Algorithms for Planning and Decision making Under Uncertainty

Research Proposal, Chandrashekar L

I. Introduction

Real world is often uncertain, starting from the time taken to service a customer in a queue, to the return-on-investment in a particular stock. Decision making in the face of such uncertainty is hence crucial while designing practical systems with good performance. As Fig. 1 illustrates, *decision making under uncertainty* is at the heart of several important research areas that have had major practical impact in the last few decades. In practice, the underlying uncertainty is captured using an appropriate stochastic model and an optimal decision is made based on the model. However, in complex real world systems, due to the *curse-of-dimensionality* deriving such decision rules often become computationally intractable. While dimensionality free methods [3] have had partial success in alleviating this problem, it is still an active area of research. More importantly, with the advent of the big data era (**http://dst.gov.in/big-data-initiative-1**) , it is crucial to develop decision making algorithms which are data driven (Fig. 2) as opposed to just being model based.

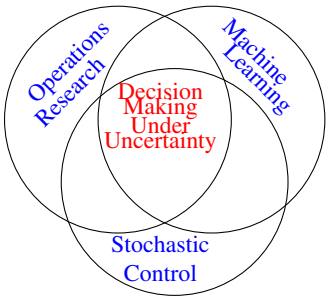

Fig. 1: Decision making under uncertainty is at the heart of many important areas such as Machine Learning, Operations Research and Stochastic Control.

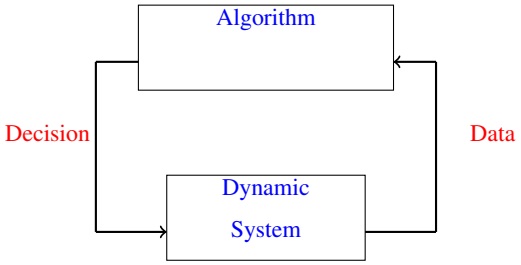

Fig. 2: The paradigm of data-driven approach to decision making is characterized by 'system-in-loop' or 'feedback' control.

II. Background

Fig. 3 shows the various aspects to be taken into account while adopting a data-driven approach towards decision making. In what follows, we discuss these aspects briefly before highlighting the challenges and the research gaps that this proposal wishes to address.

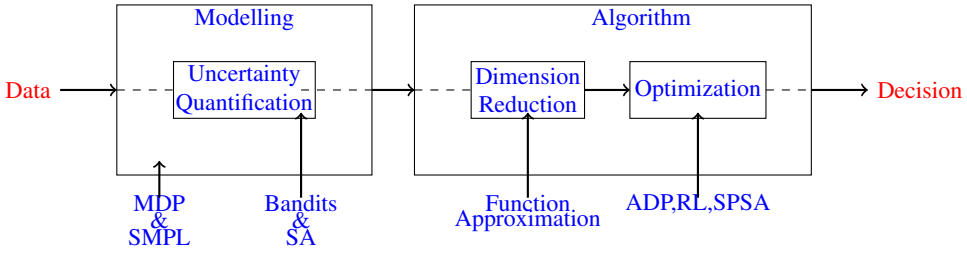

Fig. 3: The figure shows the various steps starting with *data*, followed by *modelling* and then by *algorithm* leading to the *decision rule* . The modelling step involves *uncertainty quantification* and the algorithmic step can further be split into *dimensionality reduction* and *optimization*. Markov Decision Processes (MDPs) and Stochastic Max-Plus (SMPL) system are useful mathematical models of uncertainty. Multi-arm bandit and Stochastic Approximation (SA) theories helps us in uncertainty quantification. Function approximation techniques are used to address the curse-of-dimensionality. *Approximate Dynamic Programming* (ADP), *Reinforcement Learning* (RL) and *Simultaneous Perturbation Stochastic Approximation* (SPSA) algorithms are optimization routines that eventually lead to the final decision rule.

A. Modeling

Markov Decision Processes (MDP): Many real world decision making problems such as control of sensor networks, control of queuing systems, traffic control, inventory control, terrain exploration by robots, sequential drug administration, allocation of tasks and payments in service systems etc are examples of sequential decision making problems that can be modeled as MDP. Further, the Markovian nature of the system dynamics enables MDPs to be solved using the principle of Dynamic Programming (DP) leading us to the optimal decision rule.

Stochastic Max-Plus (SMPL) Systems: are the class of discrete event systems where elementary components interact via 'synchronization' [13]. The inherent randomness in the different interacting components causes the stochastic behavior. Examples of such systems include railway networks, production chain, scheduling, queuing and digital systems.

B. Uncertainty Quantification

Multi-Armed Bandit: Theory [2] characterizes the classical *exploration-exploitation* trade-off. Bandit theory addresses the question of optimal sampling and at the same time with lesser compromise in performance, and is helpful in scenarios where samples need to be collected in an efficient way.

Stochastic Approximation (SA): Theory [7] helps us to build data-drive algorithms wherein sample data can be plugged in the place of the unknown ground truth.

C. Algorithm

Dimensionality Reduction: *Curse-of-dimensionality* denotes the fact that the number of states in the system grows exponentially in the number of state variables. Function approximation is a widely used dimensionality reduction wherein the each state is represented by a feature vector and computations are carried by taking linear combinations of the features. Dimensionality reduction is achieved by letting the dimension of the feature vector to be much less than the number of states.

Approximate Dynamic Programming: (ADP) algorithms [3] tackle the complexity of systems (i.e, of large number of states) by cleverly combining approximate representations and mathematical optimization.

Reinforcement Learning: In most cases, the underlying system model in not known explicitly. However, the samples can be obtained via simulation or direct interaction with the system. Reinforcement Learning (RL) algorithms [16] learn the optimal decision via feedback obtained by directly interacting with the system. RL algorithms such as Q-learning [18], temporal difference learning [17] have been successful in domains such as Backgammon, elevator control etc.

III. Objective: Open Problems and Potential Directions

We now list below several research gaps that we propose to address (Table I).

1) **ADP with Performance Guarantees:** Since the ADP methods are not exact, in most cases they only compute approximate or sub-optimal decision rules. Several well known ADP algorithms suffer from convergence issues [4] and the performance loss of the sub-optimal policy cannot be ascertained. However, some of our recent works [11, 14] have shown conditions under which ADP algorithms are convergent and yield provably *good* decision rule. The focus here is to explore newer ADP algorithms and approximation architectures that can be guaranteed to converge and yield a good decision rule.
2) **ADP for Constrained and Risk Sensitive MDPs:** The discounted cost and the average cost are the most widely studied formulations in the classical MDP setting. However, in practice the decisions need to balance multiple costs and sensitivities to the various costs can be different. Constrained MDP formulation allows us to take into account the various costs involved. However, ADP algorithms for constrained MDPs have not been sufficiently explored. The Approximate Linear Programming (ALP) [12] approach to MDP can accommodate multiple cost functions. A possible research direction is to extend [14] to constrained MDP to derive new ADP algorithms. *Risk* sensitive formulation of the

Problem	ADP	RL	Applications
Constrained MDP	*	[6], +	Urban Planning
Risk Sensitive MDP	*	[8], +	Health Care
SMPL	[13]	*	Traffic
Bandit	[1]	[2], +	Crowdsourcing

TABLE I: Summary of open problems and potential contributions. Here '$*$' denotes potential for fundamental contribution, and '$+$' denotes scope for significant additions.

MDP considers exponential cost and hence can be used in applications where risk avoidance is key. Exploring ADP algorithms for the risk sensitive setting is another line of research.

3) **ADP for Stochastic Max-Plus Systems:** Most of the current approaches are based on model predictive control [13]. Exploring machine learning and stochastic optimization [5] inspired ADP algorithms for SMPL systems is of interest.
4) **Reinforcement Learning:** It is of interest to explore sample efficient, stable and convergent RL algorithms for systems with large number of states especially for the constrained, risk sensitive formulations.
5) **Multi-Arm Bandits:** Several variants of the problem including linear/convex/contextual bandits, bandits with complex or partial feedback have been considered in literature. While upper bounds for regret is known in various cases finding matching lower bounds is a challenging research direction that can be explored.

IV. Significance: Applications and Impact

Addressing the open problems in Section III will help us in developing algorithms for the following domains.

Urban Planning: has separate costs related to the development of public transportation (roads/rail), facility development (housing, hospitals, industries) and resource distribution (power, water and sewage). Further, only a fixed budget is available at any given time. We propose *Constrained MDPs* that will help us to take care of these multiple costs. Also, ADP algorithms for SMPL systems can be applied to synchronize commuter traffic across the road and rail networks.

Health Care: An interesting recent advancement is to use data-driven decision making in the domain of health care. In particular, exploiting the patient specific data (history, congenital defects etc) to decide the remedy/drug-dosage to be administered from time to time is a challenging problem [15]. Another potential challenge is to use mobile devices for patient monitoring and for preventive interventions (alerting patients on the intake of medicine, therapy and rest). The algorithms for *Risk Sensitive* MDPs can be applied to factor the *risk* associated with wrong treatments.

Sensor Networks: Sensors serves as an important backbone of real-time data-driven decision making. In general, most sensors are mobile and battery operated, it is important to spend as less battery power as possible. As a result energy efficient transmission has attracted the attention in recent times [9]. This can be achieved by forming a network of sensors with one centralized controller that collates information from other nodes, and the center also has control over the *sleep-wake* schedules of all the other nodes. Multi-agent RL algorithms can be used to derive decentralized decision rules.

Human Computing: Crowdsourcing (crowd) is a new mode of organizing work in multiple groups of smaller chunks of tasks and outsourcing them to a distributed and large group of people in the form of an open call. We propose Bandit theory can be used to develop algorithms that use real time data (number of workers in the crowd, the number of pending tasks, the quality of the workers etc) to derive efficient decisions for pricing the tasks [10], allocating them to the crowd workers to their ensure timely completion.

In all the application domains, we propose to develop new algorithms that are robust, scalable and real time, and in cases where previous work exists, extend significantly the state of the art by gaining novel

insights. These newer domains are expected to lead us to develop newer theoretical frameworks and also look at some of the well known theoretical challenges under new light.

V. Long-Term Prospects

The models, analysis and algorithms developed in this research will also naturally extend to ecology (modeling prey/predator characteristics, migratory behavior, afforestation planning) and disaster management (modeling and planning of resources under calamity). Algorithms for road traffic, sensor network and city planning will eventually lead to building of smarter cities. A long-term goal is to evolve public policies more in the lines of projects such as PATH and TOPL (http://www.path.berkeley.edu/).

References

[1] A. Agarwal, D. P. Foster, D. J. Hsu, S. M. Kakade, and A. Rakhlin. Stochastic convex optimization with bandit feedback. In *Advances in Neural Information Processing Systems*, pages 1035–1043, 2011.

[2] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Foundations of Computer Science, 1995. Proceedings., 36th Annual Symposium on*, pages 322–331. IEEE, 1995.

[3] D. P. Bertsekas. Approximate dynamic programming. In *Dynamic Programming and Optimal Control*, volume II. Athena Scientific, Belmont, MA, 3rd edition, 2011.

[4] D. P. Bertsekas. *Dynamic Programming and Optimal Control*, volume II. Athena Scientific, Belmont, MA, 4th edition, 2013.

[5] S. Bhatnagar, M. C. Fu, S. I. Marcus, and I. Wang. Two-timescale simultaneous perturbation stochastic approximation using deterministic perturbation sequences. *ACM Trans. Model. Comput. Simul.*, 13:180–209, April 2003.

[6] V. S. Borkar. An actor-critic algorithm for constrained markov decision processes. *Systems & control letters*, 54(3):207–213, 2005.

[7] V. S. Borkar. *Stochastic Approximation: A Dynamical Systems Viewpoint*. TRIM, 2008.

[8] V. S. Borkar and S. P. Meyn. Risk-sensitive optimal control for markov decision processes with monotone cost. *Math. Oper. Res.*, 27(1):192–209, 2002.

[9] Q. Cao, T. Abdelzaher, T. He, and J. Stankovic. Towards optimal sleep scheduling in sensor networks for rare-event detection. In *Proceedings of the 4th international symposium on Information processing in sensor networks*, page 4. IEEE Press, 2005.

[10] L. Chandrashekar, A.Dubey, S. Bhatnagar, and B. Chithralekha. A markov decision process framework for predictable job completion times on crowdsourcing platforms. In *Proceedings of the Seconf AAAI Conference on Human Computation and Crowdsourcing, HCOMP 2014, November 2-4, 2014, Pittsburgh, Pennsylvania, USA*, 2014.

[11] L. Chandrashekar and S. Bhatnagar. Approximate dynamic programming with (min; +) linear function approximation for markov decision processes. In *53rd IEEE Conference on Decision and Control, CDC 2014, Los Angeles, CA, USA, December 15-17, 2014*, pages 1588–1593, 2014.

[12] D. P. de Farias and B. Van Roy. The linear programming approach to approximate dynamic programming. *Operations Research*, 51(6):850–865, 2003.

[13] S. S. Farahani, T. van den Boom, and B. De Schutter. Exact and approximate approaches to the identification of stochastic max-plus-linear systems. *Discrete Event Dynamic Systems*, 24(4):447–471, 2014.

[14] C. Lakshminarayanan and S. Bhatnagar. A generalized reduced linear program for markov decision processes. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA.*, pages 2722–2728, 2015.

[15] P. Liao, P. Klasnja, A. Tewari, and S. A. Murphy. Sample size calculations for micro-randomized trials in mhealth. *Statistics in medicine*, 2015.

[16] R. S. Sutton and A. G. Barto. *Introduction to Reinforcement Learning*. MIT Press, Cambridge, MA, USA, 1st edition, 1998.

[17] J. N. Tsitsiklis and B. Van Roy. An analysis of temporal-difference learning with function approximation. 1997.

[18] C. J. Watkins and P. Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.