Theoretical and Real-Life Reinforcement Learning

Leona F. Fass

P.O. Box 2914
Carmel CA 93921
lff4 [AT] cornell [DOT] edu

Abstract

We describe processes by which a theoretical learner or system may acquire knowledge through feedback, reinforcement and adaptation. We then discuss applications to real-life adaptation and learning in such areas as link analysis; reactive agent and multiagent system design; user interfaces; and development of the Semantic Web. Instances include real-life reinforcement in e-environment modeling; e-process assurance; adaptive system synthesis; assistive technology; and design of communicating agents to navigate the Web.

Keywords: AI, software design, HCI, agents

1. Introduction

Reinforcement learning has come to mean determination of actions that approach an optimal behavior, perhaps to be exhibited by agents, based on observations of real environments [9]. But techniques we initially devised for learning by reinforcement and adaptation in formal problem domains can also be applied to such learning problems of real life.

Our original research in learning and adaptive systems involved development of computational techniques for the acquisition of knowledge, and we considered problem domains where a body of knowledge to-be-acquired could constantly grow and change. We needed to determine the means by which *representative* knowledge could be conveyed to a system so that a characterizing model of knowledge could be acquired. We wanted the learning process that produced the knowledge model to effect learning adaptively, so that when observed knowledge changed so could the model. In the best case, should a new model need be found, we wished it to be discovered monotonically.

We did find that in a constrained formal language domain a system could produce perfect knowledge models and thus, learn perfectly. Autonomous learning could occur from observations, but it could occur more efficiently with some interaction and reinforcement. Approximating models could also be found and adapted, with reinforcement and feedback, to approach a specified behavioral ideal. When prior knowledge exists to establish appropriate constraints, this too can facilitate the learning process (e.g., imposing a bias on *what* can be learned). These techniques of computational learning theory employed in a formal learning domain extend to other suitably constrained domains, producing perfect results.

More interesting to us is today's the use of similar approaches to learning and adaptation in real-life applications. From link analysis, to adaptive user interfaces, to e-process design and assurance, to development of reactive agents and multiagent systems, and to construction of the Semantic Web, there are numerous realized and potential applications of interactive reinforcement-based learning and adaptive systems. Here we discuss our initial theoretical research involving such learning and adaptation. We describe applications we have found, and related results of other researchers.

2. Adaptation and Formal Learning

In our initial learning research we sought finite generative and recognitive models to represent what could be infinite bodies of linguistic knowledge. We showed that given a domain of language structures, a characterizing grammar or recognizer could be found. Either represented the entire language, as distinguished from its complement in that domain. Once a learner or system acquired either model the knowledge it represented would be learned. Constraining the domain enabled definition of characterizing models, and reduction of each to a finite minimal form. Experiments over the entire domain determined categories of knowledge constructs corresponding to the models' components. Establishing existence and components of characterizing models, through experiments on an infinite domain, was some very satisfying theory.

Better yet, effective experiments with a finite demonstrative sample of elements representing the categories provided sufficient information to determine the minimal models. An unsupervised learner could observe the elements of the knowledge domain, experiment and obtain the minimal results. Realistically, a supervised learner could experiment effectively to obtain the knowledge characterization, interacting with a competent informant in constructive or adversarial language games [3]. Informant reinforcement may distinguish correct examples from those *in*correct.

A knowledgeable informant can provide appropriate examples that will incrementally lead a system to learn: by constructing the prefect result. The informant, furthermore, can guide the system through sufficient test examples to determine adversarially if a potential model is incorrect. An iterative process may detect defects to be corrected until the perfect result is found. So, by reinforced incremental construction or adversarial adaptive iterations, the system could discover a minimal language model, and the knowledge it represents would be acquired.

Should the informant not know how much demonstration or experimentation is sufficient, our techniques insured that, monotonically, in-the-limit a perfect model would be found. In less constrained cases where there is no perfect model, our techniques adaptively produce approximating models representing knowledge subsets, with anomalous behaviors learnable later, as they arise. This is often the situation in real-life learning of dynamical systems and of growing, changing bodies of knowledge.

In our original work in computational learning theory, interaction between the informant and the system made learning more efficient and guaranteed it would occur. Adaptation could occur when the body of knowledge to-be-acquired changed, the domain under consideration changed, or the standard of acceptance changed (e.g., preferring a time-efficient to a space-efficient outcome). For any of these cases, with suitable interaction and reinforcement, the system could adapt and learn. These were our results, in the specific domain of computational learning theory.

3. Real Applications

In the research just described constructive or adaptive learning could occur effectively, within a constrained and formalized problem domain. A system could obtain a perfect or approximating knowledge model and thus acquire the knowledge, by finite means. The approach was successful because the problems considered were finitely representable (or approximately so) and because a theoretical approach enabled the imposition of domain constraints. Behaviors could be completely specified, and correct behavior distinguished from its complement. A system could learn or adapt a knowledge model effectively, because an informant could finitely characterize all that was correct, and all that was *not.* But learning could *also* occur if no specific informant were present to provide positive or negative reinforcement. As long as the learner's environment provided sufficient behavioral examples for observation, *and* the learner had an effective procedure to decide which of the examples were correct, learning would occur. These results generalized to acquisition of any finitely-realizable behavior in which membership was effectively decidable, relative to complementary behavior, within a containing knowledge domain. Real-life application domains typically lack the constraints that yielded our theoretical results, but we find that aspects of our work can be useful in various areas of practice.

For example, the field of *link analysis* involves behavioral modeling and adaptive processes applied in very practical e-environments today [2]. There masses of behavioral data are analyzed to determine patterns so that future actions may be predicted and, with feedback, existing behavioral models may be revised. We applied such techniques to determine approximating models in a financial activities domain that could never, realistically, be modeled perfectly. Our approximating model predicted activities correctly "often enough" to have genuine utility. But when a truly anomalous behavior was observed, giving negative reinforcement, the approximating model was adaptively revised. We have noted [2] that *e-process assurance* is another practical area that can productively apply a reinforcement learning and adaptive systems approach. Developers unable to depict perpetual potential system behavior (e.g., an on-line ordering system, expected to operate "forever") devise system approximations that they may subsequently model-check. Seeking to verify correctness using automated techniques, they actually detect existing system defects and flaws. With feedback the system, an approximate behavioral model, may be adaptively improved.

More obvious as a practical real-life instance of reinforcement learning and adaptiveness is the development of *reactive agents*, some of which are already deployed. E.g., a voice-recognition component of a brokerage-house system adapts and, with interaction, learns. We were able to train one to distinguish "Verizon" from "Verisign" and to re-train it when, after weeks of non-use, it "forgot". Now, after enough reinforcing voice-examples and positive feedback, the agent's recognition has stabilized.

Related instances are those we have found in today's *user interfaces* and improvements to *human-computer interaction*. In [4] we discuss some potential hardware and software system adaptations to interactively facilitate usage by small children and by adults who are not technically trained. For our present purposes, we note we have discovered financial systems that have learned to slow down and reduce choices in response to slow interactions with some users, but that can speed up and become more complex in response to users who are speedier. In the most ordinary activities we find deployed examples of learning and adaptive systems that are fine-tuned with interaction and/or some feedback. E.g., as a learning theorist we take notice of Word's ability to infer what it "thinks" we want and subsequently automate those inferred intentions. We also find it worthy of notice that we have the feedback option to correct the inferences, and the option to turn the automation off.

The *Semantic Web* [3,4] is an fine example of a system that must adapt and learn. It will benefit from human and agent interaction, feedback and reinforcement. Web-navigating agents are being developed to make logical decisions, choose actions, and obey instructions provided explicitly and *implicitly* by real-life humans. Agents may learn from examples, adapt with feedback, and exhibit adjustable autonomy as Personal Assistants to individual, diverse human users. Applying our theory, each user might teach and each system might adapt to the user's unique application of feedback and positive or negative reinforcement.

In the related research area of *adaptive multiagent systems* we have proposed [5] that our theory-based approach to learning be applied to formation, adaptation and maintenance of agent coalitions that can achieve complex specified behavioral goals. Appropriate selection and composition of agents forms a coalition whose elements solve components of complex problems, within local environments or on the Web. Selected agents may be configured dynamically to produce a specified goal-directed result, adapted when a preferable element or configuration is found (e.g., a more efficient component), and dynamically re-configured if a different behavioral goal is specified. In the traditional approach to agent coalitions as described throughout [13], agents convince themselves and each other that collective effort is beneficial. In our interpretation oversight and feedback guide the agent selection/collection. If reinforcement establishes successful behavior of a group of communicating, interacting agents, that agent structure may be maintained and re-employed in future coalitions.

We have shown a theoretically-designed adaptive Travel Assistant to be an instance of such a goal-directed agent coalition. It may choose the *best* route and transportation mode from a specified departure point to a specified destination, within such constraints as schedules and finances. It may reuse components, adapt or reconfigure as factors may change, and learn from reinforcing feedback just what choice is "best". E.g., components of the best route to LAX airport, discovered for our trip to Edinburgh to deliver [4], may be reused for a trip to Utah for JCIS.

4. Related Work and Conclusions

Initially, we used interaction, feedback and reinforcement to effect learning and adaptation in a theoretically constrained problem domain. There a knowledgeable informant (or a sufficiently observable environment) provided demonstrative examples leading a system to learn, by iteratively constructing a characterizing behavioral model; or to adapt, by interactively detecting and correcting behavioral defects and flaws. In less constrained real-life problems perfect results may not be obtained, but our iterative, adaptive modeling approach still has utility. Our techniques produce satisfactory results in our link-analysis based adaptation of a financial activities approximating model; system software error detection and correction evidenced in e-process assurance and design; reactive agent training of a voice-recognizer, with examples; user interfaces we activate; adjustable autonomy seen in development of agents navigating the Semantic Web; and formation, adaptation and maintenance of goal-directed communicating agent coalitions (such as our learning Travel Assistant) within adaptive multiagent systems. Thus we can apply our computational learning theory productively.

Many others have engaged our interest with their practical work utilizing feedback and reinforcement for learning and adaptation in real-life processes. There is a plethora of relevant AI for Electronic Commerce examples [6], such as collaborative filtering with elicitation and feedback to develop recommender systems; automated contract negotiations; interactive rule-authoring to let clients give some feedback for automation of purchasing systems; and information extraction using intelligent filtering to let users of interest share knowledge. In [10] we have found additional, Adaptive User Interface, examples including personalization of access to servers; programming by demonstration; multiagent architecture for Web-based distance learning (adapted to the needs of each user); and mastery learning enabling each user to proceed at his or her own pace.

In the theoretically-oriented work appearing in [7] we find examples of automated synthesis of systems related to the techniques of reinforcement learning, and applied to problems of real life. These include automated synthesis of computational finance systems by refinement of generic systems to meet analyst needs; deductive techniques for forming coalitions from autonomous Web-dwelling agents; scheduling with constraints; and the implementation of large-scale high performance network systems from expanding logical foundations and libraries of knowledge.

Within the multiagent systems area [9] combines case-oriented reinforcement learning and peer-related reinforcement learning for coalition formation, to facilitate adaptability to dynamic environments. Offline strategic planning is case-based and real-time, with peer-related reinforcement learning online and tactical. Based on simulated tasks, [9] concludes that with reinforcement learning agents form coalitions more effectively and efficiently.

Utility of reinforcement learning in conjunction with other learning approaches is noted in many of the recent supervisory control studies described in [11]. These include: applications to learning of robot skills and tasks from demonstration; human-robot interactions with flexible loci of learning; another look at programming by demonstration; and assistive technology. We are truly impressed with the cognitive orthotics work presented therein by Rudary, Singh and Pollack [12], using reinforcement learning to build adaptive reminder systems assisting the impaired. Combining reinforcement learning and temporal constraint reasoning, their systems are personalized and can adapt to changes in a user's cognitive abilities. We believe both science and humanity can benefit substantially from such research.

Language was the domain of our initial learning and adaptation research, and language is the foundation of the Semantic Web. Potential Semantic Web developments using reinforcement learning and adaptation include the Staab[14] inference of semantic links from observation of user behaviors, and the Steels[15] development of communicating agents, learning by language games. Cercone *et al* [1] propose computational techniques developed for Natural Language processing and learning be scaled up to the Web, including feedback and reinforcement techniques for machine translation. The need for some reinforcement is acknowledged by [8] where translation techniques include the option of user intervention to resolve ambiguities. We, ourselves, are pleased when a system asks us if we mean "Fuss" or "Fast", even though "Fass" is what we have specified and exactly what we mean. We believe there will *always* be a need for such interaction. Not *every* process can be automatically or adaptively learned by agents navigating the Semantic Web.

Only some of the promising applications of which we are aware have been described above. We believe that reinforcement learning, in its purest form of environmental observation and combined with other learning and decision-making techniques, is the key to significant real-life learning.

5. References

[1] N. Cercone, L. Hou, V. Keselj, A. An, K. Naruedomkul and X. Hu, "From Computational Intelligence to Web Intelligence", *IEEE Computer* (Vol 35, No 11), Nov 2002: pp.72-76.

[2] L.F. Fass, "Adaptive Modeling Techniques for Intelligent Business", in *Proc. of the Joint Conference on Information Sciences*, Cary NC, Sept. 2003: pp. 1739-1742.

[3] L.F. Fass, "Language, Mathematics and Other Dangerous Things", *ACM SIGACT News*, Vol. 35, No.1 (March 2004): pp. 74-79.

[4] L.F. Fass, "The 'Digital Divide' Just Isn't What It Used to Be", in *Proc. of the Int. Conf. on Software Engineering IFIP Workshop on Bridging the Gaps Between SE and HCI*, Edinburgh Scotland, May 2004, The IEE W1L: pp.83-87.

[5] L.F. Fass, "An Automata-Theoretic View of Agent Coalitions", in [13]: pp. 18-21.

[6] T. Finin and B. Grosof, (Eds), *Artificial Intelligence for Electronic Commerce*, Orlando FL, July 1999, AAAI Press WS99-01.

[7] B. Fischer and D.R. Smith, (Eds), *Logic-Based Program Synthesis: State of the Art and Future Trends*, Stanford Univ, March 2002, AAAI Press SS02-05.

[8] H. Li, Y. Cao and C. Li, "Using Bilingual Web Data to Mine and Rank Translations", *IEEE Intelligent Systems* (Vol 18, No 4), July/August 2003: pp.54-59.

[9] X.Li and L-K Soh, "Investigating Reinforcement Learning in Multiagent Coalition Formation", in [13]: pp.22-28.

[10] S. Rogers and W.Iba (Eds),. *Adaptive User Interfaces*, Stanford Univ., March 2000, AAAI Press SS00-01.

[11] M.Rosenstein and M. Ghavamzadeh (Eds), *Supervisory Control of Learning and Adaptive Systems*, San Jose CA, July 2004, AAAI Press WS04-10.

[12] M.R .Rudary, S. Singh and M.E. Pollack, "Reinforcement Learning for Adaptive Cognitive Orthotics", in [11]: pp. 63-64.

[13] L-K Soh and J.E. Anderson (Eds), *Forming and Maintaining Coalitions & Teams in Adaptive Multiagent Systems*, San Jose CA, July 2004, AAAI Press WS04-06.

[14] S. Staab (Ed), "Emergent Semantics", Trends & Controversies Section, *IEEE Intelligent Systems* (Vol 17, No.1), Jan/Feb 2002: pp. 78-86.

[15] L. Steels, "Language Games for Emergent Semantics" in [14]: pp. 83-85.