Linearly Solvable Markov Games

Krishnamurthy Dvijotham and Emo Todorov

***Abstract*—Recent work has led to an interesting new theory of linearly solvable control, where the Bellman equation characterizing the optimal value function is reduced to a linear equation. Already, this work has shown promising results in planning and control of nonlinear systems in high dimensional state spaces. In this paper, we extend the class of linearly solvable problems to include certain kinds of 2-player Markov Games. In terms of modeling power, the new framework is more general than previous work, and can apply to any noisy dynamical system. Also, we obtain *analytical* solutions to continuous-state control problems with linear dynamics and a very flexible class of cost functions: Mixtures of** $\text{Gaussians} \times \text{Polynomials}$**. The linearity leads to many other useful properties: the ability to compose solutions to simple control problems to obtain solutions to new problems, a convex optimization formulation of inverse optimal control etc. We demonstrate the usefulness of the framework through examples of forward and inverse optimal control problems in continuous as well as discrete state spaces.**

I. Introduction

Optimal Control is a conceptually appealing framework for building solutions to complex control problems: One specifies a high level cost-function encoding the desired goals of the task, and the optimization process fills in all the details. However, the huge computational costs of solving optimal control problems has severely limited the application of optimal control to practical problems with nonlinear dynamics, high dimensional state/control spaces. Motivated by this, researchers have tried to find restricted classes of control problems that are easier to solve, yet general enough to model interesting control problems. One class of interesting problems that are easier are *Linearly Solvable MDPs (***LMDPs***)*[1] and related path-integral control problems [2], for which the Bellman Equation (BE) characterizing the optimal value function can be made linear. This has several other interesting consequences: the ability to build solutions to new control problems by combining the solutions to simpler control problems[3], an efficient unconstrained convex formulation of inverse optimal control[4] etc. Already, this work has had encouraging success in domains like character control for animation[5] and robotic control[6].

In recent work [7], the framework of **LMDPs** was extended to the risk sensitive setting where the controller optimizes a risk-sensitive objective. This allows one to tune the controller to trade-off risk and return: For example, a risk averse controller will settle for a less energy-efficient control strategy if it means that the probability of something going wrong (due to the noise in the system) is reduced. However, one limitation of this framework is that the controls and noise are required to act in the same subspace, that is, one cannot have actuation in state dimensions where there is no noise, and conversely, one cannot have noise in the state dimensions that are not directly actuated. This is a serious limitation for many real systems that we seek to overcome in this paper. We develop a family of control problems (Linearly Solvable Markov Games (**LMGs**)) formulated in a game theoretic setting, for which we can obtain a linear BE without imposing this restriction. The first player is the controller and the second player is adversarial noise. Another advantage of this new formulation is that we get a Bellman equation linear in the *value function* space, while **LMDPs** need an exponential transformation to make the value function linear. This leads to nicer numerical behavior and other interesting properties. We call this new class of problems *Linearly Solvable Markov Games* (**LMGs**).

We believe that the primary use of this framework is to design robust control policies in a computatefficiently. The game theoretic setting we consider falls within the standard setting of Robust or H_∞ control [8] that has been studied extensively and shown to produce controllers robust to model errors. We show experimentally that optimizing the new objective does produces sensible behaviors (section VII), and that by adjusting the cost function slightly, one can even get *exactly* the same behavior as a traditional MDP formulation.

II. Background and Notation

A. Notation

We use $\mathcal{X}$ to denote the state space, $\mathbf{x}$ to denote states, $\mathbf{v}(\mathbf{x})$ for the optimal value function. Let $\mathcal{U}$ denote the space of feasible control signals, $\mathcal{P}[\mathcal{U}]$ be the set of probability distributions over $\mathcal{U}$ and $\mathcal{U}^{\mathbf{R}^+}$ be the set of positive functions on $\mathcal{U}$. For any $p \in \mathcal{P}[\mathcal{U}]$, let $\text{supp}[p] = \{\mathbf{u} \in \mathcal{U} : p(\mathbf{u}) > 0\}$. Define the KL-divergence between two members of $\mathcal{P}[\mathcal{U}]$ by $\text{KL}\left(\Pi \parallel \Pi^0\right) = \sum_{u \in \mathcal{U}} \Pi(u) \log\left(\frac{\Pi(u)}{\Pi^0(u)}\right)$, which is well-defined when $\text{supp}[\Pi] \subseteq \text{supp}\left[\Pi^0\right]$. Let f be any real-valued function on $\mathcal{U}$. We denote the expectation of f under π as $\text{E}_\pi[f] = \sum_u \pi(u) f(u)$ and let $\Psi_\pi[f] = \log\left(\text{E}_\pi[\exp(f)]\right)$. We denote an un-normalized Gaussian with mean μ and covariance Σ as $f(\mathbf{x}) = \tilde{\mathcal{N}}(\mathbf{x}; \mu, \Sigma)$ and a normalized Gaussian distribution as $\mathcal{N}(\mathbf{x}; \mu, \Sigma)$. We will work with discrete time problem and in the finite horizon formulation for most derivations, but all results extend to other formulations as well.

B. Markov Games

A zero-sum 2-player Markov game (aka Dynamic Game) [9] is described by a state space $\mathcal{X}$, control spaces for the 2 players: $\mathcal{U}$,$\mathcal{U}_\text{a}$, a stochastic Markovian Dynamics $\mathbb{P}(\mathbf{x}'|\mathbf{x}, \mathbf{u}, \mathbf{u}_\text{a})$

K. Dvijotham is with the Department of Computer Science and Engineering, University of Washington, Seattle, USA - 98195 dvij@cs.washington.edu

E. Todorov is with the Departments of Computer Science and Engineering & Applied Mathemetics, University of Washington, Seattle, USA - 98195 todorov@cs.washington.edu

and a cost function $\ell(\mathbf{x}, \mathbf{u}, \mathbf{u}_\mathrm{a})$. The objective of player 1 is to minimize expected cost under the stochastic dynamics, while that of player 2 is to maximize it. As opposed to MDPs, the best strategy for a player critically depends on the strategy of the other player and hence there is no universally optimal policy. However, game theory resolves this dilemma by prescribing that a player choose actions so as to minimize the worst-case cost, ie, the highest cost over all possible moves of the second player. The optimal solution comes out of the following Bellman-Issacs equation[9]:

$$\begin{aligned}\mathbf{v}_t(\mathbf{x}) &= \min_{\mathbf{u}} \max_{\mathbf{u}_\mathrm{a}} \ell_\mathrm{t}(\mathbf{x}, \mathbf{u}, \mathbf{u}_\mathrm{a}) + \mathop{\mathrm{E}}_{\mathbf{x}' \sim \mathbb{P}(\cdot|\mathbf{x},\mathbf{u},\mathbf{u}_\mathrm{a})} [\mathbf{v}_t(\mathbf{x}')] \\ \mathbf{v}_T(\mathbf{x}) &= \ell_\mathrm{f}(\mathbf{x}) \end{aligned} \quad (1)$$

The value function defined here is actually the *upper value function* [9]. By switching the max and min, one can obtain the *lower value function*. These two are not equal in general, and in fact, the games we will consider in this paper have different upper and lower value functions, but we will stick to the upper value function. In the context of control, the upper value function has the following nice interpretation: Since the adversary acts second, he can choose actions that maximize the future expected cost given the current action chosen by the controller, hence behaving like worst-case noise and leading to more robust control policies.

III. Linearly Solvable Markov Games

In this section, we describe the restricted family of Markov Games for which the Bellman Isaacs reduces to a linear equation (**LMGs**). We will consider noisy discrete time dynamical systems defined by an equation of the form

$$\mathbf{x}' \sim \mathbb{P}(\cdot|\mathbf{x}, \mathbf{u}_\mathrm{net})$$

where $\mathbf{u}_\mathrm{net}$ is the control input to the system. In the game theoretic setting we consider, this net control input $\mathbf{u}_\mathrm{net}$ is probabilistic and its distribution is determined by the net effect of the controller and adversary. In the absence of any interference by the controller or the adversary, we assume that there is a **passive policy** over control inputs $\Pi^0(\mathbf{x}) \in \mathcal{P}[\mathcal{U}]$. For most control problems, one would pick $\Pi^0(\mathbf{x})$ so that it places high probability on small (cheap) controls and low probability on large (expensive) controls. If one has a control cost $\ell(\mathbf{x}, u)$ in mind, a sensible way to choose $\Pi^0(\mathbf{x})$ is $\Pi^0(\mathbf{x})[u] \propto \exp(-\ell(\mathbf{x}, u))$.

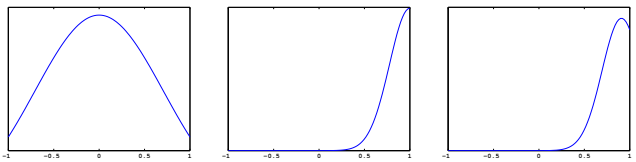
The controller and adversary work by modifying the **passive policy** distribution to a new distribution. Mathematically, this is formulated as a probability shift operator. Given two positive functions p, q on some set S, define

$$(p \otimes q)(s) = (q \otimes p)(s) = \frac{p(s)q(s)}{\sum_{s \in S} p(s)q(s)}$$

This is visualized in figure 1.

An **LMG** game proceeds as follows

- The system is in state $\mathbf{x}$ at time t.
- The controller picks $\mathbf{u} \in \mathcal{P}[\mathcal{U}]$
- The adversary picks $\mathbf{u}_\mathrm{a} \in \mathcal{P}[\mathcal{U}]$



(a) p (b) q (c) $p \otimes q$

Fig. 1. Probability Shift

- The system transitions to a new state $\mathbf{x}' \sim \mathbb{P}(\cdot|\mathbf{x}, \mathbf{u}_\mathrm{net}), \mathbf{u}_\mathrm{net} \sim \Pi^0(\mathbf{x}) \otimes \mathbf{u} \otimes \mathbf{u}_\mathrm{a}$.

At each step, the controller incurs a cost equal to the sum of a state and control cost:

$$\begin{aligned}\ell_\mathrm{t}(\mathbf{x}, \mathbf{u}, \mathbf{u}_\mathrm{a}) = \underbrace{\ell_\mathrm{t}(\mathbf{x})}_{\text{State Cost}} &+ \underbrace{\mathrm{KL}\left(\Pi^0(\mathbf{x}) \parallel \Pi^0(\mathbf{x}) \otimes \mathbf{u}\right)}_{\text{Control Cost for Controller}} \\ &- \underbrace{\mathrm{KL}\left(\Pi^0(\mathbf{x}) \otimes \mathbf{u} \otimes \mathbf{u}_\mathrm{a} \parallel \Pi^0(\mathbf{x}) \otimes \mathbf{u}\right)}_{\text{Control Cost for Adversary}}\end{aligned}$$

The adversary incurs the negative of this cost. The objective of both players is to minimize the cost incurred to them over time. Thus, the controller and adversary shift the passive policy $\Pi^0(\mathbf{x})$ by picking $\mathbf{u}, \mathbf{u}_\mathrm{a}$ to $\Pi^0(\mathbf{x}) \otimes \mathbf{u} \otimes \mathbf{u}_\mathrm{a}$. The net effect of both shifts is the resulting distribution of control inputs. The cost function is composed of 3 parts: an arbitrary state cost $\ell_\mathrm{t}(\mathbf{x})$ and a control cost $\mathrm{KL}\left(\Pi^0(\mathbf{x}) \parallel \Pi^0(\mathbf{x}) \otimes \mathbf{u}\right)$ for the controller, which measures how much $\mathbf{u}$ shifts the passive policy, and $\mathrm{KL}\left(\Pi^0(\mathbf{x}) \otimes \mathbf{u} \otimes \mathbf{u}_\mathrm{a} \parallel \Pi^0(\mathbf{x}) \otimes \mathbf{u}\right)$, the control cost for the adversary, which measures how much further the adversary shifts the passive policy beyond the controller.

Definition 1. A Linearly Solvable Markov Game (**LMG**) is a 2-player zero-sum game parameterized by a state space $\mathcal{X}$, a control space $\mathcal{U}$, stochastic dynamics $\mathbb{P}(\mathbf{x}, \mathbf{u}_\mathrm{net}|\in)\mathcal{P}[\mathcal{X}]$ for each $\mathbf{x}, \mathbf{x}' \in \mathcal{X}, \mathbf{u}_\mathrm{net} \in \mathcal{U}$, a passive policy $\Pi^0(\mathbf{x}) \in \mathcal{P}[\mathcal{U}] \forall \mathbf{x} \in \mathcal{X}$. The dynamics of the system are given by:
$\mathbf{x}' \sim \mathbb{P}(\mathbf{x}, \mathbf{u}_\mathrm{net}), \mathbf{u}_\mathrm{net} \sim \Pi^0(\mathbf{x}) \otimes \mathbf{u} \otimes \mathbf{u}_\mathrm{a}$
$\mathbf{u}, \mathbf{u}_\mathrm{a} \in \mathcal{U}^{\mathbf{R}^+}, \mathbf{x}, \mathbf{x}' \in \mathcal{X}, \mathbf{u}_\mathrm{net} \in \mathcal{U}$

and the cost function by:

$\ell_\mathrm{t}(\mathbf{x}, \mathbf{u}, \mathbf{u}_\mathrm{a}) = \ell_\mathrm{t}(\mathbf{x}) + \mathrm{KL}\left(\Pi^0(\mathbf{x}) \parallel \Pi^0(\mathbf{x}) \otimes \mathbf{u}\right)$
$- \mathrm{KL}\left(\Pi^0(\mathbf{x}) \otimes \mathbf{u} \otimes \mathbf{u}_\mathrm{a} \parallel \Pi^0(\mathbf{x}) \otimes \mathbf{u}\right)$

The first player is called the *controller* and the second player the *adversary*. For the objective to be well defined, we require that $\mathbf{u} > 0$. The game can be formulated both in finite and infinite horizon settings. The time-dependence of the cost drops in infinite horizon cases.

Theorem 1. *The* BE *for an* **LMG** *is linear. The* BE *for*

different problem formulations is:

$$\begin{aligned}
\textit{Finite-Horizon}: \mathbf{v}_t(\mathbf{x}) &= \ell_{\mathrm{t}}(\mathbf{x}) + \mathop{\mathrm{E}}_{\mathbf{u}_{\mathrm{net}} \sim \Pi^0(\mathbf{x}), \mathbb{P}(\mathbf{x}, \mathbf{u}_{\mathrm{net}})} [\mathbf{v}_{t+1}] \\
\mathbf{v}_T(\mathbf{x}) &= \ell_{\mathrm{f}}(\mathbf{x}) \\
\textit{First-Exit}: \mathbf{v}(\mathbf{x}) &= \ell(\mathbf{x}) + \mathop{\mathrm{E}}_{\Pi^0(\mathbf{x}), \mathbb{P}} [\mathbf{v}] \; \forall \mathbf{x} \in \mathcal{N} \\
\mathbf{v}(\mathbf{x}) &= \ell_{\mathrm{f}}(\mathbf{x}) \, \forall \mathbf{x} \in \mathcal{T} \\
\textit{Average Cost}: \mathbf{v}(\mathbf{x}) + c &= \ell_{\mathrm{t}}(\mathbf{x}) + \mathop{\mathrm{E}}_{\Pi^0(\mathbf{x}), P} [\mathbf{v}] \\
\textit{Discounted Cost}: \mathbf{v}(\mathbf{x}) &= \ell(\mathbf{x}) + \gamma \mathop{\mathrm{E}}_{\Pi^0(\mathbf{x}), P} [v] \qquad (2)
\end{aligned}$$

where $\mathcal{T}, \mathcal{N}$ are the terminal and non-terminal states,respectively, for first-exit problems, c is the average cost parameter for infinite-horizon average cost problems, and γ is the discounted factor for infinite horizon discounted problems.

Proof: We prove the result for the finite-horizon case. The proof for other formulations is similar. Plugging the cost and dynamics from Definition 1 into the Bellman-Issacs equation (1), we get:

$$\begin{aligned}
\mathbf{v}_t(\mathbf{x}) &= \ell_{\mathrm{t}}(\mathbf{x}) + \min_{\mathbf{u}} \max_{\mathbf{u}_{\mathrm{a}}} \mathrm{KL}\left(\Pi^0(\mathbf{x}) \parallel \Pi^0(\mathbf{x}) \otimes \mathbf{u}\right) \\
&\quad - \mathrm{KL}\left(\Pi^0(\mathbf{x}) \otimes \mathbf{u} \otimes \mathbf{u}_{\mathrm{a}} \parallel \Pi^0(\mathbf{x}) \otimes \mathbf{u}\right) + \mathop{\mathrm{E}}_{\Pi^0(\mathbf{x}) \otimes \mathbf{u} \otimes \mathbf{u}_{\mathrm{a}}} \mathop{\mathrm{E}}_{P} \mathbf{v}_{t+1}
\end{aligned}$$

Since the first KL divergence doesn't depend on $\mathbf{u}_{\mathrm{a}}$, we can bring the $\max_{\mathbf{u}_{\mathrm{a}}}$ inside. Letting $p = \Pi^0(\mathbf{x}) \otimes \mathbf{u}$, we get

$$\begin{aligned}
&\max_{\mathbf{u}_{\mathrm{a}}} - \mathrm{KL}\left(q \otimes \mathbf{u}_{\mathrm{a}} \parallel q\right) + \mathop{\mathrm{E}}_{\mathbf{u}_{\mathrm{net}} \sim q \otimes \mathbf{u}_{\mathrm{a}}} \left[\mathop{\mathrm{E}}_{\mathbb{P}(\mathbf{x}, \mathbf{u}_{\mathrm{net}})} [\mathbf{v}_{t+1}] \right] \\
&= - \min_{\mathbf{u}_{\mathrm{a}}} \mathrm{KL}\left(q \otimes \mathbf{u}_{\mathrm{a}} \parallel q\right) - \mathop{\mathrm{E}}_{\mathbf{u}_{\mathrm{net}} \sim q \otimes \mathbf{u}_{\mathrm{a}}} \left[\mathop{\mathrm{E}}_{\mathbb{P}(\mathbf{x}, \mathbf{u}_{\mathrm{net}})} [\mathbf{v}_{t+1}] \right] \\
&= \Psi_q [\mathbf{v}_{t+1}] = \Psi_{\Pi^0(\mathbf{x}) \otimes \mathbf{u}} [\mathbf{v}_{t+1}]
\end{aligned}$$

where the last line follows from lemma (1). Thus the problem reduces to

$$\begin{aligned}
&\min_{\mathbf{u}} \mathrm{KL}\left(\Pi^0(\mathbf{x}) \parallel \Pi^0(\mathbf{x}) \otimes \mathbf{u}\right) + \Psi_{\Pi^0(\mathbf{x}) \otimes \mathbf{u}} [\mathbf{v}_{t+1}] \\
&= \mathop{\mathrm{E}}_{\mathbf{u}_{\mathrm{net}} \sim \Pi^0(\mathbf{x})} \left[\mathop{\mathrm{E}}_{\mathbb{P}(\mathbf{x}, \mathbf{u}_{\mathrm{net}})} [\mathbf{v}_{t+1}] \right]
\end{aligned}$$

where the last lins follows from lemma (2). Thus, we get a *Linear Bellman Equation*:

$$\begin{aligned}
\mathbf{v}_t(\mathbf{x}) &= \ell_{\mathrm{t}}(\mathbf{x}) + \mathop{\mathrm{E}}_{\mathbf{u}_{\mathrm{net}} \sim \Pi^0(\mathbf{x}), \mathbb{P}(\mathbf{x}, \mathbf{u}_{\mathrm{net}})} [\mathbf{v}_{t+1}] \\
\mathbf{v}_t(\mathbf{x}) &= \ell_{\mathrm{f}}(\mathbf{x}) \\
\Pi^*(\mathbf{x}; t) &= \exp(-\mathbf{v}_{t+1}) \\
\Pi^*{}_{\mathrm{a}}(\mathbf{x}; t) &= \exp(\mathbf{v}_{t+1})
\end{aligned}$$

■

A. Interpretation of the Result

We can see that $\Pi^0(\mathbf{x}) \otimes \Pi^*(\mathbf{x}; t) \otimes \Pi^*{}_{\mathrm{a}}(\mathbf{x}; t) = \Pi^0(\mathbf{x})$, since $\Pi^*(\mathbf{x}; t) \Pi^*{}_{\mathrm{a}}(\mathbf{x}; t) = 1$. Thus, the controller and adversary effectively cancel each other, so that the optimally controlled dynamics is just the dynamics under the passive policy. This explains why the optimal value function is just the value function corresponding to the passive dynamics with only the state cost. However, the policy $\Pi^*(\mathbf{x}; t)$ obtained for the controller is still sensible and can be applied even in the absence of an adversary: in fact, this policy has been optimized to deal with a very powerful adversary, and should be robust to a wide variety of perturbations.

B. Implications

We briefly summarize the major implications of this result, many of which are discussed in detail in the upcoming sections:

Solving Optimal Control Problems Efficiently: One can leverage sparse linear solvers and methods like TD,LSTD [10] for policy evaluation.

Modeling Power: This framework can be applied to any noisy dynamical system $\mathbf{x}' \sim \mathbb{P}(\mathbf{x}, \mathbf{u}_{\mathrm{net}})$.

Analytical Solutions: We obtain analytical solutions for systems with linear dynamics $\mathbf{x}' = A\,\mathbf{x} + B\,\mathbf{u}$ and costs that are mixtures of Gaussians $\times$ Quadratics (section IV-C).

Compositionality: Given the solutions to a set of control problems with "simple" cost functions, we can construct the optimal solution for any linear combination of the costs analytically (section V).

Inverse Optimal Control: We have a tractable convex-optimization based solution to the inverse optimal control problem (section VI).

C. Relationship to Previous Work

The results here are most closely related to results in [7]. In that paper, the **LMDP** framework was extended to the case of risk-sensitive control, replacing the standard KL divergence in **LMDPs** with a Rényi divergence $\mathbb{D}_\alpha$ []. It was also shown that the results in that paper can be re-interpreted in a game theoretic setting very similar to the one here, except that the control cost for the controller was replaced by $\mathbb{D}_\alpha\left(\Pi^0(\mathbf{x}) \parallel \Pi^0(\mathbf{x}) \otimes \mathbf{u}\right)$ and the dynamics were required to be deterministic $\mathbf{x}' = \mathcal{F}(\mathbf{x}, \mathbf{u}_{\mathrm{net}})$. It can be shown that as $\alpha \to 1$, $\mathbb{D}_\alpha(p \parallel q) \to \mathrm{KL}(p \parallel q)$ and thus we reduce to the results of [7] with $\alpha = 1$. However, this is only true when the **dynamics are deterministic**. For general stochastic dynamics, $\mathbf{x}' \sim \mathbb{P}(\mathbf{x}, \mathbf{u}_{\mathrm{net}})$, the results presented here are more general and have no analog in previous work on linearly solvable control. Thus, in general, the **LMG** framework has greater modeling power, as it can handle arbitrary stochastic dynamics, as opposed to **LMDP**, which requires the controls and noise to act in the same subspace. This modeling power could be very useful in practical applications. For example, we can handle control-dependent noise, which has proved very useful in modeling human movements [11].

IV. Problems with Analytical Solutions

In this section, we study special cases that admit closed-form solutions to the linear Bellman equation. We first discuss classical results on Linear-Quadratic Games and then show how the new **LMG** results compare with them.

A. Deterministic Linear Quadratic Games

Consider a 2-player zero sum Markov Game with linear dynamics affine in the controls of both players:

$\mathbf{x}_{t+1} = A_t\,\mathbf{x}_t + B_t\,\mathbf{u} + D_t\,\mathbf{u}_{\mathrm{a}}$

with quadratic costs

$\frac{1}{2}\left[\mathbf{x}^T\,Q_t\,\mathbf{x} + \mathbf{u}^T\,\mathbf{u} - \gamma^2\,\mathbf{u}_{\mathrm{a}}^T\,\mathbf{u}_{\mathrm{a}}\right]$

It can be shown [8] that this game admits a saddle point solution under certain assumptions on D, B and the optimal value function is quadratic $\mathbf{v}_t(\mathbf{x}) = \frac{1}{2}\,\mathbf{x}^T\,V_t\,\mathbf{x}$ with the V_t satisfying the generalized Ricatti equation:

$$V_t = Q_t + A_t^T\left(V_{t+1}^{-1} + B_t^T B_t - \tfrac{D_t^T D_t}{\gamma^2}\right)^{-1} A_t$$

$$\Pi^*(\mathbf{x}; t) = -B_t^T\left(V_{t+1}^{-1} + B_t^T B_t - \tfrac{D_t^T D_t}{\gamma^2}\right)^{-1} A_t x$$

B. Linear Quadratic Gaussian (LQG) Problems

Theorem 2. *Consider* **LMGs** *with Linear dynamics, Gaussian noise and quadratic state/costs:*

$\mathbf{x}_{t+1} \sim \mathcal{N}(A_t\,\mathbf{x}_t + B_t\,\mathbf{u}_{\mathrm{net}t}, C_t C_t^T)$

$\ell_{\mathrm{t}}(\mathbf{x}) = \frac{1}{2}\,\mathbf{x}^T\,Q_t\,\mathbf{x}, \Pi^0(\mathbf{u}_{\mathrm{net}}\,|\,\mathbf{x}) = \mathcal{N}(0, I)$

It can be shown that the optimal value function is Quadratic, and the Generalized Ricatti equations are given by:

$$\mathbf{v}_t(\mathbf{x}) = \frac{1}{2}\,\mathbf{x}^T\,V_t\,\mathbf{x} + c_t, V_t = Q_t + A_t^T V_{t+1} A_t$$

$$c_t = c_{t+1} + \mathrm{tr}\left((B_t B_t^T + C_t C_t^T) V_{t+1}\right) \tag{3}$$

1) Relationship to Deterministic LQ games: If $D_t = \gamma B_t$, the Bellman equation for Deterministic LQ Games (section IV-A) reduces to $V_t = Q_t + A_t^T V_{t+1} A_t$, which basically corresponds to the value of the null policy $u_t = w_t = 0$. What actually happens is that the optimal controls for both players cancel each other out in this case, leaving just the passive uncontrolled system. This is the same as the generalized Ricatti equation obtained above (except for constants).

C. Non-LQG Problems with Analytic Solutions

It turns out that if we have a (possibly time-varying) linear dynamical system, any finite-horizon problem with costs of the form

$\ell_{\mathrm{t}}(\mathbf{x}) = \sum_i \tilde{\mathcal{N}}(\mathbf{x}; \mu_{i,t}, \Sigma_{i,t})\mathrm{Poly}_{i,t}(x)$

where Poly denotes a polynomial, can be solved analytically in **LMG** framework. This is a very powerful result, as almost any cost can be approximated with a cost in the above form. Also, this might allow us to even solve problems with non-linear dynamics approximately, by having a linear dynamical system but penalizing for the violations in the cost function. In this section, we derive the solution for a special case (to keep the math simple). We assume that there is no running cost $\ell_{\mathrm{t}}(\mathbf{x}) = 0\,\forall t < T$, we have a noiseless system ($C_t = 0$,$\Pi^0(u|x) = \mathcal{N}(0, I)$) and the polynomial in the final cost is a single Gaussian times quadratic. Define $\tilde{\mathcal{N}}_Q(x; \mu, \Sigma, m, S, s)$ to be $\tilde{\mathcal{N}}(x; \mu, \Sigma)\left(\frac{(x-m)^T S (x-m)}{2} + s\right)$. The Bellman equation becomes $\mathbf{v}_t(\mathbf{x}) = E_{\mathbf{u}_{\mathrm{net}}}\mathbf{v}_{t+1}(A_t\,\mathbf{x} + B_t\,\mathbf{u}_{\mathrm{net}})$. If $\mathbf{v}_{t+1}$ is a Gaussian $\times$ quadratic, its easy to see that $\mathbf{v}_t(\mathbf{x})$ is also a Gaussian $\times$ quadratic $\mathbf{v}_t(\mathbf{x}) = \tilde{\mathcal{N}}_Q(\mathbf{x}; \mu_t, \Sigma_t, m_t, S_t, s_t)$. With a little algebra, it is easy to show that the parameters satisfy the Ricatti equations:

$$\begin{aligned}
&\text{Let } M_t = \left(\Sigma_{t+1}^{-1} + B_t B_t^T\right), W_t = I + B_t^T \Sigma_t B_t \\
&S_t = \frac{A_t^T S_{t+1} A_t}{\det(W_t)}, s_t = \frac{s_{t+1} + \mathrm{tr}\left(W_t^{-1} B_t^T S_{t+1} B_t\right)}{\det(W_t)} \\
&m_t = A_t^{-1}((I - \Sigma_{t+1}^{-1} M_t^{-1})(m_{t+1} - \mu_{t+1}) + m_{t+1}) \\
&\mu_t = A_t^{-1}\mu_{t+1}, \Sigma_t = A_t^T M_t^{-1} A_t
\end{aligned} \tag{4}$$

Unfortunately, the optimal control law doesn't have a closed form expression in this case. However, once we compute the value function, numerically approximating the optimal control law shouldn't be a big problem: one can even try finding the mode of the optimal control policy $\Pi^*(\mathbf{x}; t)$ using numerical optimization:

$$\Pi^*(\mathbf{x}; t) = \operatorname*{amin}_u u^T u + \mathbf{v}_{t+1}(A_t\,\mathbf{x} + B_t u)$$

V. Compositionality of Optimal Control Laws

In this section, we discuss compositionality: the idea that solutions to complex control problems can be constructed by combining solutions to simpler control problems in certain ways. Linearly Solvable Problems often offer nice compositionality properties [3], which have been used to construct solutions to complex control problems like walking [5]. We have even nicer compositionality properties for **LMGs**:

Theorem 3. *Suppose that $\mathbf{v}^1, \mathbf{v}^2, \ldots, \mathbf{v}^k$ are the optimal value functions corresponding to* **LMGs** *with cost functions ${\ell_{\mathrm{t}}}^1, {\ell_{\mathrm{t}}}^2, \ldots, {\ell_{\mathrm{t}}}^m$ and dynamics $\mathbb{P}(\mathbf{x}'|\mathbf{x}, \mathbf{u}_{\mathrm{net}})$, then the optimal value function for $\ell_{\mathrm{t}} = \sum_i w_i\,{\ell_{\mathrm{t}}}^i$ is $\sum_i w_i \mathbf{v}^i$. This result is valid for all formulations: For the finite horizon and first-exit formulations, both the running costs and final costs must be combined with the same weights.*

The above result follows directly from the Linear Bellman equation (2). These results are more powerful than the ones presented in [3], since they apply to all problem formulations and allow composing running costs as well.

VI. Inverse Optimal Control

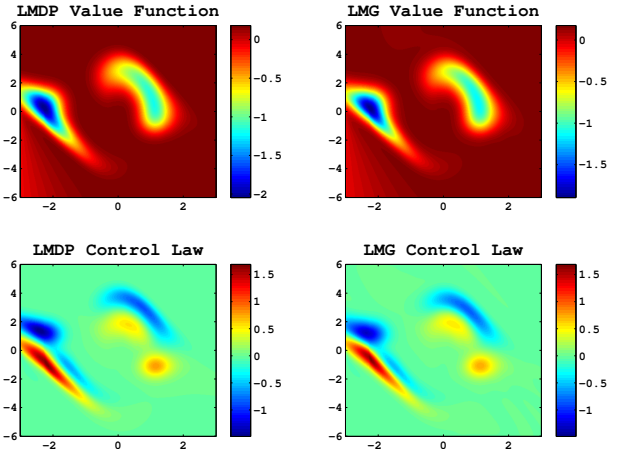
Given trajectories sampled from an optimal controller, can one infer the cost function with respect to which that controller is optimal? This is the **inverse optimal control** problem. The inverse optimal control problem also has an efficient solution in the **LMG** framework. The problem can be stated as follows: Given trajectories generated by an optimal controller and the system dynamics $\mathbb{P}(\mathbf{x}'|\mathbf{x}, \mathbf{u}_{\mathrm{net}})$, estimate the cost/value function of the controller. In the **LMG** framework, we solve this problem assuming that we can do inverse dynamics, ie, given that the system transitioned from $\mathbf{x}$ to $\mathbf{x}$', get a reliable estimate of $\mathbf{u}_{\mathrm{net}}$. We then propose a two-step process: First do inverse dynamics to figure out $\mathbf{u}_{\mathrm{net}\,t}$ given $\mathbf{x}_t, \mathbf{x}_{t+1}$ for every pair of consecutive states along a trajectory to get a dataset $\mathcal{D} = \{\mathbf{x}_i, \mathbf{u}_{\mathrm{net}\,i}\}$. Then, use maximum-likelihood estimation on the $(\mathbf{x}, \mathbf{u}_{\mathrm{net}})$ pairs to infer the value/cost function:

$$\max_{\theta} \sum_{i} \mathop{\mathrm{E}}_{\mathbb{P}(\mathbf{x}_i,\mathbf{u}_{\mathrm{net}\,i})} \left[\mathbf{v}^{\theta}\right] - \Psi_{\Pi^0(\mathbf{x}_i)} \left[- \mathop{\mathrm{E}}_{\mathbb{P}(\mathbf{x}_i,\mathbf{u}_{\mathrm{net}\,i})} \left[\mathbf{v}^{\theta}\right]\right]$$

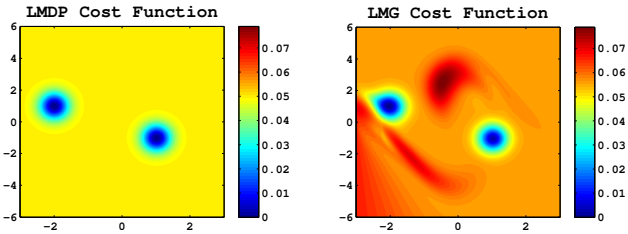
This is a convex optimization problem in θ if $\mathbf{v}^{\theta} = f(\mathbf{x})^T\theta$, ie , if $\mathbf{v}$ is parameterized linearly in θ. Also, like the work in [4], this method is computationally more efficient than other methods for inverse optimal control [12] [13] since it does not require

VII. Experiments

A. Car on a Hill Problem



(a) Value Functions and Control Laws for the Car-on-the-Hill Problem. X-axes represent position, Y-axes velocity and the colors represent the value of the differential cost-to-go or the scalar control signal

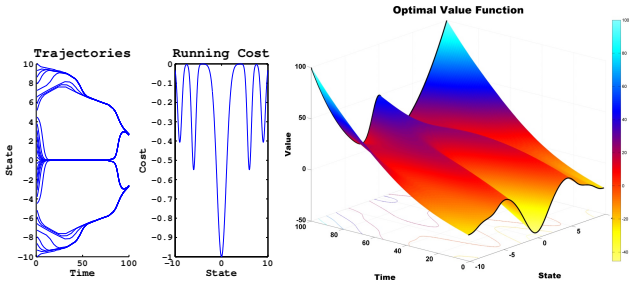


(b) Cost Functions in Both Frameworks that Produce the Same Control Law

Fig. 2.

We consider one of the benchmark problems of Reinforcement Learning first: The Car-on-a-Hill problem. We do this mainly as a sanity check: To see that the new formulation gives us sensible control policies. The state space is 2 dimensional: position p and velocity v of the car moving on a hill shaped like a Gaussian Curve $y = \exp(-0.5x^2)$. The cost function asks the car to oscillate between the desirable states $(-2, 1), (1, -2)$. We consider the infinite-horizon average-cost formulation of the problem. We solve this problem by discretizing the state space with a 400×400 grid, in both the traditional **LMDP**[1] and the new game-theoretic **LMG**(with unrestricted actions) frameworks. The solver takes about 1.5 seconds to converge in both cases, on an Intel i7 2.93Ghz CPU. The resulting value functions and optimal control laws are plotted in figure 2(a). One can see from the figures that the solution to both problems look very similar, demonstrating that the **LMG**framework can model traditional optimal control problems well. One can also ask the question: How should I change my cost function for the **LMG** so that the resulting control law matches the **LMDP**? Since the mapping between the control law and the value function is identical in both frameworks for the noiseless case, it is sufficient that the value functions match. Given the optimal value function v from the **LMDP** solution, we construct the cost function $\ell(\mathbf{x}) = \mathbf{v}(\mathbf{x}) - \mathrm{E}_{\mathbb{P}(\mathbf{x},\mathbf{u}_{\mathrm{net}}),\mathbf{u}_{\mathrm{net}}\sim\Pi^0(\mathbf{x})}[\mathbf{v}]$ that makes $\mathbf{v}$ optimal for the **LMG**(figure 2(b)).

B. Continuous State Problems with Analytical Solutions



(a) Trajectories/Cost Function (b) Time Varying Value Function

Fig. 3. **LMGs** with Analytical Solution

1) Randomly Generated Cost Function: We now consider a non-LQG problem that has an analytical solution IV-C. We consider a problem with a one dimensional continuous state $x \in \mathbf{R}$, so that the solution can be visualized, and linear dynamics $\dot{x} = -0.1x$. The running cost is a randomly generated mixture of inverted Gaussians and the final cost is an Gaussian centered at 0. The point of this experiment is mainly to demonstrate that we can handle fairly complicated costs with this, and the value function generated has fairly nontrivial structure. The time-varying value function for this problem is plotted in figure 3(b). Trajectories sampled from the optimal controller and the cost function are plotted in figure 3(a).

2) Obstacle Avoidance: We can model obstacles using Gaussians centered at obstacles. Targets can be modeled similarly using inverted Gaussians. If we have linear dynamics, we can take advantage of the analytical solution available for

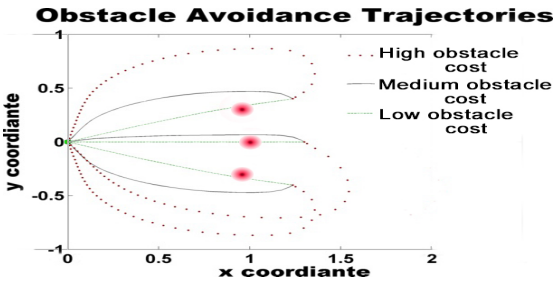


Fig. 4. Obstacle Avoidance ((Noisy) Obstacles in red, Target in green)

this case to solve this problem efficiently. Here, as a simple example, we consider a 2d point mass trying to reach a target at the origin, with 3 obstacles in between it needs to avoid. The final cost is a negatively scaled Gaussian centered at the target and the running cost is a scaled version of the final cost plus a sum of Gaussians centered at the obstacles. The trajectories from the resulting controller are plotted in figure 4. We plot 3 sets of trajectories: solid black (appropriate obstacle cost), dashed green (low obstacle cost) and dotted red (high obstacle cost), starting from various initial positions. This algorithm could be potentially very useful in high dimensional path-planning applications.

C. Inverse Optimal Control

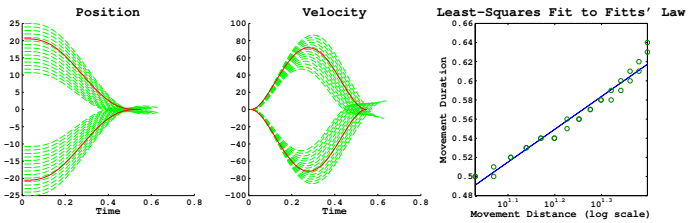


Fig. 5. Fitts' Law: Ideal Trajectories- Red Solid Lines, Learned Controller-Green Dashed Lines

We now present results evaluating the inverse optimal control algorithm VI parameterizing the value function as a mixture of Gaussians $\times$ Quadratics. The means Gaussians are set by running k-means on the data, and the variances adjusted to interpolate smoothly between the means. We consider the problem of modeling pointing movements: Fitts' law [14] characterizes the time T required for the motion as a function of the distance d to be moved and the size of the target w: $T = a + b\log_2(2d/w)$. Given the movement duration, it has been shown [15] that the trajectory of the movement is described well by a minimum-jerk model, ie, humans pick the trajectory that minimizes the magnitude of change in acceleration summed over time. We generate data from this model (with fixed target size $w = 1$,$a = 0, b = 0.1$), and run the inverse algorithm to infer a controller. We then sample trajectories from this controller and compare them to the ideal model: the results are shown in figure 5. We also fit the coefficients a, b based on the data generated by the learned controller. The resulting fit (figure 5) is also quite good, the recovered coefficients $\hat{a} = 0.03, \hat{b} = 0.103$ are very close to those used to generate the data.

VIII. Conclusions

We have presented a new class of Markov Games for which the Bellman-Isaacs equation can be made linear. This expands the family of linearly solvable control problems beyond the general class presented in [7]. The problems presented in this paper can deal with arbitrary stochastic dynamics, which gives it more modeling power than the previous framework. This extra modeling power comes with the restriction that we're forced to work with a KL divergence control cost, as opposed to the general Rényi divergence used in [7]. We have showed through numerical examples though, that this restriction does not seem particularly severe and we can model a variety of interesting problems as **LMGs**. The results here show that this is a promising direction of research and future work on developing numerical approximation techniques for scaling **LMGs** to high dimensional spaces will hopefully lead to practical methods for solving hard and interesting real world control problems.

IX. Appendix

To avoid measure-theoretic complications, we do the proof only for the case when $\mathcal{U}$ is finite. Let $\Pi^0 \in \mathcal{P}[\mathcal{U}]$, f be any real-valued function over $\mathcal{U}$.

Lemma 1. *$min_{\mathbf{u}\in\mathcal{U}^{\mathbf{R}^+}} \left[\mathrm{KL}\left(\Pi^0 \otimes \mathbf{u} \parallel \Pi^0\right) - E_\pi[f]\right] = -\Psi_{\Pi^0}[f]$ with the min achieved at $\mathbf{u}^* = \exp(f)$.*

Proof:

$$\begin{aligned}
&\min_{\mathbf{u}} \mathrm{KL}\left(\mathbf{u}\otimes\Pi^0 \parallel \Pi^0\right) - \mathop{\mathrm{E}}_{\mathbf{u}\otimes\Pi^0}[f] \\
&= \min_{\mathbf{u}} \mathop{\mathrm{E}}_{\mathbf{u}\otimes\Pi^0}\left[\log\left(\frac{\mathbf{u}\otimes\Pi^0}{\Pi^0\exp(f)}\right)\right] \\
&= -\Psi_{\Pi^0(\mathbf{x})}[f] + \min_{\mathbf{u}} \mathrm{KL}\left(\Pi^0\otimes\mathbf{u} \parallel \Pi^0\otimes\exp(f)\right)
\end{aligned}$$

Since the KL divergence is minimized when the distributions are equal, choosing $\mathbf{u} = \exp(f)$ gives us the optimal value $-\Psi_{\Pi^0(\mathbf{x})}[f]$. ■

Lemma 2. *$min_{\mathbf{u}\in\mathcal{U}^{\mathbf{R}^+},\mathbf{u}>0} \left[\mathrm{KL}\left(\Pi^0 \parallel \Pi^0\otimes\mathbf{u}\right) + \Psi_{\Pi^0\otimes\mathbf{u}}[f]\right] = \mathrm{E}_{\Pi^0}[f]$ with the min achieved at $\mathbf{u}^* = \exp(-f)$.*

Proof: $\Psi_{\Pi^0\otimes\mathbf{u}}[f] = \log\left(\mathrm{E}_{\Pi^0\otimes\mathbf{u}}[\exp(f)]\right) = \log\left(\mathrm{E}_{\Pi^0}\left[\frac{\mathbf{u}\exp(f)}{\mathrm{E}_{\Pi^0}[\mathbf{u}]}\right]\right) \geq \mathrm{E}_{\Pi^0}\left[\log\left(\frac{\mathbf{u}\exp(f)}{\mathrm{E}_{\Pi^0}[\mathbf{u}]}\right)\right]$ by Jensen's inequality. This last term is equal to $\mathrm{E}_{\Pi^0}\left[\log\left(\frac{\mathbf{u}}{\mathrm{E}_{\Pi^0}[\mathbf{u}]}\right)\right] + \mathrm{E}_{\Pi^0}[f] = -\mathrm{KL}\left(\Pi^0 \parallel \Pi^0\otimes\mathbf{u}\right) + \mathrm{E}_{\Pi^0}[f]$. Thus, the objective is bounded below by $\mathrm{E}_{\Pi^0}[f]$ and this bound is achieved when $\mathbf{u} = \exp(-f)$. Hence the result. ■

References

[1] E. Todorov, "Efficient computation of optimal actions," *Proceedings of the National Academy of Sciences*, vol. 106, no. 28, p. 11478, 2009.

[2] H. Kappen, "Linear theory for control of nonlinear stochastic systems," *Physical Review Letters*, vol. 95, no. 20, p. 200201, 2005.

[3] E. Todorov, "Compositionality of optimal control laws," in *Advances in Neural Information Processing Systems 22*, Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, Eds., 2009, pp. 1856–1864.

[4] K. Dvijotham and E. Todorov, "Inverse optimal control with linearly-solvable mdps," in *International Conference on Machine Learning*, J. Fürnkranz and T. Joachims, Eds. Omnipress, 2010, pp. 335–342.

[5] M. Da Silva, F. Durand, and J. Popović, "Linear Bellman combination for control of character animation," *ACM Transactions on Graphics (TOG)*, vol. 28, no. 3, pp. 1–10, 2009.

[6] E. Theodorou, J. Buchli, and S. Schaal, "Reinforcement learning of motor skills in high dimensions: A path integral approach," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 2397–2403.

[7] K. Dvijotham and E. Todorov, "A unifying framework for linearly solvable control," in *Proceedings of the Proceedings of the Twenty-Seventh Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-11)*. Corvallis, Oregon: AUAI Press, 2011, pp. 179–186.

[8] T. Başar and P. Bernhard, *H-infinity optimal control and related minimax design problems: a dynamic game approach*. Birkhauser, 1995.
[9] T. Başar and G. Olsder, *Dynamic noncooperative game theory*. Society for Industrial Mathematics, 1999.
[10] R. Sutton and A. Barto, *Reinforcement learning: An introduction*. The MIT press, 1998.
[11] E. Todorov, "Stochastic optimal control and estimation methods adapted to the noise characteristics of the sensorimotor system," *Neural computation*, vol. 17, no. 5, pp. 1084–1108, 2005.
[12] P. Abbeel and A. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, p. 1.
[13] B. Ziebart, A. Maas, J. Bagnell, and A. Dey, "Maximum entropy inverse reinforcement learning," in *Proc. AAAI*, 2008, pp. 1433–1438.
[14] F. PM, "The information capacity of the human motor system in controlling the amplitude of movement." *Journal of experimental psychology*, vol. 47, no. 6, p. 381, 1954.
[15] T. Flash and N. Hogan, "The coordination of arm movements: an experimentally confirmed mathematical model," *Journal of neuroscience*, vol. 5, no. 7, p. 1688, 1985.