2011 IEEE International Symposium on Computer-Aided Control System Design (CACSD)
Part of 2011 IEEE Multi-Conference on Systems and Control
Denver, CO, USA. September 28-30, 2011

Min-Max Approximate Dynamic Programming

Brendan O'Donoghue Yang Wang Stephen Boyd

Abstract— **In this paper we describe an approximate dynamic programming policy for a discrete-time dynamical system perturbed by noise. The approximate value function is the pointwise supremum of a family of lower bounds on the value function of the stochastic control problem; evaluating the control policy involves the solution of a min-max or saddle-point problem. For a quadratically constrained linear quadratic control problem, evaluating the policy amounts to solving a semidefinite program at each time step. By evaluating the policy, we obtain a lower bound on the value function, which can be used to evaluate performance: When the lower bound and the achieved performance of the policy are close, we can conclude that the policy is nearly optimal. We describe several numerical examples where this is indeed the case.**

I. Introduction

We consider an infinite horizon stochastic control problem with discounted objective and full state information. In the general case this problem is difficult to solve, but exact solutions can be found for certain special cases. When the state and action spaces are finite, for example, the problem is readily solved. Another case for which the problem can be solved exactly is when the state and action spaces are finite dimensional real vector spaces, the system dynamics are linear, the cost function is convex quadratic, and there are no constraints on the action or the state. In this case optimal control policy is affine in the state variable, with coefficients that are readily computable [1], [2], [3], [4].

One general method for finding the optimal policy is to use dynamic programming (DP). DP represents the optimal policy in terms of an optimization problem involving the value function of the stochastic control problem [3], [4], [5]. However, due to the 'curse of dimensionality', even representing the value function can be intractable when the state or action spaces are infinite, or as a practical matter, when the number of states or actions is very large. Even when the value function can be represented, evaluating the optimal policy can still be intractable. As a result approximate dynamic programming (ADP) has been developed as a general method for finding suboptimal control policies [6], [7], [8]. In ADP we substitute an approximate value function for the value function in the expression for the optimal policy. The goal is to choose the approximate value function (also known as a control-Lyapunov function) so that the performance of the resulting policy is close to optimal, or at least, good.

In this paper we develop a control policy which we call the *min-max approximate dynamic programming policy*. We first parameterize a family of lower bounds on the true value function; then we perform control, taking the pointwise supremum over this family as our approximate value function. The condition we use to parameterize our family of bounds is related to the 'linear programming approach' to ADP, which was first introduced in [9], and extended to approximate dynamic programming in [10], [11]. The basic idea is that any function which satisfies the Bellman inequality is a lower bound on the true value function [3], [4].

It was shown in [12] that a better lower bound can be attained via an iterated chain of Bellman inequalities, which we use here. We relate this chain of inequalities to a forward look-ahead in time, in a similar sense to that of model predictive control (MPC) [13], [14]. Indeed many types of MPC can be thought of as performing min-max ADP with a particular (generally affine) family of underestimator functions.

In cases where we have a finite number of states and inputs, evaluating our policy requires solving a linear program at every time step. For problems with an infinite number of states and inputs, the method requires the solution of a semi-infinite linear program, with a finite number of variables, but an infinite number of constraints (one for every state-control pair). For these problems we can obtain a tractable semidefinite program (SDP) approximation using methods such as the $\mathcal{S}$-procedure [8], [12]. Evaluating our policy then requires solving an SDP at each time step [15], [16].

Much progress has been made in solving structured convex programs efficiently (see, *e.g.*, [17], [18], [19], [20]). These fast optimization methods make our policies practical, even for large problems, or those requiring fast sampling rates.

II. Stochastic control

Consider a discrete time-invariant dynamical system, with dynamics described by

$$x_{t+1} = f(x_t, u_t, w_t), \quad t = 0, 1, \ldots, \qquad (1)$$

978-1-4577-1061-2/11/$26.00 ©2011 IEEE

where $x_t \in \mathcal{X}$ is the system state, $u_t \in \mathcal{U}$ is the control input or action, $w_t \in \mathcal{W}$ is an exogenous noise or disturbance, at time t, and $f : \mathcal{X} \times \mathcal{U} \times \mathcal{W} \to \mathcal{X}$ is the state transition function. The noise terms w_t are independent identically distributed (IID), with known distribution. The initial state x_0 is also random with known distribution, and is independent of w_t. We consider causal, time-invariant state feedback control policies

$$u_t = \phi(x_t), \quad t = 0, 1, \ldots,$$

where $\phi : \mathcal{X} \to \mathcal{U}$ is the *control policy* or *state feedback function*.

The stage cost is given by $\ell : \mathcal{X} \times \mathcal{U} \to \mathbf{R} \cup \{+\infty\}$, where the infinite values of ℓ encode constraints on the states and inputs: The state-action *constraint set* $\mathcal{C} \subset \mathcal{X} \times \mathcal{U}$ is $\mathcal{C} = \{(z, v) \mid \ell(z, v) < \infty\}$. (The problem is unconstrained if $\mathcal{C} = \mathcal{X} \times \mathcal{U}$.)

The *stochastic control problem* is to choose ϕ in order to minimize the infinite horizon discounted cost

$$J_\phi = \mathbf{E} \sum_{t=0}^{\infty} \gamma^t \ell(x_t, \phi(x_t)), \tag{2}$$

where $\gamma \in (0, 1)$ is a discount factor. The expectations are over the noise terms w_t, $t = 0, 1, \ldots$, and the initial state x_0. We assume here that the expectation and limits exist, which is the case under various technical assumptions [3], [4]. We denote by $J^\star$ the optimal value of the stochastic control problem, *i.e.*, the infimum of J_ϕ over all policies $\phi : \mathcal{X} \to \mathcal{U}$.

A. Dynamic programming

In this section we briefly review the dynamic programming characterization of the solution to the stochastic control problem. For more details, see [3], [4].

The *value function* of the stochastic control problem, $V^\star : \mathcal{X} \to \mathbf{R} \cup \{\infty\}$, is given by

$$V^\star(z) = \inf_\phi \mathbf{E} \left(\sum_{t=0}^{\infty} \gamma^t \ell(x_t, \phi(x_t)) \right),$$

subject to the dynamics (1) and $x_0 = z$; the infimum is over all policies $\phi : \mathcal{X} \to \mathcal{U}$, and the expectation is over w_t for $t = 0, 1, \ldots$. The quantity $V^\star(z)$ is the cost incurred by an optimal policy, when the system is started from state z. The optimal total discounted cost is given by

$$J^\star = \mathop{\mathbf{E}}_{x_0} V^\star(x_0).$$

It can be shown that the value function is the unique fixed point of the Bellman equation

$$V^\star(z) = \inf_v \left(\ell(z, v) + \gamma \mathop{\mathbf{E}}_w V^\star(f(z, v, w)) \right)$$

for all $z \in \mathcal{X}$. We can write the Bellman equation in the form

$$V^\star = \mathcal{T} V^\star, \tag{3}$$

where we define the Bellman operator $\mathcal{T}$ as

$$(\mathcal{T} g)(z) = \inf_v \left(\ell(z, v) + \gamma \mathop{\mathbf{E}}_w g(f(z, v, w)) \right)$$

for any $g : \mathcal{X} \to \mathbf{R} \cup \{+\infty\}$.

An optimal policy for the stochastic control problem is given by

$$\phi^\star(z) = \operatorname*{argmin}_v \left(\ell(z, v) + \gamma \mathop{\mathbf{E}}_w V^\star(f(z, v, w)) \right), \tag{4}$$

for all $z \in \mathcal{X}$.

B. Approximate dynamic programming

In many cases of interest, it is intractable to compute (or even represent) the value function $V^\star$, let alone carry out the minimization required evaluate the optimal policy (4). In such cases, a common alternative is to replace the value function with an *approximate value function* $\hat{V}$ [6], [7], [8]. The resulting policy, given by

$$\hat{\phi}(z) = \operatorname*{argmin}_v \left(\ell(z, v) + \gamma \mathop{\mathbf{E}}_w \hat{V}(f(z, v, w)) \right),$$

for all $z \in \mathcal{X}$, is called an *approximate dynamic programming* (ADP) policy. Clearly, when $\hat{V} = V^\star$, the ADP policy is optimal. The goal of approximate dynamic programming is to find a $\hat{V}$ for which the ADP policy can be easily evaluated (for instance, by solving a convex optimization problem), and also attains near-optimal performance.

III. Min-max approximate dynamic programming

We consider a family of linearly parameterized (candidate) value functions $V_\alpha : \mathcal{X} \to \mathbf{R}$,

$$V_\alpha = \sum_{i=1}^{K} \alpha_i V^{(i)},$$

where $\alpha \in \mathbf{R}^K$ is a vector of coefficients and $V^{(i)} : \mathcal{X} \to \mathbf{R}$ are fixed basis functions. Now suppose we have a set $\mathcal{A} \subset \mathbf{R}^K$ for which

$$V_\alpha(z) \leq V^\star(z), \quad \forall z \in \mathcal{X}, \quad \forall \alpha \in \mathcal{A}.$$

Thus $\{V_\alpha \mid \alpha \in \mathcal{A}\}$ is a parameterized family of underestimators of the value function. (We will discuss how to obtain such a family later.) For any $\alpha \in \mathcal{A}$ we have

$$V_\alpha(z) \leq \sup_{\alpha \in \mathcal{A}} V_\alpha(z) \leq V^\star(z), \quad \forall z \in \mathcal{X},$$

i.e., the pointwise supremum over the family of underestimators must be at least as good an approximation of $V^\star$ as any single function from the family. This suggests the ADP control policy

$$\tilde{\phi}(z) = \operatorname*{argmin}_v \left(\ell(z,v) + \gamma \operatorname*{\mathbf{E}}_w \sup_{\alpha \in \mathcal{A}} V_\alpha(f(z,v,w)) \right),$$

where we use $\sup_{\alpha\in\mathcal{A}} V_\alpha(z)$ as an approximate value function. Unfortunately, this policy may be difficult to evaluate, since evaluating the expectation of the supremum can be hard, even when evaluating $\mathbf{E}\, V_\alpha(f(z,v,w))$ for a particular α can be done.

Our last step is to exchange expectation and supremum to obtain the *min-max control policy*

$$\begin{aligned}\phi^{\mathrm{mm}}(z) = \operatorname{argmin}_v \sup_{\alpha\in\mathcal{A}} (\ell(z,v) \\ +\gamma\, \mathbf{E}_w\, V_\alpha(f(z,v,w)))\end{aligned} \tag{5}$$

for all $z \in \mathcal{X}$. Computing this policy involves the solution of a min-max or saddle-point problem, which we will see is tractable in certain cases. One such case is where the function $\ell(z,v)+\mathbf{E}_w\, V_\alpha(f(z,v,w))$ is convex in v for each z and α and the set $\mathcal{A}$ is convex.

A. Bounds

The optimal value of the optimization problem in the min-max policy (5) is a lower on the value function at every state. To see this we note that

$$\begin{aligned}&\inf_v \sup_{\alpha\in\mathcal{A}} \left(\ell(z,v) + \gamma\, \mathbf{E}\, V_\alpha(f(z,u,w)) \right) \\ &\leq \inf_v \left(\ell(z,v) + \gamma\, \mathbf{E} \sup_{\alpha\in\mathcal{A}} V_\alpha(f(z,u,w)) \right) \\ &\leq \inf_v \left(\ell(z,v) + \gamma\, \mathbf{E}\, V^\star(f(z,u,w)) \right) \\ &= (\mathcal{T}V^\star)(z) = V^\star(z),\end{aligned}$$

where the first inequality is due to Fatou's lemma [21], the second inequality follows from the monotonicity of expectation, and the equality comes from the fact that $V^\star$ is the unique fixed point of the Bellman operator.

Using the pointwise bounds, we can evaluate a lower bound on the optimal cost $J^\star$ via Monte Carlo simulation:

$$J^{\mathrm{lb}} = (1/N) \sum_{j=1}^{N} V^{\mathrm{lb}}(z_j)$$

where $z_1, \ldots, z_N$ are drawn from the same distribution as x_0 and $V^{\mathrm{lb}}(z_j)$ is the lower bound we get from evaluating the min-max policy at z_j.

The performance of the min-max policy can also be evaluated using Monte Carlo simulation, and provides an upper bound J^{ub} on the optimal cost. Ignoring Monte Carlo error we have

$$J^{\mathrm{lb}} \leq J^\star \leq J^{\mathrm{ub}}.$$

Theses upper and lower bounds on the optimal value of the stochastic control problem are readily evaluated numerically, through simulation of the min-max control policy. When J^{lb} and J^{ub} are close, we can conclude that the min-max policy is almost optimal. We will use this technique to evaluate the performance of the min-max policy for our numerical examples.

B. Evaluating the min-max control policy

Evaluating the min-max control policy often requires exchanging the order of minimization and maximization. For any function $f : \mathbf{R}^p \times \mathbf{R}^q \to \mathbf{R}$ and sets $\mathcal{W} \subseteq \mathbf{R}^p$, $\mathcal{Z} \subseteq \mathbf{R}^q$, the *max-min inequality* states that

$$\sup_{z\in\mathcal{Z}} \inf_{w\in\mathcal{W}} f(w,z) \leq \inf_{w\in\mathcal{W}} \sup_{z\in\mathcal{Z}} f(w,z). \tag{6}$$

In the context of the min-max control policy, this means we can swap the order of minimization and maximization in (5) and maintain the lower bound property. To evaluate the policy, we solve the optimization problem

$$\begin{array}{ll}\text{maximize} & \inf_v \left(\ell(z,v) + \gamma\, \mathbf{E}_w\, V_\alpha(f(z,v,w))\right) \\ \text{subject to} & \alpha \in \mathcal{A}\end{array} \tag{7}$$

with variable α. If $\mathcal{A}$ is a convex set, (7) is a convex optimization problem, since the objective is the infimum over a family of affine functions in α, and is therefore concave. In practice, solving (7) is often much easier than evaluating the min-max control policy directly.

In addition, if there exist $\tilde{w} \in \mathcal{W}$ and $\tilde{z} \in \mathcal{Z}$ such that

$$f(\tilde{w}, z) \leq f(\tilde{w}, \tilde{z}) \leq f(w, \tilde{z}),$$

for all $w \in \mathcal{W}$ and $z \in \mathcal{Z}$, then we have the *strong max-min property* (or *saddle-point property*) and (6) holds with equality. In such cases the problems (5) and (7) are equivalent, and we can use Newton's method or duality considerations to solve (5) or (7) [16], [22].

IV. Iterated Bellman inequalities

In this section we describe how to parameterize a family of underestimators of the true value function. The idea is based on the Bellman inequality, [10], [8], [12], and results in a convex condition on the coefficients α that guarantees $V_\alpha \leq V^\star$.

A. Basic Bellman inequality

The basic condition works as follows. Suppose we have a function $V : \mathcal{X} \to \mathbf{R}$, which satisfies the Bellman inequality

$$V \leq \mathcal{T}V. \tag{8}$$

Then by the monotonicity of the Bellman operator, we have

$$V \leq \lim_{k\to\infty} \mathcal{T}^k V = V^\star,$$

so any function that satisfies the Bellman inequality must be a value function underestimator. Applying this condition to V_α and expanding (8) we get

$$V_\alpha(z) \leq \inf_v \left(\ell(z,v) + \gamma \mathbf{E}_w V_\alpha(f(z,v,w))\right),$$

for all $z \in \mathcal{X}$. For each z, the left hand side is linear in α, and the right hand side is a concave function of α, since it is the infimum over a family of affine functions. Hence, the Bellman inequality leads to a convex constraint on α.

B. Iterated Bellman inequalities

We can obtain better (*i.e.*, larger) lower bounds on the value function by considering an iterated form of the Bellman inequality [12]. Suppose we have a sequence of functions $V_i : \mathcal{X} \to \mathbf{R}$, $i = 0, \ldots, M$, that satisfy a chain of Bellman inequalities

$$V_0 \leq \mathcal{T}V_1, \quad V_1 \leq \mathcal{T}V_2, \quad \ldots \quad V_{M-1} \leq \mathcal{T}V_M, \tag{9}$$

with $V_{M-1} = V_M$. Then, using similar arguments as before we can show $V_0 \leq V^\star$. Restricting each function to lie in the same subspace

$$V_i = \sum_{j=1}^K \alpha_{ij} V^{(j)},$$

we see that the iterated chain of Bellman inequalities also results in a convex constraint on the coefficients α_{ij}. Hence the condition on α_{0j}, $j = 1, \ldots, K$, which parameterizes our underestimator V_0, is convex. It is easy to see that the iterated Bellman condition must give bounds that are at least as good as the basic Bellman inequality, since any function that satisfies (8) must be feasible for (9) [12].

V. Box constrained linear quadratic control

This section follows a similar example presented in [12]. We have $\mathcal{X} = \mathbf{R}^n$, $\mathcal{U} = \mathbf{R}^m$, with linear dynamics

$$x_{t+1} = Ax_t + Bu_t + w_t,$$

where $A \in \mathbf{R}^{n\times n}$ and $B \in \mathbf{R}^{n\times m}$. The noise has zero mean, $\mathbf{E}\, w_t = 0$, and covariance $\mathbf{E}\, w_t w_t^T = W$. (Our bounds and policy will only depend on the first and second moments of w_t.) The stage cost is given by

$$\ell(z,v) = \begin{cases} v^T R v + z^T Q z, & \|v\|_\infty \leq 1 \\ +\infty, & \|v\|_\infty > 1, \end{cases}$$

where $R = R^T \succeq 0$, $Q = Q^T \succeq 0$.

A. Iterated Bellman inequalities

We look for convex quadratic approximate value functions

$$V_i(z) = z^T P_i z + 2p_i^T z + s_i, \quad i = 0, \ldots, M,$$

where $P_i = P_i^T \succeq 0$, $p_i \in \mathbf{R}^n$ and $r_i \in \mathbf{R}$, are the coefficients of our linear parameterization. The iterated Bellman inequalities are

$$V_{i-1}(z) \leq \ell(z,v) + \gamma \mathbf{E}\, V_i(Az + Bv + w),$$

for all $\|v\|_\infty \leq 1$, $z \in \mathbf{R}^n$, $i = 1, \ldots, M$. Defining

$$S_i = \begin{bmatrix} 0 & 0 & 0 \\ 0 & P_i & p_i \\ 0 & p_i^T & s_i \end{bmatrix}, \quad L = \begin{bmatrix} R & 0 & 0 \\ 0 & Q & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$G_i = \begin{bmatrix} B^T P_i B & B^T P_i A & B^T p_i \\ A^T P_i B & A^T P_i A & A^T p_i \\ p_i^T B & p_i^T A & \mathbf{Tr}(P_i W) + s_i \end{bmatrix},$$

for $i = 0, \ldots, M$, we can write the Bellman inequalities as a quadratic form in $(v, z, 1)$

$$\begin{bmatrix} v \\ z \\ 1 \end{bmatrix}^T (L + \gamma G_i - S_{i-1}) \begin{bmatrix} v \\ z \\ 1 \end{bmatrix} \geq 0, \tag{10}$$

for all $\|v\|_\infty \leq 1$, $z \in \mathbf{R}^n$, $i = 1, \ldots, M$.

We will obtain a tractable sufficient condition for this using the $\mathcal{S}$-procedure [15], [12]. The constraint $\|v\|_\infty \leq 1$ can be written in terms of quadratic inequalities

$$1 - v^T (e_i e_i^T) v \geq 0, \quad i = 1, \ldots, m,$$

where e_i denotes the ith unit vector. Using the $\mathcal{S}$-procedure, a sufficient condition for (10) is the existence of diagonal matrices $D_i \succeq 0$, $i = 1, \ldots, M$ for which

$$L + \gamma G_i - S_{i-1} + \Lambda_i \succeq 0, \quad i = 1, \ldots M, \tag{11}$$

where

$$\Lambda_i = \begin{bmatrix} D_i & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -\mathbf{Tr}(D_i) \end{bmatrix}. \tag{12}$$

Finally we have the terminal constraint, $S_{M-1} = S_M$.

B. Min-max control policy

For this problem, it is easy to show that the strong max-min property holds, and therefore problems (5) and (7) are equivalent. To evaluate the min-max control policy we solve problem (7), which we can write as

$$\begin{array}{ll} \text{maximize} & \inf_v(\ell(z,v)+\gamma\,\mathbf{E}_w\,V_0(Az+Bv+w)) \\ \text{subject to} & (11), \quad S_{M-1}=S_M, \quad P_0\succeq 0 \\ & P_i\succeq 0, \quad D_i\succeq 0, \quad i=1,\ldots,M, \end{array}$$

with variables P_i, p_i, s_i, $i=0,\ldots,M$, and diagonal D_i, $i=1,\ldots,M$. We will convert this max-min problem to a max-max problem by forming the dual of the minimization part. Introducing a diagonal matrix $D_0\succeq 0$ as the dual variable for the box constraints, we obtain the dual function

$$\inf_v \begin{bmatrix} v \\ z \\ 1 \end{bmatrix}^T (L+\gamma G_0-\Lambda_0) \begin{bmatrix} v \\ z \\ 1 \end{bmatrix},$$

where Λ_0 has the form given in (12). We can minimize over v analytically. If we block out the matrix $L+\gamma G_0-\Lambda_0$ as

$$(L+\gamma G_0-\Lambda_0)=\begin{bmatrix} M_{11} & M_{12} \\ M_{12}^T & M_{22} \end{bmatrix} \tag{13}$$

where $M_{11}\in\mathbf{R}^{m\times m}$, then

$$v^\star=-M_{11}^{-1}M_{12}\begin{bmatrix} z \\ 1 \end{bmatrix}.$$

Thus our problem becomes

$$\begin{array}{ll} \text{maximize} & \begin{bmatrix} z \\ 1 \end{bmatrix}(M_{22}-M_{12}^TM_{11}^{-1}M_{12})\begin{bmatrix} z \\ 1 \end{bmatrix} \\ \text{subject to} & (11), \quad S_{M-1}=S_M \\ & P_i\succeq 0, \quad D_i\succeq 0, \quad i=0,\ldots,M, \end{array}$$

which is a convex optimization problem in the variables P_i, p_i, r_i, D_i, $i=0,\ldots,M$, and can be solved as an SDP.

To implement the min-max control policy, at each time t, we solve the above problem with $z=x_t$, and let

$$u_t=-M_{11}^{\star-1}M_{12}^\star\begin{bmatrix} x_t \\ 1 \end{bmatrix},$$

where $M_{11}^\star$ and $M_{12}^\star$ denote the matrices M_{11} and M_{12}, computed from $P_0^\star$, $p_0^\star$, $s_0^\star$, $D_0^\star$.

C. Interpretations

We can easily verify that the dual of the above optimization problem is a variant of model predictive control, that uses both the first and second moments of the state. In this context, the number of iterations, M, is the length of the prediction horizon, and we can interpret our lower bound as a finite horizon approximation to an infinite horizon problem, which underestimates the optimal infinite horizon cost. The $\mathcal{S}$-procedure relaxation also has a natural interpretation: in [23], the author obtains similar LMIs by relaxing almost sure constraints into constraints that are only required to hold in expectation.

Policy / Bound	Value
MPC policy	1.3147
Min-max policy	1.3145
Lower bound	1.3017

TABLE I: Performance comparison, box constrained example.

D. Numerical instance

We consider a numerical example with $n=8$, $m=3$, and $\gamma=0.9$. The parameters Q, R, A and B are randomly generated; we set $\|B\|=1$ and scale A so that $\max|\lambda_i(A)|=1$ (*i.e.*, so that the system is marginally stable). The initial state x_0 is Gaussian, with zero mean.

Table I shows the performance of the min-max policy and certainty equivalent MPC, both with horizons of $M=15$ steps, as well as the lower bound on the optimal cost. In this case, both the min-max policy and MPC are no more than around 1% suboptimal, modulo Monte Carlo error.

VI. Dynamic portfolio optimization

In this example, we manage a portfolio of n assets over time. Our state $x_t\in\mathbf{R}^n$ is the vector of dollar values of the assets, at the beginning of investment period t. Our action $u_t\in\mathbf{R}^n$ represents buying or selling assets at the beginning of each investment period: $(u_t)_i>0$ means we are buying asset i, for dollar value $(u_t)_i$, $(u_t)_i<0$ means we sell asset i. The post-trade portfolio is then given by x_t+u_t, and the total gross cash put in is $\mathbf{1}^Tu_t$, where $\mathbf{1}$ is the vector with all components one.

The portfolio propagates over time (*i.e.*, over the investment period) according to

$$x_{t+1}=A_t(x_t+u_t)$$

where $A_t=\mathbf{diag}(\rho_t)$, and $(\rho_t)_i$ is the (total) return of asset i in investment period t. The return vectors ρ_t are IID, with first and second moments

$$\mathbf{E}(\rho_t)=\mu, \qquad \mathbf{E}(\rho_t\rho_t^T)=\Sigma.$$

(Here too, our bounds and policy will only depend on the first and second moments of ρ_t.) We let $\hat{\Sigma} = \Sigma - \mu\mu^T$ denote the return covariance.

We now describe the constraints and objective. We constrain the risk of our post-trade portfolio, which we quantify as the portfolio return variance over the period:

$$(x_t + u_t)^T \hat{\Sigma}(x_t + u_t) \leq l,$$

where $l \geq 0$ is the maximum variance (risk) allowed. Our action (buying and selling) u_t incurs a transaction cost with an absolute value and a quadratic component, given by $\kappa^T|u| + u^T R u$, where $\kappa \in \mathbf{R}^n_+$ is the vector of linear transaction cost rates (and $|u|$ means elementwise), and $R \in \mathbf{R}^{n\times n}$, which is diagonal with positive entries, represents a quadratic transaction cost coefficients. (Linear transactions cost model effects such as crossing the bid-ask spread, while quadratic transaction costs model effects such as price-impact.) Thus at time t, we put into our portfolio the net cash amount

$$g(u_t) = \mathbf{1}^T u_t + \kappa^T |u_t| + u_t^T R u_t.$$

(When this is negative, it represents revenue.) The first term is the gross cash in from purchases and sales; the second and third terms are the transaction fees. The stage cost, including the risk limit, is

$$\ell(z, v) = \begin{cases} g(v), & (z+v)^T\hat{\Sigma}(z+v) \leq l \\ +\infty, & \text{otherwise.} \end{cases}$$

Our goal is to minimize the discounted cost (or equivalently, to maximize the discounted revenue). In this example, the discount factor has a natural interpretation as reflecting the time value of money.

A. Iterated Bellman Inequalities

We incorporate another variable, $y \in \mathbf{R}^n$, to remove the absolute value term from the stage cost function, and add the constraints

$$-(y_t)_i \leq (v_t)_i \leq (y_t)_i, \quad i = 1, \ldots, n. \tag{14}$$

We define the stage cost with these new variables to be

$$\ell(z, v, y) = \mathbf{1}^T v + \kappa^T y + (1/2) v^T R v + (1/2) y^T R y$$

for (z, v, y) that satisfy

$$-y \leq v \leq y, \quad (z+v)\hat{\Sigma}(z+v) \leq l, \tag{15}$$

and $+\infty$ otherwise. Here, the first set of inequalities is interpreted elementwise.

We look for convex quadratic candidate value functions, *i.e.*

$$V_i(z) = z^T P_i z + 2 p_i^T z + s_i, \quad i = 0, \ldots, M,$$

where $P_i \succeq 0$, $p_i \in \mathbf{R}^n$, $s_i \in \mathbf{R}$ are the coefficients of our linear parameterization. Defining

$$L = \begin{bmatrix} R/2 & 0 & 0 & \mathbf{1}/2 \\ 0 & R/2 & 0 & \kappa/2 \\ 0 & 0 & 0 & 0 \\ \mathbf{1}^T/2 & \kappa^T/2 & 0 & 0 \end{bmatrix},$$

$$G_i = \begin{bmatrix} P_i \circ \Sigma & 0 & P_i \circ \Sigma & p_i \circ \mu \\ 0 & 0 & 0 & 0 \\ P_i \circ \Sigma & 0 & P_i \circ \Sigma & p_i \circ \mu \\ (p_i \circ \mu)^T & 0 & (p_i \circ \mu)^T & 0 \end{bmatrix},$$

$$S_i = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & P_i & p_i \\ 0 & 0 & p_i^T & s_i \end{bmatrix}, \quad i = 0, \ldots, M,$$

where $\circ$ denotes the Hadamard product, we can write the iterated Bellman inequalities as

$$\begin{bmatrix} v \\ y \\ z \\ 1 \end{bmatrix}^T (L + \gamma G_i - S_{i-1}) \begin{bmatrix} v \\ y \\ z \\ 1 \end{bmatrix} \geq 0,$$

for all (v, y, z) that satisfy (15), for $i = 1, \ldots, M$.

A tractable sufficient condition for the Bellman inequalities is (by the $\mathcal{S}$-procedure) the existence of $\lambda_i \geq 0$, $\nu_i \in \mathbf{R}^n_+$, $\tau_i \in \mathbf{R}^n_+$, $i = 1, \ldots, M$ such that

$$L + \gamma G_i - S_{i-1} + \Lambda_i \succeq 0, \quad i = 1, \ldots, M, \tag{16}$$

where

$$\Lambda_i = \begin{bmatrix} \lambda_i\hat{\Sigma} & 0 & \lambda_i\hat{\Sigma} & \nu_i - \tau_i \\ 0 & 0 & 0 & \nu_i + \tau_i \\ \lambda_i\hat{\Sigma} & 0 & \lambda_i\hat{\Sigma} & 0 \\ \nu_i^T - \tau_i^T & \nu_i^T + \tau_i^T & 0 & -\lambda_i l \end{bmatrix}. \tag{17}$$

Lastly we have the terminal constraint, $S_{M-1} = S_M$.

B. Min-max control policy

The discussion here follows almost exactly the one presented for the previous example. It is easy to show that in this case we have the strong max-min property. At each step, we solve problem (7) by converting the max-min problem into a max-max problem, using Lagrangian duality. We can write problem (7) as

$$\begin{array}{ll} \text{maximize} & \inf_{v,y}(\ell(z, v, y) + \gamma\, \mathbf{E}_\rho\, V_0(A(z+v))) \\ \text{subject to} & (16), \quad S_{M-1} = S_M \\ & P_i \succeq 0, \quad i = 0, \ldots, M, \end{array}$$

with variables P_i, p_i, s_i, $i = 0, \ldots, M$, and $\lambda_i \in \mathbf{R}_+$, $\nu_i \in \mathbf{R}^n_+$, $\tau_i \in \mathbf{R}^n_+$, $i = 1, \ldots, M$.

Next, we derive the dual function of the minimization part. We introduce variables $\lambda_0 \in \mathbf{R}_+$, $\nu_0 \in \mathbf{R}^n_+$, $\tau_0 \in \mathbf{R}^n_+$, which are dual variables corresponding to the constraints (15) and (14). The dual function is given by

$$\inf_{v,y} \begin{bmatrix} v \\ y \\ z \\ 1 \end{bmatrix}^T (L + \gamma G_0 + \Lambda_0) \begin{bmatrix} v \\ y \\ z \\ 1 \end{bmatrix},$$

where Λ_0 has the form given in (17). If we define

$$L + \gamma G_0 + \Lambda_0 = \begin{bmatrix} M_{11} & M_{12} \\ M_{12}^T & M_{22} \end{bmatrix}, \qquad (18)$$

where $M_{11} \in \mathbf{R}^{2n \times 2n}$, then the minimizer of the Lagrangian is

$$(v^\star, y^\star) = -M_{11}^{-1} M_{12} \begin{bmatrix} z \\ 1 \end{bmatrix}.$$

Thus our problem becomes

$$\begin{array}{ll} \text{maximize} & \begin{bmatrix} z \\ 1 \end{bmatrix} (M_{22} - M_{12}^T M_{11}^{-1} M_{12}) \begin{bmatrix} z \\ 1 \end{bmatrix} \\ \text{subject to} & (16), \quad S_{M-1} = S_M \\ & P_i \succeq 0, \quad i = 0, \ldots, M, \end{array}$$

which is convex in the variables P_i, p_i, s_i, $i = 0, \ldots, M$, and $\lambda_i \in \mathbf{R}_+$, $\nu_i \in \mathbf{R}^n_+$, $\tau_i \in \mathbf{R}^n_+$, $i = 0, \ldots, M$.

To implement the policy, at each time t we solve the above optimization problem (as an SDP) with $z = x_t$, and let

$$u_t = -\begin{bmatrix} I_m & 0 \end{bmatrix} M_{11}^{\star -1} M_{12}^\star \begin{bmatrix} x_t \\ 1 \end{bmatrix},$$

where $M_{11}^\star$ and $M_{12}^\star$ denote the matrices M_{11} and M_{12}, computed from $P_0^\star$, $p_0^\star$, $s_0^\star$, $\lambda_0^\star$, $\nu_0^\star$, $\tau_0^\star$.

C. Numerical instance

We consider a numerical example with $n = 8$ assets and $\gamma = 0.96$. The initial portfolio x_0 is Gaussian, with zero mean. The returns follow a log-normal distribution, *i.e.*, $\log(\rho_t) \sim \mathcal{N}(\tilde{\mu}, \tilde{\Sigma})$. The parameters μ and Σ are given by

$$\mu_i = \exp(\tilde{\mu}_i + \tilde{\Sigma}_{ii}/2), \quad \Sigma_{ij} = \mu_i \mu_j \exp(\tilde{\Sigma}_{ij}).$$

Table II compares the performance of the min-max policy for $M = 40$ and $M = 5$, and certainty equivalent model predictive control with horizon $T = 40$, over 1150 simulations each consisting of 150 time steps. We can see that the min-max policy significantly outperforms MPC, which actually makes a loss on average (since the average cost is positive). The cost achieved by the min-max policy is close to the lower bound, which shows that both the policy and the bound are nearly optimal. In fact, the gap is small even for $M = 5$, which corresponds to a relatively myopic policy.

Bound / Policy	**Value**
MPC policy, $T = 40$	25.4
Min-max policy, $M = 5$	-224.1
Min-max policy, $M = 40$	-225.1
Lower bound, $M = 40$	-239.9
Lower bound, $M = 5$	-242.0

TABLE II: Performance comparison, portfolio example.

VII. Conclusions

In this paper we introduce a control policy which we refer to as *min-max approximate dynamic programming*. Evaluating this policy at each time step requires the solution of a min-max or saddle point problem; in addition, we obtain a lower bound on the value function, which can used to estimate (via Monte Carlo simulation) the optimal value of the stochastic control problem.

We demonstrate the method with two examples, where the policy can be evaluated by solving a convex optimization problem at each time-step. In both examples the lower bound and the achieved performance are very close, certifying that the min-max policy is very close to optimal.

References

[1] R. Kalman, "When is a linear control system optimal?" *Journal of Basic Engineering*, vol. 86, no. 1, pp. 1–10, 1964.

[2] S. Boyd and C. Barratt, *Linear controller design: Limits of performance*. Prentice-Hall, 1991.

[3] D. Bertsekas, *Dynamic Programming and Optimal Control: Volume 1*. Athena Scientific, 2005.

[4] ——, *Dynamic Programming and Optimal Control: Volume 2*. Athena Scientific, 2007.

[5] D. Bertsekas and S. Shreve, *Stochastic optimal control: The discrete-time case*. Athena Scientific, 1996.

[6] D. Bertsekas and J. Tsitsiklis, *Neuro-Dynamic Programming*, 1st ed. Athena Scientific, 1996.

[7] W. Powell, *Approximate dynamic programming: solving the curses of dimensionality*. John Wiley & Sons, Inc., 2007.

[8] Y. Wang and S. Boyd, "Performance bounds for linear stochastic control," *Systems & Control Letters*, vol. 58, no. 3, pp. 178–182, Mar. 2009.

[9] A. Manne, "Linear programming and sequential decisions," *Management Science*, vol. 6, no. 3, pp. 259–267, 1960.

[10] D. De Farias and B. Van Roy, "The linear programming approach to approximate dynamic programming," *Operations Research*, vol. 51, no. 6, pp. 850–865, 2003.

[11] P. Schweitzer and A. Seidmann, "Generalized polynomial approximations in markovian decision process," *Journal of mathematical analysis and applications*, vol. 110, no. 2, pp. 568–582, 1985.
[12] Y. Wang and S. Boyd, "Approximate dynamic programming via iterated bellman inequalities," 2010, manuscript.
[13] C. Garcia, D. Prett, and M. Morari, "Model predictive control: theory and practice," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.
[14] J. Maciejowski, *Predictive Control with Constraints*. Prentice-Hall, 2002.
[15] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*. Society for Industrial Mathematics, 1994.
[16] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, Sept. 2004.
[17] Y. Wang and S. Boyd, "Fast model predictive control using online optimization," *IEEE Transactions on Control Systems Technology*, vol. 18, pp. 267–278, 2010.
[18] ——, "Fast evaluation of quadratic control-lyapunov policy," *IEEE Transactions on Control Systems Technology*, pp. 1–8, 2010.
[19] J. Mattingley, Y. Wang, and S. Boyd, "Code generation for receding horizon control," in *IEEE Multi-Conference on Systems and Control*, 2010, pp. 985–992.
[20] J. Mattingley and S. Boyd, "CVXGEN: A code generator for embedded convex optimization," 2010, manuscript.
[21] D. Cohn, *Measure Theory*. Birkhäuser, 1997.
[22] A. Ben-Tal, L. E. Ghaoui, and A. Nemirovski, *Robust optimization*. Princeton University Press, 2009.
[23] A. Gattami, "Generalized linear quadratic control theory," *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 1510–1514, 2006.