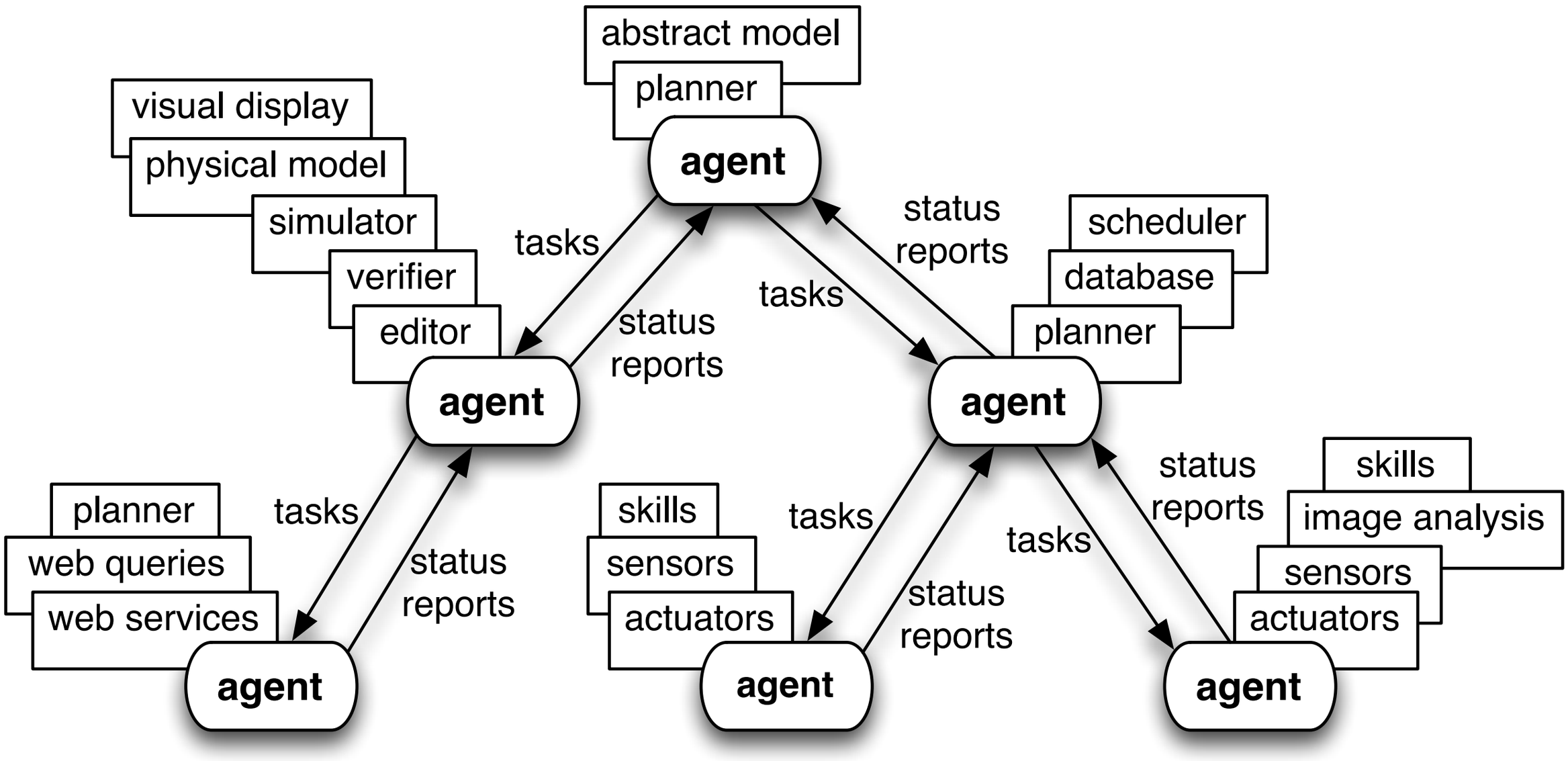






Integration

- Organizational principle: ***actor as a hierarchy of agents***



Outline

✓Motivations

✓Deliberation functions

- Planning
- Refining
- Monitoring
- Perceiving
- Goal reasoning
- Learning
- Integration

- Research Challenges
  - Representation
  - Model acquisition & Verification
  - Synthesis & Refinement
  - Monitoring and Goal reasoning
  - Integration

Representation Challenges

- ***Descriptive models***
  - Map known representations: PDDLx, ANML, RDDL, $\mathcal{K}$
    - Where in the actors hierarchy
    - Link to monitoring, sensing, control
    - Suitability for online plan repair
    - Criticality issues
  - Link to open domain representations: ontologies/DL, e.g.,RoboEarth, OMRKF, ORO, RACE
- ***Operational models***: procedural, automata and graphical
- Relationships between descriptive and operational models
- Simulation and sampling techniques

Model Acquisition, Learning and Verification

- Tools for the specification of descriptive and operational models
- Learning to acquire or improve these models
  - Reinforcement Learning: hierarchical, relation RL, factored MDP
  - Learning from demonstration: teleoperation, external observation
- Verification
  - Hierarchy of actors: of the consistency of their models
  - Heterogeneity of representations
  - Program verification techniques
  - Model checking

Synthesis and refinement

- Online plan synthesis, extension and repair, while acting
- Online skill selection and adaptation
- Integrate temporal dimensions:
  - Time in reasoning about a peculiar resource
  - Time as a computational resource for reasoning
  - Real-time constraints on acting and deliberation
- Planning with sensing and information gathering actions
- Integrate risk and criticality considerations to plan horizon and optimization issues

Monitoring and goal reasoning

- Derive monitoring conditions from descriptive and operational models
- Focus of attention mechanism and link to perceiving for acquiring information needed for monitoring
- Model-based diagnosis for the robot-environment interactions
- Recovery actions and link to criticality analysis issues
- Qualify current goals with respect of longer term objectives and motivations, express reservations and conditions to be monitored
- How to synthesize new goals for current mission
- Map monitoring functions to the actor's hierarchy

Integration

- How to organize actors hierarchy
- Static, i.e., mapped to the robot architecture, or dynamic
- Actor’s enablers, including executors
- Concurrency of actors
- Temporal constraints

Other Deliberation challenges in robotics

- Observing the environment semantics
- Interacting
- Learning
  - Models of the robot and the environment
  - Categories
  - Functions, skills and behaviors
- Architecture
  - Specification
  - Robust adaptation

Other Deliberation challenges in robotics

- Observing the environment semantics
- Interacting
- Learning
  - Models of the robot and the environment
  - Categories
  - Functions, skills and behaviors
- Architecture
  - Specification
  - Robust adaptation

Actor's view of deliberation: numerous challenges
Planning is just the tip of the iceberg