Multiagent Decision Making and Learning in Urban Environments

Akshat Kumar
School of Information Systems, Singapore Management University
akshatkumar@smu.edu.sg

Abstract

Our increasingly interconnected urban environments provide several opportunities to deploy intelligent agents—from self-driving cars, ships to aerial drones—that promise to radically improve productivity and safety. Achieving coordination among agents in such urban settings presents several algorithmic challenges—ability to scale to thousands of agents, addressing uncertainty, and partial observability in the environment. In addition, accurate domain models need to be learned from data that is often noisy and available only at an aggregate level. In this paper, I will overview some of our recent contributions towards developing planning and reinforcement learning strategies to address several such challenges present in large-scale urban multiagent systems.

1 Introduction

Our society and urban environments are rapidly getting interconnected by the internet of things (IoT). A number of smart devices embedded in everyday objects are capable of sensing their environment, and taking decisions to increase our productivity, safety and efficiency. As an example, autonomous self-driving cars are able to perceive their environment, and interact with each other to create future applications such as smart traffic light intersections [Au et al., 2016]. Similarly, for maritime traffic, e-navigation aims to improve the management of the sea traffic by digitizing both on-board marine information and the communication between vessels and maritime traffic control authorities1. Such e-navigation would pave the way for autonomous vessels, and has tremendous potential to improve coordination among vessels to reduce congestion and improve safety of navigation in busy ports of the world [Agussurja et al., 2018]. There is no lack of such interconnected urban environments where learning and modeling interactions among agents (which may represent self-driving cars and trucks, autonomous vessels, drones) is the key to enable the overall productivity and safety of the resulting large multiagent system.

Our recent work is directed towards modeling such large urban systems, and developing scalable planning and reinforcement learning (RL) based approaches that enable effective coordination among agents. Decentralized partially observable MDP (Dec-POMDPs) have emerged as a popular framework for modeling such multiagent sequential decision making problems under uncertainty [Bernstein et al., 2002; Kumar and Zilberstein, 2009; Amato et al., 2010; Kumar et al., 2015; Kumar et al., 2016]. However, it is known to be challenging (NEXP-Hard complexity) even for the smallest two-agent systems [Bernstein et al., 2002]. To address the complexity, various models are explored where agent interactions are limited by design by enforcing various conditional and contextual independencies such as transition and observation independence among agents [Nair et al., 2005] where agents are coupled primarily via joint-rewards, event driven interactions [Becker et al., 2004], and weakly coupled agents [Spaan and Melo, 2008].

Our contributions. Previous such models and algorithms in multiagent decision making have been either relatively general but not very scalable, or relatively scalable but with limited applicability. Our recent and ongoing work challenges this current state of affairs by proposing new models and algorithms that (a) are applicable to a wide range of problems of practical importance, particularly in urban system optimization, (b) lead to scalable algorithms for coordinating thousands of agents in settings where agents partially observe their environment, and there is uncertainty present, and (c) constructing faithful domain simulators for different urban settings (e.g., taxi fleet optimization, maritime traffic management) learned from real world historical data. A key part of our innovation involves using graphical models and probabilistic inference to learn models of urban systems, and the development of new reinforcement learning strategies, that allow agents to rapidly discover more efficient decisions through feedback on past behavioral outcomes using a domain simulator.

2 The Dec-POMDP Model

A Dec-POMDP generalizes single agent Markov decision process to account for multiple agents operating in the environment. A distinguishing feature is that agents observe their environment and other agents only partially. Based on

the local information agents receive (which may be different for different agents), each agent chooses the next action to take (in parallel) operating in a sequential manner over a finite or an infinite horizon. At each time step, the agent-team also obtains a joint-reward. The goal is to compute policies (mapping from local observation history to actions for each agent) to maximize the total reward over the planning horizon. The joint-reward makes the problem cooperative, and action selection based on local observations makes the problem decentralized.

A Dec-POMDP can be defined by a tuple $\{I, S, \{A^i\}, P, R, \{Y^i\}, O, \gamma\}$, where $I$ denotes a finite set of $n$ agents; $S$ denotes a finite set of states with designated initial state distribution $\pi_0$; $A^i$ denotes a finite set of actions for each agent $i$; $P$ denotes state transition probabilities: $P(s'|s, a_i)$, the probability of transitioning from state $s$ to $s'$ when the joint-action $a = (a_1, \ldots, a_n)$ is taken by the agents; $R$ denotes the reward function: $R(s, a)$ is the immediate reward for being in state $s$ and joint-action taken as $a$; $Y^i$ denotes a finite set of observations for each agent $i$; $O$ denotes the observation probabilities: $O(y|s', a)$ is the probability of receiving the joint-observation $y$ when the last joint-action taken was $a$ that resulted in the environment state being $s'$; $\gamma$ denotes the reward discounting factor. An agent $i$'s policy, $\theta^i : Y^i \rightarrow A^i$, maps the set of all possible observation histories $Y^i$ to actions. Solving a Dec-POMDP entails finding the joint-policy $\theta = (\theta^1, \ldots, \theta^n)$ that maximizes the total expected reward:

$$
E \left[ \sum_{t=0}^{\infty} \gamma^t R(s_t, a_t; \theta) \right]
$$

where $\theta$ denotes the joint-policy and subscript $t$ denotes the dependence on time. There are several representations possible for local policies $\theta^i$ such as policy trees, finite-state controllers [Amato et al., 2010; Kumar et al., 2015], and deep neural networks [Nguyen et al., 2018].

3 Modeling Urban Environments

Several urban environments can be modeled as a diffusion, cascade or flow of entities (e.g., vehicles, vessels, humans) over an underlying geographical network [Kumar et al., 2013]. For example, traffic flow can be modeled as diffusion of vehicles over the network [Kumar et al., 2013], maritime traffic as vessel movement between sea zones [Singh et al., 2019], and people flow over a geographical area [Iwata and Shimizu, 2019]. However, in many such domains, individual data tracking the movement of each entity is either not available (e.g., to protect privacy of individuals) or too expensive to collect. Only the aggregate or collective data (which may be noisy or missing) is observed. For example, consider a road traffic network. A key learning problem in such traffic networks is estimating the turn probabilities for each road segment of this network [Kumar et al., 2013]. Several popular analytical models of traffic flow such as the cell transmission model [Daganzo, 1994] are based on the assumption that turn probabilities are known a priori for each location. In several urban traffic networks, aggregate data in the form of vehicle count is already collected for each road segment using inductive-loop traffic detectors, and we show that such aggregate level information is sufficient to learn turn probabilities for traffic networks and model the traffic flow [Kumar et al., 2013].

As another example, figure 1 shows the e-navigation chart (ENC) of a strait [Singh et al., 2019]. The ENC is composed of several features such as anchorages where vessels anchor and wait for services, berths, pilot boarding grounds, and the traffic separation scheme or TSS. The TSS (figure 1) is the set of mandatory unidirectional routes designed to carry bulk of the maritime traffic to reduce collision risk among vessels transitioning through or entering the Straits. Based on geographical features, the TSS can be further divided into smaller zones, and maritime traffic can be thought of as flow of vessels over zones. In the maritime case, although individual vessel trajectories are available, modeling the precise movement of each vessel is intractable (it requires modeling interaction with other vessels, effects of weather on the movement among other factors). Therefore, modeling the traffic at the aggregate level of zones (where we observe how many vessels are present in which zone at each time step) is significantly more tractable, and in our empirical tests, we show such an aggregate modeling is accurate enough to replicate historical patterns [Singh et al., 2019].

3.1 Collective Graphical Models and Learning Domain Simulators

We use the framework of collective graphical models (CGMs) to model several types of urban environments [Sheldon and Dietterich, 2011] where we fit a model of the behavior of individuals but our data consist only of aggregate information or counts. CGMs compactly describe the distribution of the aggregate statistics of a population sampled independently from a discrete graphical model. Let $G = (V, E)$ denote an undirected graph. Consider the following pairwise graphical model over the discrete random vector $X = (X_1, \ldots, X_{|V|})$:

$$
p(x; \theta) = \Pr(X = x; \theta) = \frac{1}{Z(\theta)} \prod_{(i,j) \in E} \phi_{ij}(x_i, x_j; \theta). \tag{2}
$$

Here, $\phi_{ij}(:; \theta)$ is a local potential defined on the setting of variables $(X_i, X_j)$. The local potentials depend on the parameter vector $\theta$, and $Z(\theta)$ is the partition function. We assume that each variable $X_i$ takes values in the same finite set $\mathcal{X}$. Now, consider an ordered sample $x^{(1)}, \ldots, x^{(M)}$ of random vectors drawn independently from the graphical model.
We also refer to this sample as a population (of size $M$). We define the contingency tables $\mathbf{n}_i = (n_i(x_i) : x_i \in X)$ over nodes of the model and $\mathbf{n}_{ij} = (n_{ij}(x_i, x_j) : x_i, x_j \in X')$ over edges of the model, whose entries count the number of times particular variable settings occur in the population. Define the vector $\mathbf{n}$ to be the concatenation of all edge-based contingency tables $\mathbf{n}_{ij}$ together with all node-based contingency tables $\mathbf{n}_i$. This is a random vector that depends on the entire population and comprises sufficient statistics of the population, which can be seen by writing the joint probability:

$$p(x^{(1)}, \ldots, x^{(M)}; \theta) = g(n, \theta) = \frac{1}{Z(\theta)^M} \prod_{(i,j) \in E} \phi_{ij}(x_i, x_j; \theta)^{n_{ij}(x_i, x_j)}.$$  \hspace{1cm} (3)

In CGMs, one makes noisy observations $y$ of some subset of the sufficient statistics $n$ and then seeks to answer queries about the sufficient statistics given $y$ (e.g., for the purpose of learning the parameters $\theta$) through the conditional distribution $p(n | y; \theta) \propto p(n; \theta)p(y | n)$. The first term in this product, $p(n; \theta)$, is the prior distribution over the sufficient statistics or the CGM distribution. Its exact form is shown in [Sheldon et al., 2013]. The second term, $p(y | n)$, is the noise model. Figure 2 shows a graphical representation of a CGM over a chain structured individual model. We have worked on developing several probabilistic inference based approaches to develop message-passing algorithms to compute the maximum-a-posteriori assignment (or the most likely $n$ given the noisy observations $y$) and learning the parameters $\theta$ using maximum likelihood estimation [Sheldon et al., 2013; Kumar et al., 2013; Sun et al., 2015; Nguyen et al., 2016]. CGMs are an ideal formalism to represent urban domains such as traffic. We have used models based on CGMs to represent the maritime traffic and learned parameters of such models using historical aggregate data [Singh et al., 2019]. We envision that constructing domain simulators based on such aggregate modeling of data would be crucial for scaling up learning and decision making in large scale urban environments as CGMs provide a tractable representation to model a large number of agents.

### 3.2 Collective Multiagent Decision Making

Our recent work focuses towards developing general decision theoretic frameworks for collective multiagent decision making that allow to control the behavior of a population of nearly identical agents operating collaboratively in an uncertain and partially observable environment. Our key enabling insight and related assumption is that in several urban environments (such as transportation, supply-demand matching) agent interactions are governed by the aggregate count and types of agents, and do not depend on the specific identities of individual agents. This insight makes it possible to construct scalable and general approaches to multiagent modeling, simulation and optimization that are capable of addressing a range of practical problems in urban systems. Such modeling also addresses shortcomings of previous multiagent planning approaches which are either general but not scalable or scalable but with very limited applicability.

To formalize such collective decision making problems, we have recently developed the framework of CDec-POMDP [Nguyen et al., 2017a; Nguyen et al., 2017b; Nguyen et al., 2018] or collective decentralized POMDPs. The CDec-POMDP model is based on the idea of partial exchangeability [Diaconis and Freedman, 1980; Niepert and Van den Broeck, 2014], and collective graphical models [Sheldon and Dieterich, 2011; Sun et al., 2015]. Partial exchangeability in probabilistic inference is complementary to the notion of conditional and contextual independence, and combining all of them leads to a larger class of tractable models and inference algorithms [Niepert and Van den Broeck, 2014]. Previous works in multiagent planning have mostly explored only conditional and contextual independences in multiagent models [Nair et al., 2005; Witwicki and Durfee, 2010]. CDec-POMDPs combine both conditional independences and partial exchangeability to solve much larger instances of multiagent decision making.

Figure 3(a) shows how different agents $m$ in a population of $M$ agents interact with each other. We assume that different agents share the state space $S$. E.g., in a transportation network, agents move in different zones of the same city. An agent $m$’s local state at time $t$ is denoted by $s^m_t$. The local states of all the agents are aggregated to form the state count table $n^t_i$ which simply counts how many agents are present in each state $i \in S$. Based on its local state and the state count table, an agent $i$ receives its local observation $o^i_t$, which it
uses to take the next action $a_t^n$ (using the policy $\pi$). Based on
the joint states and actions of all the agents, the state-action
count table $n_{i,j}$ is generated, which simply counts how many
agents in state $i$ took action $j$ (for each $i \in S$ and $j \in A$). As a
result of joint actions, the environment transitions to the next
state, and a reward is given per agent. This particular model
is equivalent to the individual model similar to figure 2(a).

**Scalability.** In urban settings, we may have thousands of
agents. Sampling individual trajectories of each agent would
be computationally intractable. Therefore, we exploit similar
properties as in CGMs, and marginalize away individual
agents to arrive at the collective planning model in figure 3(b)
(analogous to figure 2(b)). This model only consists of count
tables, and similar to CGMs, we show how to define a distri-
bution $p(n)$ over these count tables [Nguyen et al., 2017a].
Notice that sampling from $p(n)$ is highly scalable as the di-
ensions of count tables do not depend on the population size
$M$. Therefore, the $\text{CDec-POMDP}$ model is able to effectively
reason about a large population of agents.

**4 Solution Approaches**

**Planning-as-inference.** We have developed different types
of solution approaches for computing policies for agents in
the $\text{CDec-POMDP}$ model. One direction is based on the
planning-as-inference strategy [Toussaint and Storkey, 2006]
where we cast the planning problem to that of likelihood max-
imization (LM) problem in a graphical model [Nguyen et al., 2017a].
We have explore extensively such planning-as-inference strategy
for multiagent decision making in several contexts [Kumar and Zilberstein, 2010; Kumar et al., 2011; Kumar et al., 2015; Ghosh et al., 2015; Singh and Kumar, 2019].

The main benefit of this strategy is that it opens the
door to the application of machine learning approaches to
planning. We have used a popular LM approach Expectation-
Maximization (EM) for multiagent decision making. A key
benefit of EM is that its updates often take the form of
message-passing among agents, and are thus highly scalable
for large multiagent systems.

**Multiagent RL.** Another direction we have explored for
solving $\text{CDec-POMDP}$s is using multiagent RL (MARL).
The MARL approaches are useful in settings when only the
access to domain simulator is available, which is a fairly
common setting for several urban environments.

There exist several previous MARL approaches such as independent
Q-learning, counterfactual multiagent policy gradients and
actor-critic methods [Foerster et al., 2018; Lowe et al., 2017],
and SARSA-based MARL for Dec-POMDPs [Dibangoye
and Buffet, 2018]. However, most of these approaches are
limited to few dozens of agents in contrast to the collective
setting with thousands of agents, which is our goal.

**Lifted multiagent RL.** The key idea that our MARL
approaches exploit is to lift the RL algorithms to work with
count-based representations. We show how to define action-
value function and value function over count tables, and prove
that they are sufficient statistic for planning and RL for $\text{CDec-
POMDP}$s [Nguyen et al., 2017a; Nguyen et al., 2018].

As shown in figure 4(b), the environment directly generates
different count tables and the associated reward by exploiting
the CGM-like distribution defined over the graphical model
in figure 3(b). As a result, our RL methods do not have
to sample individual agent trajectories that would have been
prohibitively expensive. In addition, there are two main chal-
enges we address for collective MARL—multiagent credit
assignment (actions of which agents were more/less impor-
tant), and computing low variance policy gradient estimates
for faster convergence to high quality solutions even with
thousands of agents. Without addressing these issues, stan-
dard policy gradient based approaches do not converge at all.

**5 Conclusion**

We have developed several approaches for achieving coordi-
nation in large multiagent systems that are increasingly be-
coming common in our urban environments. Our work in-
cludes representing urban domains using collective graphical
models that exploit the property that agent interactions in sev-
eral urban settings depend on their aggregate effects rather
than their identities. We have developed several domain simu-
lators for urban transportation settings (such as maritime
traffic). We have used these simulators to develop efficient
and scalable multiagent RL approaches that exploit such ag-
gregate nature of interaction among agents.

We have also addressed several challenges that arise when doing planning
and RL with thousands of agents such as multiagent credit assign-
ment and low variance gradient estimates.

**Acknowledgments**

I thank my collaborators and mentors. I also thank the UNi-
CEN center at SMU (https://unicen.smu.edu.sg/) for provid-
ing a conducive environment. The author is supported by the
Singapore Ministry of Education Academic Research Fund
(AcRF) Tier 2 grant MOE2018-T2-1-179.

**References**

[Agussurja et al., 2018] Lucas Agussurja, Akshat Kumar, and
Hoong Chuin Lau. Resource-constrained scheduling for mar-
time traffic management. In AAAI Conference on Artificial In-

[Amato et al., 2010] Christopher Amato, Daniel S. Bernstein, and
Shlomo Zilberstein. Optimizing fixed-size stochastic controllers
for pomdps and decentralized pomdps. Autonomous Agents and

[Au et al., 2016] Tsz-Chiu Au, Shun Zhang, and Peter Stone. Au-
tonomous intersection management for semi-autonomous vehi-
cles. In Dusan Teodorović, editor, Handbook of Transportation,


