Some Approaches to Learning in Problem Solving

DANIEL BORRAJO MILLÁN

Universidad Carlos III de Madrid
Avda. de la Universidad, 30
28911 Madrid, SPAIN

Teléfono: +(34 91) 624 9459
Email: dborrajo@ia.uc3m.es
Web: http://scalab.uc3m.es/~dborrajo

Some Approaches to Learning in Problem Solving

1. **Introduction**
2. Review of learning in problem solving
3. Hybrid learning. HAMLET
4. Learning by genetic programming. EVOCK
5. Mixed initiative
6. Discussion

Problem solving

Many issues to consider...

- How to **represent** the action model?
- What is the (sufficient) **initial state** of the world?
- What are the (prioritized) **goals**?
- How to **acquire domain knowledge** efficiently from human experts?
- How to **acquire control knowledge** efficiently?
- Which **algorithm** to use to generate the solution?
- How to generate solutions in a computationally **tractable** way?
- How to create solutions of **good** quality?
- How to **scale up** to real-world problems?

Our focus

- How to represent the action model?
- What is the (sufficient) initial state of the world?
- What are the (prioritized) goals?
- How to acquire domain knowledge efficiently from human experts?
- **How to acquire control knowledge efficiently?**
- Which algorithm to use to generate the solution?
- **How to generate solutions in a computationally tractable way?**
- **How to create solutions of good quality?**
- **How to scale up to real-world problems?**

Approaches

- Knowledge engineering (domain dependent):
  - ⋆ Handcode and refine domain knowledge
  - ⋆ Specify control strategies
  - ⋆ Define knowledge to produce quality solutions
- Machine learning (domain independent):
  - ⋆ **Automate** the interpretation of the problem solving experience into reusable task knowledge: domain model and heuristics (control)
  - ⋆ Most recent focus: combine with user's input (mixed initiative, domain dependent)

Motivation

- Even current very fast problem solvers have **not solved yet** the complete problem solving task
- Current focus (specially in planning) on **quality-based** solutions
- Definition of cost functions does not imply the problem solver knows for each problem and domain how to obtain **better** solutions **efficiently**
- The problem is **much harder than just satisficing**
- **Control knowledge** can be used, but
  - ⋆ it is very **difficult to define** the right knowledge
  - ⋆ defining such knowledge implies a **strong knowledge** on the problem solving technique
  - ⋆ since metrics can vary over time/users/domain, heuristics must be **continuously maintained**

Some Approaches to Learning in Problem Solving

1. Introduction
2. **Review of learning in problem solving**
3. Hybrid learning. HAMLET
4. Learning by genetic programming. EVOCK
5. Mixed initiative
6. Discussion

Learning Opportunities

- **Domain models**
  - ⋆ abstraction hierarchies
  - ⋆ domain operators
  - ⋆ state invariants
- **Control knowledge (heuristics)**
  - ⋆ choices at decision points
    - ∗ state-space planning: goal, operator, bindings, apply/subgoal
    - ∗ plan-space planning: operator, threat, threat resolution
    - ∗ hierarchical planning (HTN): method
    - ∗ heuristic search-based planning: goal, operator, bindings
    - ∗ graph-based planning: operator-bindings, 1-phase termination
  - ⋆ evaluation functions
  - ⋆ policies (MDP)
  - ⋆ complete problem solving episodes (cases)
  - ⋆ efficiency vs. solution quality

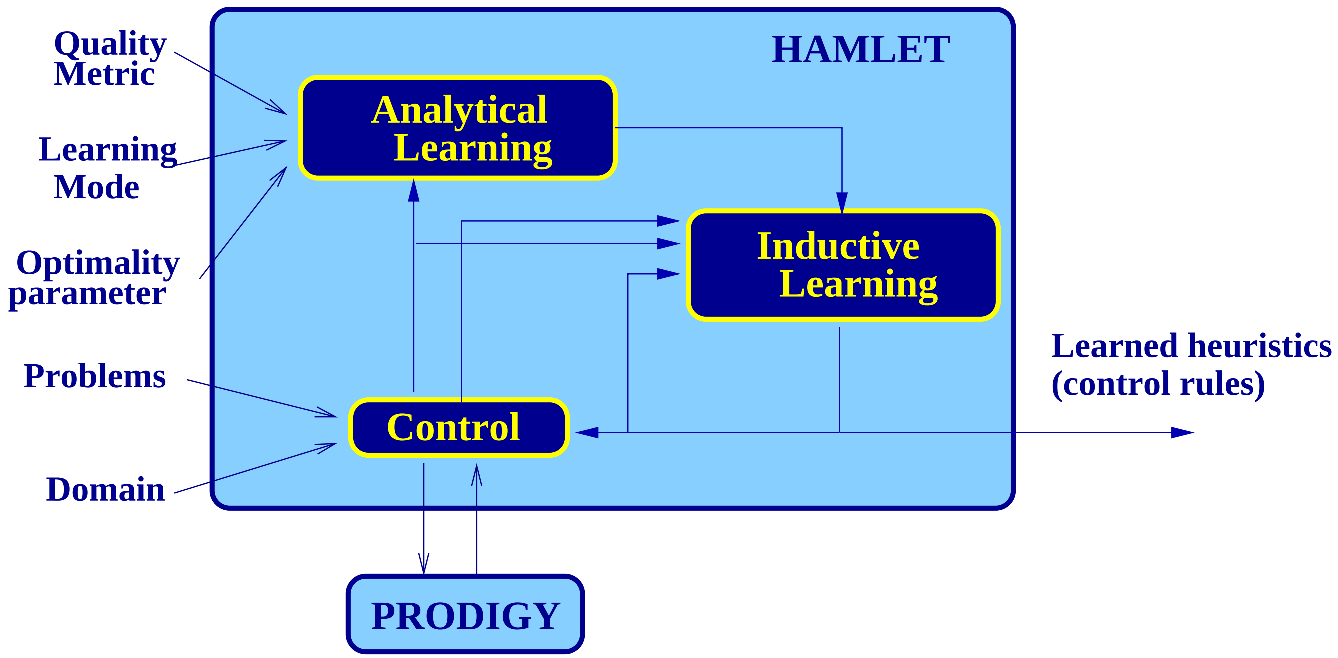
Learning Systems in Planning

- **Linear**: STRIPS [Fikes *et al.*, 1972], Rubik's cube [Korf, 1985], PRODIGY/EBL [Minton, 1988], STATIC [Etzioni, 1990], DYNAMIC [Pérez and Etzioni, 1992], ALPINE [Knoblock, 1991], GRASSHOPER [Leckie and Zukerman, 1998], LEX [Mitchell *et al.*, 1983], ACM [Langley, 1983], LEBL [Tadepalli, 1989], EXPERIMENTER [Carbonell and Gil, 1990]
- **Nonlinear**: SOAR [Laird *et al.*, 1986], FAILSAFE [Bhatnagar, 1992], OBSERVE [Wang, 1994], COMPOSER [Gratch and DeJong, 1992], PRIAR [Kambhampati, 1989], SNLP+EBG [Kambhampati and Kedar, 1991], SNLP+EBL [Katukam and Kambhampati, 1994], UCPOP+EBL [Qu and Kambhampati, 1995], QUALITY [Pérez and Carbonell, 1994], SteppingStone [Ruby and Kibler, 1992], UCPOP+FOIL [Estlin and Mooney, 1995], PRODIGY/ANALOGY [Veloso, 1994], HAMLET [Borrajo and Veloso, 1997], EVOCK [Aler *et al.*, 2001, Aler *et al.*, 2002], GRAPHPLAN+EBL [Kambhampati, 1999], SATPLAN+FOIL [Huang *et al.*, 2000], generalized policies [Khardon, 1999, Martín and Geffner, 2000]
- **MDP models:** reinforcement learning [Kaelbling *et al.*, 1996], Q-LEARNING [Watkins and Dayan, 1992], temporal differences [Sutton, 1988, Tesauro, 1992], LOPE [García-Martínez and Borrajo, 2000]

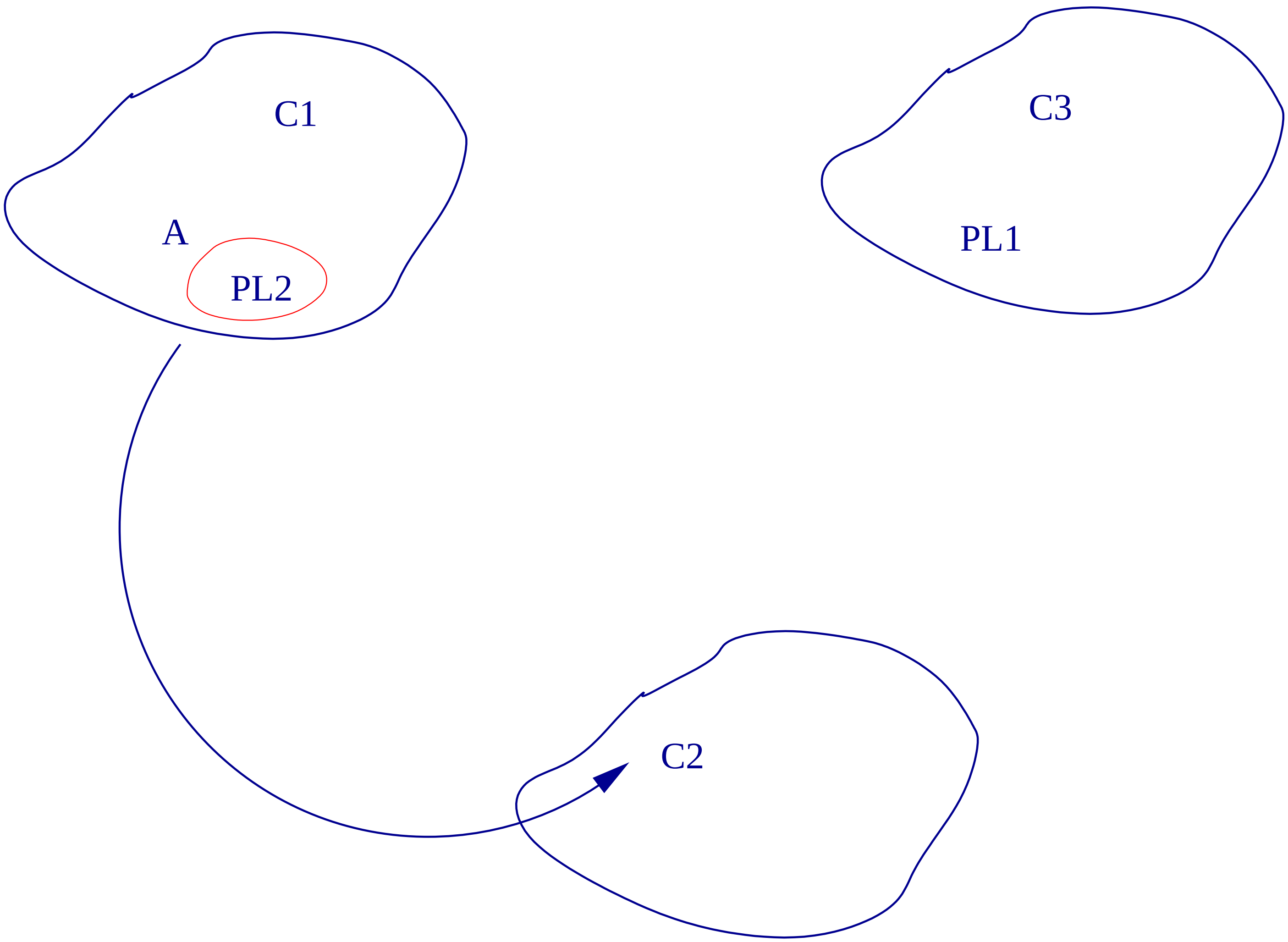
Some Approaches to Learning in Problem Solving

1. Introduction
2. Review of learning in problem solving
3. **Hybrid learning.** HAMLET
4. Learning by genetic programming. EVOCK
5. Mixed initiative
6. Discussion

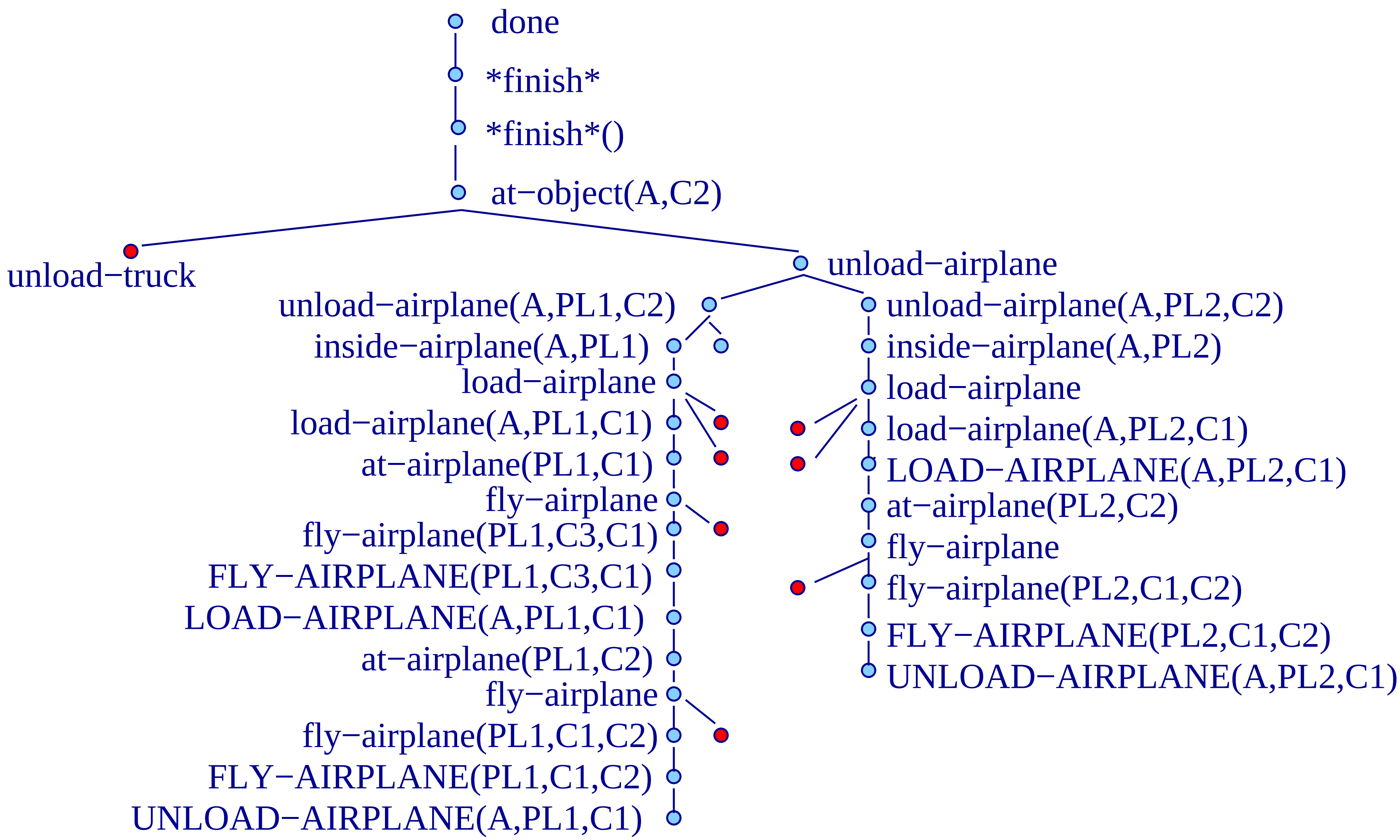
Learning for quality. HAMLET



Example of logistics problem

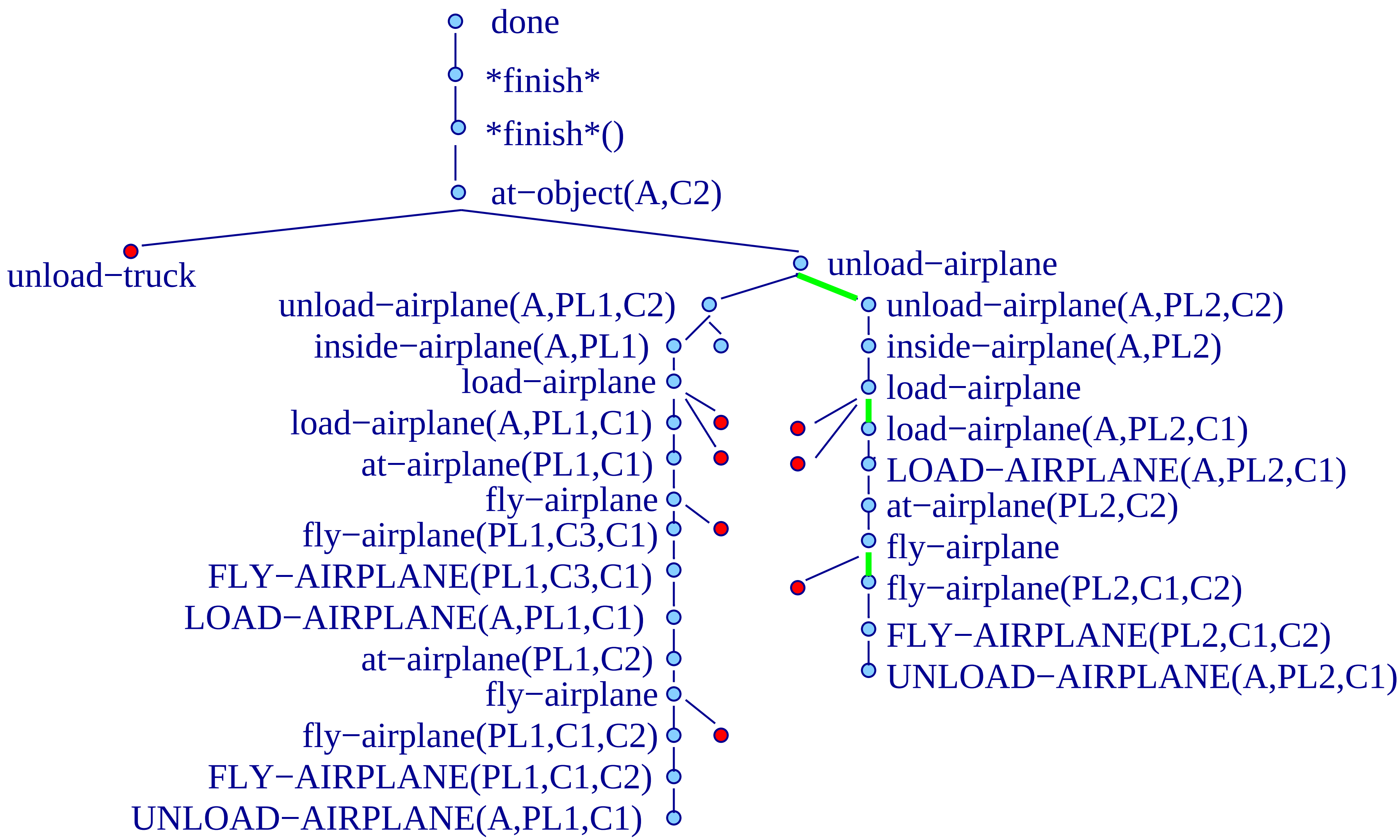


Example of search tree



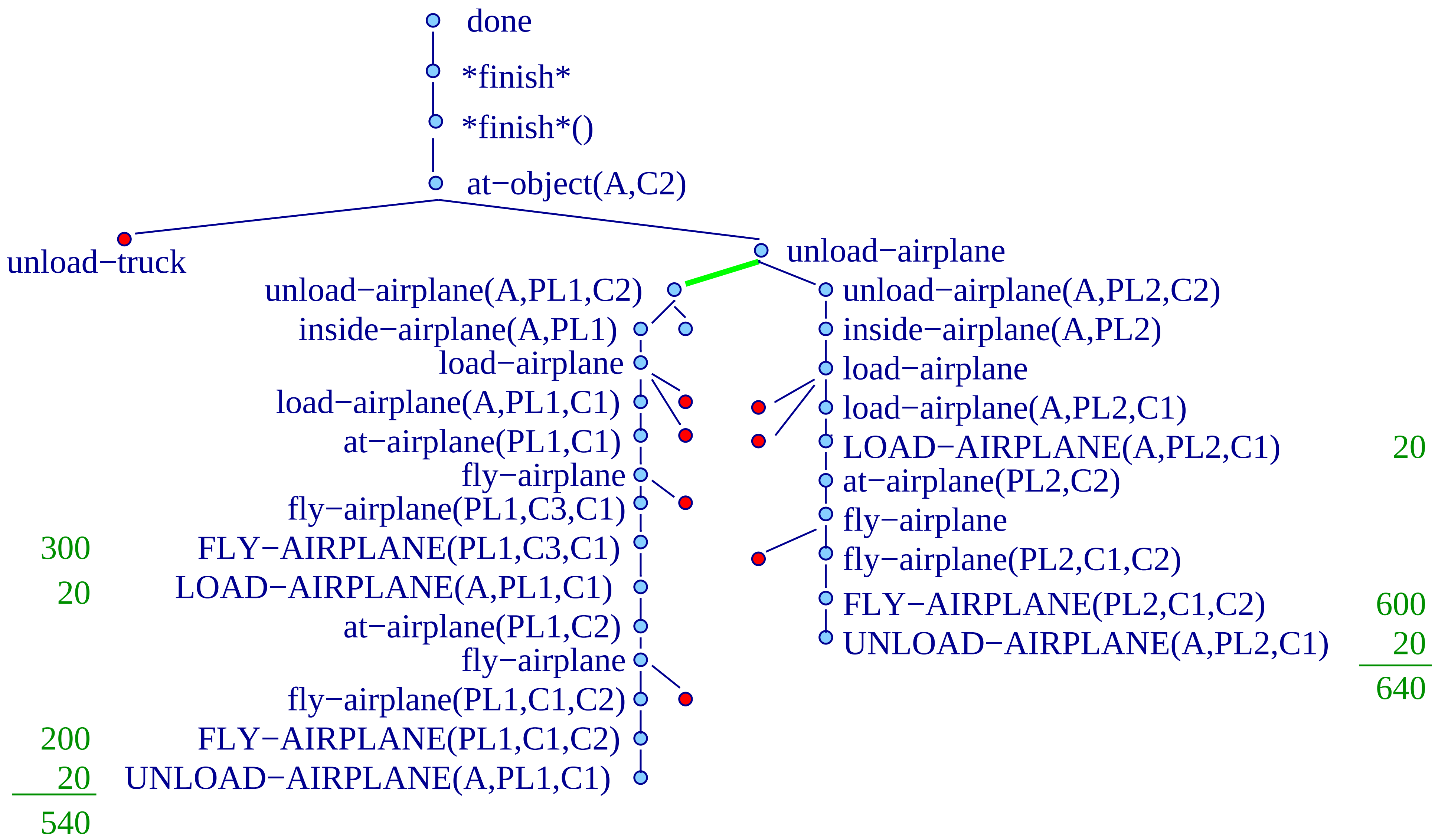
Hybrid Learning. HAMLET

Learning for plan length



Hybrid Learning. HAMLET

Learning for quality



Example of control rule

```
(control-rule select-operators-unload-airplane
    (if (current-goal (at <object> <location1>))
        (true-in-state (at <object> <location2>))
        (true-in-state (loc-at <location1> <city1>))
        (true-in-state (loc-at <location2> <city2>))
        (different-vars-p)
        (type-of-object <object> object)
        (type-of-object <location1> airport))
    (then select operator unload-airplane))
```

Difficulties:

- variables have to be bound to different values (cities)
- constants have to be of a specific type (`object` and `location1`)
- there are conditions that do not relate to the goal regression (`loc-at`)

Problems with HAMLET

- It does not always generate more accurate control knowledge by observing more and more examples
- Causes
  - ⋆ Incrementality
  - ⋆ Generalization and specialization procedures
  - ⋆ It learns from simple problems search trees, preferably fully expanded
  - ⋆ It depends very much on the training examples (inductive method)
  - ⋆ Reduced language for describing control rules
- But, it provides a very good starting point for another type of learner (machine or human)

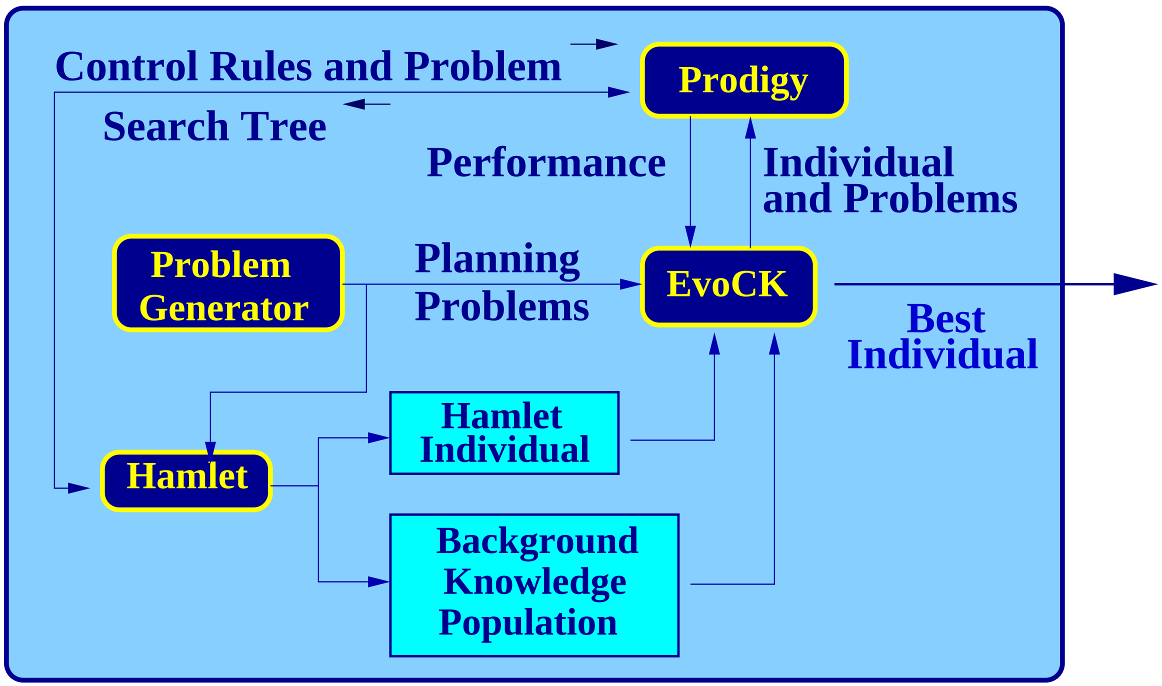
Some Approaches to Learning in Problem Solving

1. Introduction
2. Review of learning in problem solving
3. Hybrid learning. HAMLET
4. **Learning by genetic programming. EVOCK**
5. Mixed initiative
6. Discussion

Relevant characteristics of GP

- Learning operators (mutation and crossover) do not require examples to modify candidate hypotheses
- It is not limited by significant examples being rare
- Evaluation function can consider many types of features
- It can be guided by an initial population and an auxiliary population for informed crossover
- It considers all problems at once
- It is based on an explicit grammar

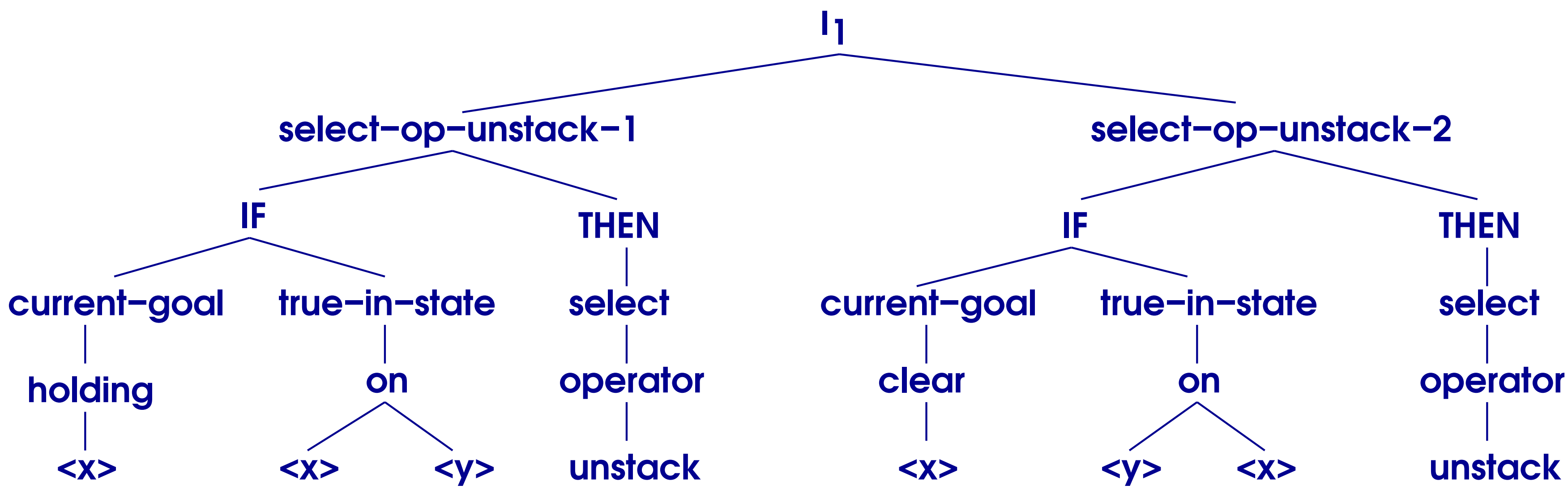
EvoCK architecture



Genetic Programming of control knowledge. EvoCK

- Grammar-based
- Individual: set of control rules
- Genetic operators
  - ⋆ Crossover (standard, informed, adding)
  - ⋆ Mutation (standard, removing, adding)
  - ⋆ Specific (renaming variables, generalization)
- Fitness function
  - ⋆ Completeness
    - ∗ Number of solved problems
    - ∗ Number of solved problems better than PRODIGY
  - ⋆ Generality
  - ⋆ Compactness

Example of an individual

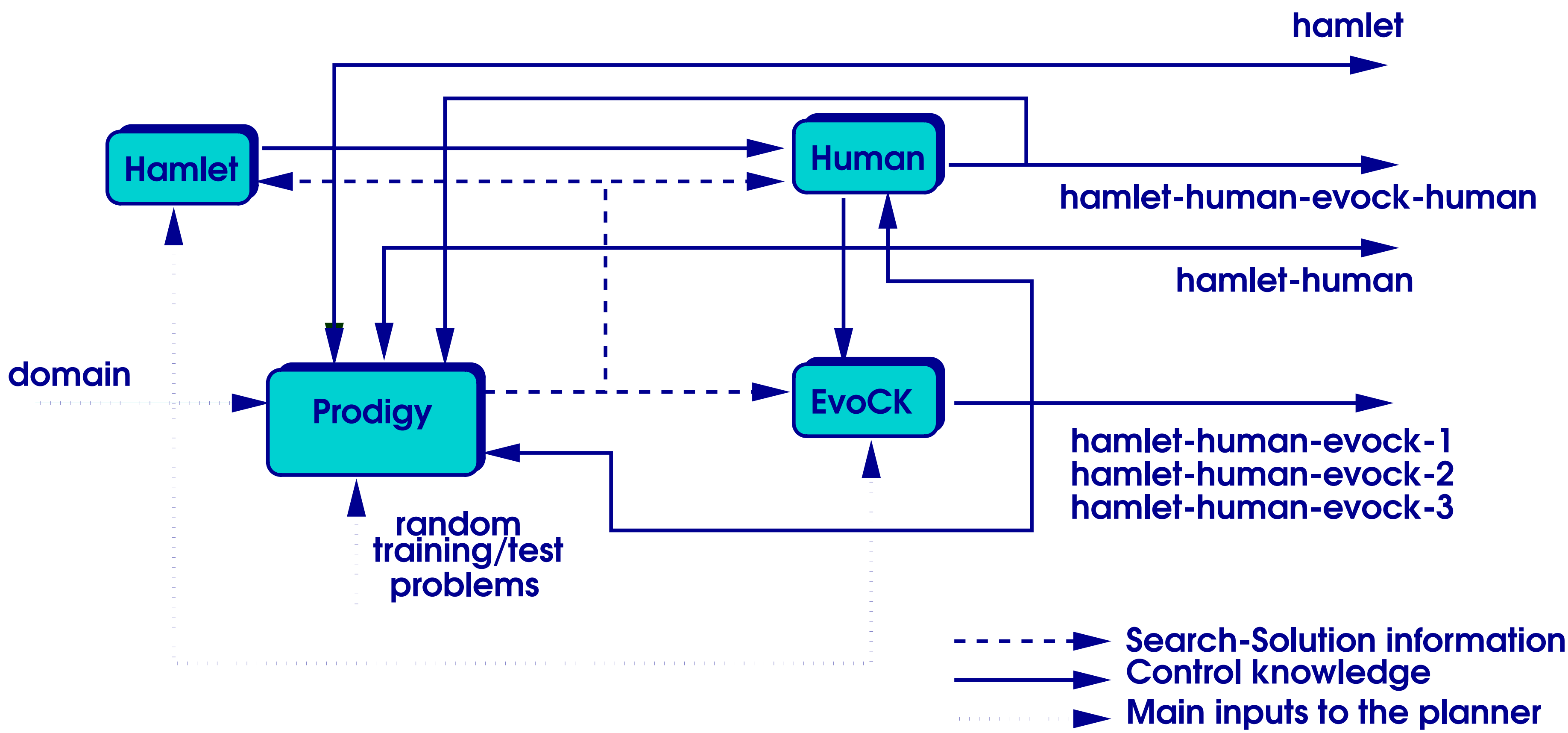


Some Approaches to Learning in Problem Solving

1. Introduction
2. Review of learning in problem solving
3. Hybrid learning. HAMLET
4. Learning by genetic programming. EVOCK
5. **Mixed initiative**
6. Discussion

Mixed initiative

HAMLET-HUMAN-EVOCK-HUMAN



Results experiment 1. Logistics

Planner	Solved problems	Time	Plan length	Number of rules
PRODIGY4.0	0	3600	-	-
HAMLET	0	3600	-	32
HUMAN	36	63.43	2578	35
HAMLET-HUMAN	36	268.12	2604	39
HAMLET-HUMAN-EVOCK-1	36	84.27	3111	18
HAMLET-HUMAN-EVOCK-2	36	255.41	2585	40
HAMLET-HUMAN-EVOCK-3	36	79.79	3051	18
HAMLET-HUMAN-EVOCK-HUMAN	36	74.39	3051	18
TALPLANNER	36	9.53	2329	
PBR	29	1089.58	2409	
SHOP	36	14.93	2236	
R	36	39.22	4133	

Some Approaches to Learning in Problem Solving

1. Introduction
2. Review of learning in problem solving
3. Hybrid learning. HAMLET
4. Learning by genetic programming. EVOCK
5. Mixed initiative
6. **Discussion**

Discussion

- Learning techniques should reflect somehow the way by which decisions are made by the problem solver
- The knowledge about how to make a decision should be made explicit in the state
- The base problem solver should be able to solve training problems somehow
- If quality is important, it should also provide at least two different-quality solutions
- If a learning technique acquires individual control knowledge, the decisions should be reproducible to be of use (utility problem)
- Learning in problem solving also needs to worry about representativeness of examples; in fact more than usual inductive techniques
- It is difficult to add negative constrains (and quantification) to the rules
- Combining machine learning and humans is a very effective approach

More recent and future work

- Learning in more recent problem solvers
- Learning for multiple criteria
- Using numerical predicates on conditions of control rules
- Active learning: on-line generation of appropriate training problems
- Use other planners as guidance
- Learning for planning and scheduling
- Learning for HTN+POP planners

References

[Aler *et al.*, 2001] Ricardo Aler, Daniel Borrajo, and Pedro Isasi. Learning to solve planning problems efficiently by means of genetic programming. *Evolutionary Computation*, 9(4):387–420, 2001.

[Aler *et al.*, 2002] Ricardo Aler, Daniel Borrajo, and Pedro Isasi. Using genetic programming to learn and improve control knowledge. *Artificial Intelligence*, 141(1-2):29–56, October 2002.

[Bhatnagar, 1992] Neeraj Bhatnagar. *On-line learning from search failures*. PhD thesis, Rutgers University, 1992.

[Borrajo and Veloso, 1997] Daniel Borrajo and Manuela Veloso. Lazy incremental learning of control knowledge for efficiently obtaining quality plans. *AI Review Journal. Special Issue on Lazy Learning*, 11(1-5):371–405, February 1997. Also in the book "Lazy Learning", David Aha (ed.), Kluwer Academic Publishers, May 1997, ISBN 0-7923-4584-3.

[Carbonell and Gil, 1990] Jaime G. Carbonell and Yolanda Gil. Learning by experimentation: The operator refinement method. In R. S. Michalski and Y. Kodratoff, editors, *Machine Learning: An Artificial Intelligence Approach, Volume III*, pages 191–213. Morgan Kaufmann, Palo Alto, CA, 1990.

[Estlin and Mooney, 1995] Tara A. Estlin and Raymond Mooney. Hybrid learning of search control for partial order planning. Technical report, University of Texas, 1995.

[Etzioni, 1990] Oren Etzioni. *A Structural Theory of Explanation-Based Learning*. PhD thesis, School of Computer Science, Carnegie Mellon University, 1990. Available as technical report CMU-CS-90-185.

[Fikes *et al.*, 1972] Richard E. Fikes, P. E. Hart, and Nils J. Nilsson. Learning and executing generalized robot plans. *Artificial Intelligence*, 3:251–288, 1972.

[García-Martínez and Borrajo, 2000] Ramón García-Martínez and Daniel Borrajo. An integrated approach of learning, planning, and execution. *Journal of Intelligent and Robotic Systems*, 29(1):47–78, September 2000.

[Gratch and DeJong, 1992] Jonathan Gratch and Gerald DeJong. COMPOSER: A probabilistic solution to the utility problem in speed-up learning. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 235–240, 1992.

[Huang *et al.*, 2000] Yi-Cheng Huang, Bart Selman, and Henry Kautz. Learning declarative control rules for constraint-based planning. In Pat Langley, editor, *Proceedings of the Seventeenth International Conference on Machine Learning, ICML'00*, Stanford, CA (USA), June-July 2000.

[Kaelbling *et al.*, 1996] Lelie Pack Kaelbling, Michael L. Littman, and Andrew W. Moore. Reinforcement learning: A survey. *International Journal of Artificial Intelligence Research*, pages 237–285, 1996.

[Kambhampati and Kedar, 1991] Subbarao Kambhampati and Smadar Kedar. Explanation based generalization of partially ordered plans. In *Proceedings of the Ninth National Conference on Artificial Intelligence*, pages 679–685, Anaheim, CA, 1991. AAAI Press.

[Kambhampati, 1989] Subbarao Kambhampati. *Flexible Reuse and Modification in Hierarchical*

Planning: A Validation Structure Based Approach. PhD thesis, Computer Vision Laboratory, Center for Automation Research, University of Maryland, College Park, MD, 1989.

[Kambhampati, 1999] Subbarao Kambhampati. Improving graphplan's search with ebl & ddb techniques. In Thomas Dean, editor, *Proceedings of the IJCAI'99*, pages 982–987, Stockholm, Sweden, July-August 1999. Morgan Kaufmann Publishers.

[Katukam and Kambhampati, 1994] Suresh Katukam and Subbarao Kambhampati. Learning explanation-based search control rules for partial order planning. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pages 582–587, Seattle, WA, 1994. AAAI Press.

[Khardon, 1999] Roni Khardon. Learning action strategies for planning domains. *Artificial Intelligence*, 113(1-2):125–148, 1999.

[Knoblock, 1991] Craig A. Knoblock. *Automatically Generating Abstractions for Problem Solving*. PhD thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 1991. Available as technical report CMU-CS-91-120.

[Korf, 1985] Richard E. Korf. Macro-operators: A weak method for learning. *Artificial Intelligence*, 26:35–77, 1985.

[Laird *et al.*, 1986] John E. Laird, Paul S. Rosenbloom, and Allen Newell. Chunking in SOAR: The anatomy of a general learning mechanism. *Machine Learning*, 1:11–46, 1986.

[Langley, 1983] Pat Langley. Learning effective search heuristics. In *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*, pages 419–421, Los Altos, CA, 1983. Morgan Kaufmann.

[Leckie and Zukerman, 1998] Christopher Leckie and Ingrid Zukerman. Inductive learning of search control rules for planning. *Artificial Intelligence*, 101(1–2):63–98, May 1998.

[Martín and Geffner, 2000] Mario Martín and Héctor Geffner. Learning generalized policies in planning using concept languages. In *Proceedings of the 7th Int. Conf. on Knowledge Representation and Reasoning (KR 2000)*, Colorado, 2000. Morgan Kaufmann.

[Minton, 1988] Steven Minton. *Learning Effective Search Control Knowledge: An Explanation-Based Approach*. Kluwer Academic Publishers, Boston, MA, 1988.

[Mitchell *et al.*, 1983] Tom M. Mitchell, Paul E. Utgoff, and R. B. Banerji. Learning by experimentation: Acquiring and refining problem-solving heuristics. In *Machine Learning, An Artificial Intelligence Approach*. Tioga Press, Palo Alto, CA, 1983.

[Pérez and Carbonell, 1994] M. Alicia Pérez and Jaime G. Carbonell. Control knowledge to improve plan quality. In *Proceedings of the Second International Conference on AI Planning Systems*, pages 323–328, Chicago, IL, 1994. AAAI Press, CA.

[Pérez and Etzioni, 1992] M. Alicia Pérez and Oren Etzioni. DYNAMIC: A new role for training problems in EBL. In *Proceedings of the Ninth International Conference on Machine Learning*, pages 367–372. Morgan Kaufmann, Aberdeen, Scotland, 1992.

[Qu and Kambhampati, 1995] Yong Qu and Subbarao Kambhampati. Learning search control rules for plan-space planners: Factors affecting the performance. Technical report, Arizona State University, February 1995.

[Ruby and Kibler, 1992] David Ruby and Dennis Kibler. Learning episodes for optimization. In

Proceedings of the Ninth International Conference on Machine Learning, pages 379–384, Aberdeen, Scotland, 1992. Morgan Kaufmann.

[Sutton, 1988] Richard Sutton. Learning to predict by the methods of temporal differences. *Machine Learning*, 3(1):9–44, August 1988.

[Tadepalli, 1989] Prasad Tadepalli. Lazy explanation-based learning: A solution to the intractable theory problem. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 694–700, San Mateo, CA, 1989. Morgan Kaufmann.

[Tesauro, 1992] Gerald Tesauro. Practical issues in temporal difference learning. *Machine Learning*, 8(3/4):257–277, May 1992.

[Veloso, 1994] Manuela Veloso. *Planning and Learning by Analogical Reasoning*. Springer Verlag, December 1994.

[Wang, 1994] Xuemei Wang. Learning planning operators by observation and practice. In *Proceedings of the Second International Conference on AI Planning Systems, AIPS-94*, pages 335–340, Chicago, IL, June 1994. AAAI Press, CA.

[Watkins and Dayan, 1992] C. J. C. H. Watkins and P. Dayan. Technical note: Q-learning. *Machine Learning*, 8(3/4):279–292, May 1992.