

State Aggregation based Linear Programming approach to Approximate Dynamic Programming

S. Darbha, K. Krishnamoorthy, M. Pachter and P. Chandler

Abstract—One often encounters the curse of dimensionality in the application of dynamic programming to determine optimal policies for controlled Markov chains. In this paper, we provide a method to construct sub-optimal policies along with a bound for the deviation of such a policy from the optimum through the use of restricted linear programming. The novelty of this approach lies in circumventing the need for a value iteration or a linear program defined on the entire state-space. Instead, the state-space is partitioned based on the reward structure and the optimal cost-to-go or value function is approximated by a constant over each partition. We associate a meta-state with each partition, where the transition probabilities between these meta-states can be derived from the original Markov chain specification. The state aggregation approach results in a significant reduction in the computational burden and lends itself to a restricted linear program defined on the aggregated state-space. Finally, the proposed method is bench marked on a perimeter surveillance stochastic control problem.

I. INTRODUCTION

The Linear Programming (LP) approach to solving dynamic programs originated from the papers [1], [2], [3], [4]. But the *curse of dimensionality* has rendered it impossible to use exact LP methods to solve large scale stochastic control problems. This has motivated development of LP techniques that give tractable approximate solutions instead [5], [6], [7]. The advantages of approaching approximate dynamic programming through linear programming can be summarized as follows:

- 1) One can restrict the value function to be of a certain form, thereby reducing the dimension of the linear program so as to make it tractable.
- 2) While computational tractability alone is not a significant advantage, the solution of the LP automatically provides bounds for the value function. Of course, the quality of bounds depends on the restrictions imposed on the value function. If they are consistent with the physics of the problem, one can expect reasonably tight bounds.

Corresponding author: K. Krishnamoorthy krishnak@ucla.edu

This research was performed while the corresponding author held a National Research Council Research Associateship Award

Approved for public release; distribution unlimited, case number: 88ABW-2010-1008

S. Darbha is with the Department of Mechanical Engineering, Texas A&M University, College Station, TX 77843, USA

K. Krishnamoorthy is a Visiting Scientist with the Control Design & Analysis Branch, Air Force Research Laboratory, Wright-Patterson AFB, OH 45433, USA

M. Pachter is with the Department of Electrical Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH 45433, USA

P. Chandler is with the Control Design & Analysis Branch, Air Force Research Laboratory, Wright-Patterson AFB, OH 45433, USA

It is well known that sometimes the linear program used in these techniques may be overly constrained leading to poor approximations and perhaps even infeasibility [8]. An efficient and therefore attractive basis function approach with guaranteed feasibility and bounds on the approximation error is proposed in [9]. Unfortunately it is not clear how one should go about choosing the basis functions for a generic problem. Within the basis function framework, we propose to partition the state space and compute approximate value functions associated with each partition. In particular, we use a state aggregation scheme wherein the optimal cost-to-go (value function) is approximated by a constant over each partition [10]. State aggregation based techniques have been widely used in the past to derive approximate solutions [11], [12], [13], [14]. These methods typically involve partitioning the state space and associating a meta-state with each partition. It is required that the transition probabilities between these meta-states be derived from the original Markov chain specification. The state-space partitioning would of course have to exploit any inherent structure in the problem under consideration. We explain the architecture behind the aggregation based restricted LP approach and also demonstrate its usefulness on a (inherently large scale) perimeter surveillance stochastic control problem.

II. STOCHASTIC DYNAMIC PROGRAMMING

Let us consider discrete-time stochastic control problems involving a finite state space $\mathcal{S} = \{1, \dots, |\mathcal{S}|\}$ and a finite set of control actions, \mathcal{U} . From current state s , taking action $u \in \mathcal{U}$ under the random influence y , the agent gains a reward $r(s, u)$ and transitions to \bar{s} following some discrete-time dynamics given by

$$\bar{s} = f(s, u, y)$$

We assume that the random input can only assume values from a finite set \mathcal{Y} and there is a probability associated with each choice. Now a policy $\pi \in \mathbb{R}^{|\mathcal{S}|}$ is a mapping from states to actions. We are interested in a stationary policy that maximizes the expected infinite-horizon discounted reward for each state $s \in \mathcal{S}$ given by,

$$\mathbf{E} \left\{ \sum_{k=0}^{\infty} \lambda^k r(s_k, \pi(s_k)) \mid s_0 = s \right\}$$

where the temporal discount factor $\lambda \in (0, 1)$ and k indicates time. Bellman's equation precisely provides the answer to this and requires the solution of the following equation: for

every $s \in \mathcal{S}$, solve for the optimal value function $V^* \in \mathbb{R}^{|\mathcal{S}|}$ [15]

$$V^*(s) = \max_{u \in \mathcal{U}} \left\{ r(s, u) + \lambda \sum_i p_i V^*(f(s, u, y_i)) \right\} \quad (1)$$

where p_i is the probability that the random event takes the value y_i . The transition probabilities will of course have to satisfy $\sum_{i=1}^{|\mathcal{Y}|} p_i = 1$. The optimal control $\pi^*(s)$ is the action that achieves the maximum in the above equation. This problem can be exactly solved using standard approaches such as *value iteration* and *policy iteration* [15], [16]. However, it is computationally expensive when the size of state space encountered becomes prohibitively large. For this reason, we would be interested in other approximate methods. Ideally, we would like to determine an optimal stationary policy. In cases where it is not computationally tractable, we would like to find a suboptimal policy with some guarantees on the deviation of the associated value function from the optimal.

III. LINEAR PROGRAMMING APPROACH

Bellman's equation (1) implies that the optimal value function automatically satisfies the following set of linear inequalities:

$$V^*(s) \geq r(s, u) + \lambda \sum_i p_i V^*(f(s, u, y_i)), \quad \forall s \in \mathcal{S}, \forall u \in \mathcal{U}$$

which may be compactly represented as

$$(I - \lambda P_u) V^* \geq R_u, \quad \forall u \in \mathcal{U} \quad (2)$$

where the i^{th} entry of the reward vector, $R_u(i) = r(i, u)$ and the $(i, j)^{th}$ entry of the transition probability matrix P_u is the probability of going from state i to state j under the influence of action u . In fact, it can be shown that any vector $V \in \mathbb{R}^{|\mathcal{S}|}$ that satisfies (2) is an automatic upper bound to the optimal value function. Furthermore, the optimal value function V^* is the unique solution to the LP [1], [2]

$$\begin{aligned} \min \quad & c^T V \quad s.t. \\ (I - \lambda P_u) V & \geq R_u \quad \forall u \in \mathcal{U} \end{aligned} \quad (3)$$

where the entries of the cost vector c are all positive. Every feasible solution $V \in \mathbb{R}^{|\mathcal{S}|}$ to (3) satisfies $V \geq V^*$. It follows that, for any positive weight vector c , V^* is the unique solution to (3). Similarly, one can also construct a lower bounding LP as follows:

$$\begin{aligned} \max \quad & c^T V \quad s.t. \\ (I - \lambda P_u) V & \leq R_u \quad \forall u \end{aligned} \quad (4)$$

In this case, every $V \in \mathbb{R}^{|\mathcal{S}|}$ satisfying the inequality

$$(I - \lambda P_u) V \leq R_u, \quad \forall u \in \mathcal{U}$$

provides a lower bound for the optimal value function. In fact, it is no more than the value function associated with every admissible policy π and hence this bound can be very conservative.

A. Approximate Value Function by State Aggregation

Let the state space be a union of disjoint sets,

$$\mathcal{S} = \cup_{i=1}^n \mathcal{S}_i, \quad i = 1, \dots, n$$

We will call the set \mathcal{S}_i as the i^{th} partition. We assume that the reward (stage cost) depends only on the partition and control input and not the individual states. So for all states $s \in \mathcal{S}_i$, $r(s, u) = r_i(u)$ for all $i = 1, \dots, n$. We will use the following notation: If $f(s, u, y_k)$ represents the future state starting from s and subject to a control input u and a stochastic disturbance y_k , then $\bar{f}(s, u, y_k)$ represents the partition to which the future state belongs. We will call a partitioning to be an A-type partitioning if $\bar{f}(s, u, y_k)$ is independent of y_k , i.e., all future states starting from any given state s under the influence of u belong to the same partition irrespective of the disturbance y_k . Otherwise, we call it a B-type partitioning. We will use $\mathcal{T}(i, u, y_k)$ to represent the set of all partitions to which future states starting from \mathcal{S}_i transition to under the influence of u and y_k . For an A-type partitioning, $\mathcal{T}(i, u, y_k)$ is independent of y_k and may simply be represented by $\mathcal{T}(i, u)$.

Now we recall the original LP

$$\begin{aligned} \min \quad & c^T V \quad s.t. \\ (I - \lambda P_u) V & \geq R_u \quad \forall u \end{aligned} \quad (5)$$

and further require that $V(s) = a(i)$ for all $s \in \mathcal{S}_i$ i.e., approximate the value function over each partition by a constant ($a \in \mathbb{R}^n$). Augmenting these constraints, one gets the restricted LP (RLP):

$$\begin{aligned} \min \quad & c^T \Phi a \quad s.t. \\ (I - \lambda P_u) \Phi a & \geq R_u \quad \forall u \end{aligned} \quad (6)$$

Where the matrix of basis functions $\Phi = [e_1 \ e_2 \ \dots \ e_n]$ (using notation in [9]). The orthogonal basis functions are defined by,

$$e_i(s) = \begin{cases} 1, & s \in \mathcal{S}_i \\ 0, & \text{otherwise} \end{cases}, \quad s = 1 \dots |\mathcal{S}|, \forall i$$

For clarity we rewrite the restricted LP (6) in expanded form:

$$\begin{aligned} \min \quad & \sum_{i=1}^n \underbrace{\sum_{s \in \mathcal{S}_i} c(s)}_{c_i} a(i) \quad s.t. \\ a(i) & \geq r_i(u) + \lambda \sum_k p_k a(j_k), \quad \forall i, \forall u \end{aligned} \quad (7)$$

where $j_k \in \mathcal{T}(i, u, y_k)$ includes all the partitions one can transition to from meta-state i under the influence of action u with the random input taking the value y_k . Notice that the transition probability from meta-state i to meta-state j_k is p_k . Clearly the number of inequalities per partition depends on the size of the transition map $|\mathcal{T}(i, u, y_k)|$. The restricted LP deals with only n variables and not more than $|\mathcal{T}| \times n \times |\mathcal{U}|$ inequalities as compared to the original LP (3), which deals with $|\mathcal{S}|$ variables and $|\mathcal{S}| \times |\mathcal{U}|$ inequalities. If the partitioning were done in an efficient manner with $n \ll |\mathcal{S}|$,

then the restricted LP should indeed be tractable. In addition if the partitions were to be of the A-type defined earlier, the restricted LP would simplify even further to

$$\min \sum_{i=1}^n c_i a(i) \quad s.t. \quad (8)$$

$$a(i) \geq r_i(u) + \lambda a(j), \quad \forall j \in \mathcal{T}(i, u), \quad \forall i, \forall u$$

In this case the transition probability from meta-state i to meta-state j is one.

B. Bounds on the approximate solution

An approximate value function $V_a \in \mathbb{R}^{|\mathcal{S}|}$ can be constructed from every feasible solution (and in particular, the optimal solution) to the restricted LP (7) according to

$$V_a(s) = a(i), \quad \forall s \in \mathcal{S}_i, \quad i = 1, \dots, n$$

Since the approximate value function satisfies by construction the inequality

$$V_a(s) \geq r(s, u) + \lambda \sum_k p_k V_a(f(s, u, y_k))$$

it is an upper bound to the optimal value function. As far as lower bound is concerned, one can construct a sub-optimal *greedy* policy based on the approximate value function [17], [18]:

$$\pi_a(s) = \operatorname{argmax}_u \{r(s, u) + \lambda \sum_k p_k V_a(f(s, u, y_k))\}.$$

The determination of the lower bound is done as follows: For every s , one determines an improvement error,

$$\tilde{V}(s) := \{-V_a(s) + r(s, \pi_a(s)) + \lambda \sum_k p_k V_a(f(s, \pi_a(s), y_k))\}$$

If $V_{\min} = \min_{s \in \mathcal{S}} \tilde{V}(s)$, then one gets the bounds [4], [17], [18]

$$\frac{V_{\min}}{1 - \lambda} + V_a(s) \leq V_{\pi_a}(s) \leq V^*(s) \leq V_a(s), \quad \forall s \in \mathcal{S} \quad (9)$$

where $V_{\pi_a} \in \mathbb{R}^{|\mathcal{S}|}$ is the value function associated with the sub-optimal policy π_a :

$$V_{\pi_a} = (I - \lambda P_{\pi_a})^{-1} R_{\pi_a}$$

where with some abuse of notation, we have $R_{\pi_a}(i) = r(i, \pi_a(i))$ and $P_{\pi_a}(i, j)$ is the probability of going from state i to state j under the influence of action $\pi_a(i)$.

For the exact LP (3), the choice of weights c does not influence the solution. Interestingly this is not true for the restricted LP (6). In fact, the choice of state-relevance weights may bear a significant impact on the quality of the resulting approximation (c.f. Section 3 [9]). We have the result (c.f. Lemma 1 [9]): $\tilde{a} \in \mathbb{R}^n$ is the optimal solution to the restricted LP (6) iff it is the optimal solution to the following LP:

$$\min \|V^* - \Phi a\|_{1,c} \quad s.t. \quad (10)$$

$$(I - \lambda P_u) \Phi a \geq R_u \quad \forall u$$

So the goal is to find a weight vector a such that Φa is *close* to the optimal value function V^* . Now the cost function in (10) can be written as

$$\begin{aligned} \|V^* - \Phi a\|_{1,c} &= \sum_s c(s) |V^*(s) - V_a(s)| \\ &= \sum_i \frac{c_i}{|\mathcal{S}_i|} \sum_{s \in \mathcal{S}_i} |V^*(s) - a(i)| \\ &= \sum_i c_i \left(a(i) - \frac{1}{|\mathcal{S}_i|} \sum_{s \in \mathcal{S}_i} V^*(s) \right) \end{aligned}$$

where the last equality follows since $a(i) \geq V^*(s)$ for all $s \in \mathcal{S}_i$. So we see that the RLP is designed to minimize the distance between $a(i)$ and the average of the optimal value function in every partition i .

In addition to the bounds established earlier (9), we also have (c.f. Theorem 2 [9]): If $a \in \mathbb{R}^n$ is the optimal solution to the restricted LP (6) and c satisfies $c(s) > 0$, $\forall s$ and $\sum_{s \in \mathcal{S}} c(s) = 1$ and if $\mathbf{1}$ (the column vector with all ones) is in the span of the columns of Φ , then

$$\begin{aligned} \|V^* - \Phi a\|_{1,c} &\leq \frac{2}{1 - \lambda} \min_v \|V^* - \Phi v\|_{\infty} \\ &= \frac{1}{1 - \lambda} \max_i \Delta_i \end{aligned} \quad (11)$$

Since by definition, $\Phi \mathbf{1} = \mathbf{1}$, the above bound is valid. The last equality in (11) holds true if Δ_i is defined to be the difference between the smallest and largest value function in any given partition i i.e.,

$$\Delta_i = \max_{s \in \mathcal{S}_i} V^*(s) - \min_{s \in \mathcal{S}_i} V^*(s)$$

So the quality (fit) of the approximation (11) depends on the (worst case) variation in the optimal value function within a partition.

IV. PERIMETER SURVEILLANCE PROBLEM

We shall showcase the partitioning approach using the perimeter patrol problem that arose out of the Cooperative Operations in Urban Terrain (COUNTER) project at the Air Force Research Laboratory (AFRL) [19], [20]. In this problem, there is a closed perimeter which must be monitored by a collection of UAVs (we will consider only one UAV here). Along the perimeter are m Unattended Ground Stations (UGS) and for the sake of simplicity, incursions into the perimeter can only occur at the stations. The stations flag an alert when there is an incursion. An incursion can be a threat or a nuisance and must be serviced by the UAV. By servicing, we mean that a UAV physically goes to the alert site, dwells near the site (we refer to this time as the dwell time), takes video of the vicinity of the site and transmits it to an operator. The operator, upon receiving and examining the transmitted video, will make the call as to whether the alert has the semblance of a nuisance or a threat. For the transmitted video to be relevant, the UAVs must service an alert within a certain response time. Otherwise, there will be no information gained and perhaps, one must penalize such a tardy response. The reward for a UAV to service an alert

not only depends on the time an alert has not been serviced (we will refer to it as the delay, τ_i , if the alert is at the i^{th} station) but also on the dwell time. We will model the reward to be bounded. Furthermore, we will treat it to be a decreasing function of the delay and a concave function of the dwell time.

To describe the problem further, we discretize time and space. Let the boundary have $N \geq m$ nodes which are uniformly separated, of which m correspond to the alert stations. Let at any time instant k , $x(k)$ be the position of the UAV on the perimeter, $d(k)$ is the dwell time if it is at a station, $\tau_i(k)$ is the duration of the time an alert at i^{th} station has not been serviced. Let $A_i(k)$ be a binary variable indicating the status of the alert at the i^{th} station and $\tilde{Y}_i(k)$ be another binary, but random variable indicating the arrival of an alert at the i^{th} station. We will assume that the statistics associated with the random variable $\tilde{Y}_i(k)$ are known and that $\tilde{Y}_i, i = 1, \dots, m$ are independent. Strictly speaking, we model the arrival of alerts as follows: There is a single queue of alerts sampled from a (continuous time) Poisson process. After the alert is queued up, the location of the alert is chosen randomly (uniform distribution over all stations). For this reason, only one alert can arrive at the m stations at any instant of time. Hence, there are $m + 1$ possibilities for the value of the vector of alerts $y(k) = [\tilde{Y}_1(k) \dots \tilde{Y}_m(k)]$, with the first one being that there is no alert at any station and the other m correspond to an alert at each of the stations. The decisions allowed at the stations are indicated by the binary variable: u . If $u = 0$, the UAV moves to the next node; if $u = 1$, the UAV dwells at the current alert station. We will assume, without loss of generality, that a UAV moves by one unit every time step if $u = 0$. Also, we assume that the time to complete one loiter is also equal to the time step. One may then write the state equations for the system as follows:

$$\begin{aligned} x(k+1) &= (x(k) + 1 - u(k)) \bmod N \\ A_i(k+1) &= (1 - \delta(x(k) - X_i)u(k)) \max\{A_i(k), \tilde{Y}_i(k)\}, \\ &\quad i = 1 \dots m \\ d(k+1) &= (d(k) + 1)u(k) \\ \tau_i(k+1) &= \min \left\{ \begin{array}{l} (\tau_i(k) + 1)A_i(k) \\ T \end{array} \right\}, \quad i = 1 \dots m \end{aligned} \quad (12)$$

with T being a positive integer imposed to make the state space finite. So, any service delay bigger than T is assumed to be equal to T . The m station locations are indicated by X_1, \dots, X_m and δ is the Kronecker delta function. Also, we have the additional constraints

$$u(k) \leq \sum_{i=1}^m \delta(x(k) - X_i)$$

i.e., UAV can only loiter at alert stations and

$$\begin{aligned} d(k) &\leq D \\ d(k) = D &\Rightarrow u(k) = 0 \end{aligned} \quad (13)$$

This constraint imposes a maximum on the number of allowed dwell orbits. If $d(k) = D$, the UAV is forced to

leave the station. Also we note that, service delay $\tau_i(k) > 0$ only if there is an active alert at station i , i.e., $A_i(k) = 1$ or if the alert was just serviced, i.e., $d(k) = 1$. Otherwise it stays at zero. We may express the state equations compactly as

$$s(k+1) = f(s(k), u(k), y(k)),$$

where $s(k)$ shall represent the system state at time k . Let us (arbitrarily) order the states and with a slight abuse of notation, henceforth use s to also mean the s^{th} state in the set of all states $\mathcal{S} = \{1, 2, \dots, |\mathcal{S}|\}$. We will model the stage cost/reward r to be a function of the current dwell state d and service delay τ , i.e., $r(k) = r(d(k), \tau_i(k))$. In particular we shall use the definition

$$r(k) = \mathcal{I}(d(k))u(k) - \beta \max_i \tau_i(k) \quad (14)$$

where \mathcal{I} is the information gain function (see Fig. 1) based on an operator error model [20]. $\beta > 0$ is a constant weighing the incremental information gain upon loitering once more against the delay in servicing active alerts. Let the alert

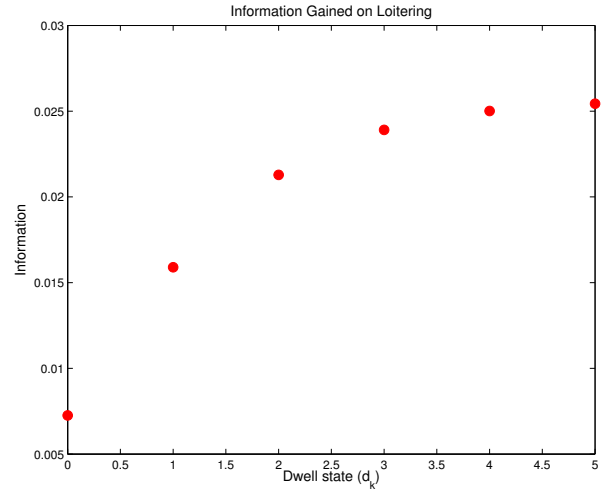


Fig. 1. Information Gained per additional Loiter

arrival rate be α and given that the alerts follow a poisson process, the probability that there is no alert ($\tilde{Y}_j = 0, \forall j$) is $p = e^{-\alpha}$ and hence, the probability that $y(k)$ takes any one of the $m + 1$ possible values is given by

$$p_j = \begin{cases} p & j = 1, \\ \frac{(1-p)}{m} & j = 2 \dots (m+1) \end{cases}$$

Now that we have set up the problem (using notation in section II), we can go ahead and write Bellman's equation to solve for the optimal value function $V^* \in \mathbb{R}^{|\mathcal{S}|}$:

$$V^*(s) = \max_u \left\{ r(s, u) + \lambda \sum_{j=1}^{m+1} p_j V^*(f(s, u, y_j)) \right\} \quad (15)$$

which not surprisingly is in the same form as for the generic problem discussed earlier (1). The number of states ζ can be

easily computed to be

$$\zeta = \sum_{k=0}^m \binom{m}{k} (T+1)^k (N + (m-k)(D+T)) \quad (16)$$

If we use the original LP (3) to solve for the optimal value function, we would end up with an exceedingly large number of variables and inequalities! Hence the need for approximate LP solutions. Towards this end, we shall first aggregate the states and then employ the restricted LP approach enumerated earlier.

A. Structure associated with the Patrol problem

We see that by definition (14) there is a structure to the reward, $r(s, u)$. To explain further, consider a station where an alert is being serviced by the UAV. The information gained by the UAV about the alert is only a function of the amount of time the UAV dwells at the station while processing the alert. There is a natural partitioning of the state-space i.e., no matter what the individual delays at the other stations are, the reward is the same as long as the maximum service delay and the dwell time at the station are the same. So we partition the state-space by aggregating all the states with the same location x , dwell d , alert status A and maximum delay ($\max_{j=1}^m \tau_j$). Then the reward associated with any partition i automatically satisfies: for all $s \in \mathcal{S}_i$,

$$r(s, u) = r_i(u) = \mathcal{I}(d_i)u - \beta t_i$$

where t_i is the maximum delay corresponding to partition i . Notice that this is a B-type partition as per our earlier definition (c.f. section III-A). The number of partitions (i.e. number of meta-states) can easily be computed to be

$$n = N + m(D+T) + \sum_{k=1}^m \binom{m}{k} (T+1)(N + (m-k)D)$$

which is linear in T as compared to the total number of states ζ , which is an m^{th} order polynomial in T (16). Henceforth we will use d_i and $A_{i,k}$, $k = 1 \dots m$ to denote the dwell and alert status corresponding to partition i .

B. Restricted LP for the Patrol Problem

Now we can write the restricted LP described earlier (7) for the patrol problem as,

$$\min \sum_{i=1}^n c_i a(i) \quad s.t. \quad (17)$$

$$a(i) \geq r_i(u) + \lambda \sum_{k=1}^{m+1} p_k a(j_k), \quad j_k \in \mathcal{T}(i, u, y_k), \quad \forall i, \forall u$$

Clearly we will transition from partition i to partition j_k with maximum delay

$$\begin{aligned} t_{j_k} &= 0, \quad A_{i,k} = 0, \forall k \\ t_{j_k} &= \min(t_i + 1, T), \quad d_i \neq 1, A_{i,k} = 1, k \in 1 \dots m \\ t_{j_k} &\in \{1, 2, \dots, \min(t_i + 1, T)\}, \quad \text{otherwise} \end{aligned} \quad (18)$$

Now the three possible scenarios above correspond respectively to states

- i) with no stations flagging an active alert,
- ii) with one or more active stations and dwell state not equal to one,
- iii) with dwell state one (loitering at some station q) and an alert at one or more of the other stations. Note that when $d = 1$ at a station q , the alert status for that station is zero i.e., $A_q = 0$ (it was just serviced and the alert state reset).

For scenarios (i) and (ii) above, all states in partition i will transition to the same future partition j_k when the random alert vector takes the value y_k . But for scenario (iii), for every instance of y_k , states in partition i can transition to any one of $\min(t_i + 1, T)$ partitions corresponding to $t_{j_k} = 1 \dots \min(t_i + 1, T)$. Which partition they transition to is determined by the highest service delay among all stations excluding the station q that the UAV is currently loitering at (i.e., $\max_{p=1 \dots m, p \neq q} \tau_p$). This means that for scenario (iii), $a(i)$ has to satisfy $\min(t_i + 1, T)$ inequality constraints for each $u \in \mathcal{U}$. The immediate question is whether all of these constraints are binding and consequently if one or more can be dropped from the LP. This is an important consideration given that the total number of constraints impacts on the computation cost. To answer this question, we first note that the inequality constraints for scenario (iii) can be written out (upon expanding the immediate future reward) as

$$\begin{aligned} a(i) &\geq r_i(u) - \lambda(\beta + \dots) \\ a(i) &\geq r_i(u) - \lambda(2\beta + \dots) \\ &\vdots \\ a(i) &\geq r_i(u) - \lambda(\min(t_i + 1, T)\beta + \dots) \end{aligned} \quad (19)$$

for all $u \in \mathcal{U}$ which appears to suggest that (since $\beta > 0$), satisfying just the first constraint i.e., the one corresponding to $t_{j_k} = 1$ ensures that all the constraints in (19) are met. This turns out to be true for the problem under consideration (corroborated by numerical evidence)! So we rewrite the restricted LP for the patrol problem (UBLP):

$$\begin{aligned} \min \sum_{i=1}^n c_i a(i) \quad s.t. \quad (20) \\ a(i) \geq r_i(u) + \lambda \sum_{k=1}^{m+1} p_k a(j_k), \quad \forall i, \quad \forall u. \end{aligned}$$

with the maximum delay corresponding to partition j_k ,

$$t_{j_k} = \begin{cases} 0, & A_{i,k} = 0, \forall k \\ \min(t_i + 1, T), & d_i \neq 1, A_{i,k} = 1, k \in 1 \dots m \\ 1, & \text{otherwise} \end{cases}$$

The UBLP above deals with n variables and exactly $n \times |\mathcal{U}|$ constraints.

Interestingly if we do the exact opposite and retain only the last constraint corresponding to $t_{j_k} = \min(t_i + 1, T)$ and

define a new LP (LBLP):

$$\begin{aligned} \min \sum_{i=1}^n c_i a(i) \quad s.t. \\ a(i) \geq r_i(u) + \lambda \sum_{k=1}^{m+1} p_k a(j_k), \quad \forall i, \quad \forall u. \end{aligned} \quad (21)$$

with the maximum delay corresponding to partition j_k ,

$$t_{j_k} = \begin{cases} 0, & A_{i,k} = 0, \forall k \\ \min(t_i + 1, T), & otherwise \end{cases}$$

Again the LBLP above deals with n variables and exactly $n \times |\mathcal{U}|$ constraints. As was done with the restricted (upper bound) LP, an approximate value function $V_b \in \mathbb{R}^{|\mathcal{S}|}$ can be constructed from the optimal solution to (21) according to

$$V_b(s) = a(i), \quad \forall s \in \mathcal{S}_i, \quad i = 1, \dots, n$$

Not surprisingly, it turns out that $V_b \leq V^*$! Similar to what was done with the upper bound solution, one can construct a sub-optimal policy based on the lower bound approximate value function:

$$\pi_b(s) = \underset{u}{\operatorname{argmax}} \{r(s, u) + \lambda \sum_{k=1}^{m+1} p_k V_b(f(s, u, y_k))\}$$

V. NUMERICAL RESULTS

For demonstration purposes, we assume a perimeter with 15 nodes of which $\{1, 4, 8, 12\}$ are stations and a maximum allowed dwell of 5 orbits. The other parameters were chosen to be weighing factor, $\beta = 0.002$ and discount factor, $\lambda = 0.9$. The alert rate α was chosen to be one every 30 sampling time instants (one every two patrols around the perimeter). For a nominal choice of $T = 15$, we already have an exceedingly large number of states, $\zeta = 1,645,855$. Upon aggregation, we end up with a manageable $n = 5935$ variables for the restricted LPs. The cost vector was chosen according to $c_i = 1, \forall i$, giving equal weight to all partitions.

We used **MATLAB linprog** [21] installed on a AMD Athlon (2.2 GHz, 2 GB RAM) Dual Core processor based computer to run the algorithms. The original LP wouldn't even run on the machine ("Out of memory" error). The upper bound (20) and lower bound (21) LP formulations were run and the results are compiled in table I. We compare the number of variables ($\#var$), the number of constraints ($\#con$), computation time and the % error in policy. For the patrol problem parameters considered herein, we could still use a *value iteration* approach to solve for the optimal value function and policy. This was done by exploiting the sparsity of the transition probability matrices involved. For larger values of m and T , this approach would breakdown and/or the computational burden would become prohibitively high. Using the optimal policy, we could compute the % error in the upper bound based sub-optimal policy (π_a) to be 32.3% and the lower bound based sub-optimal policy (π_b) to be 10.2%. Since there are one too many states to plot, we give a representative sample of the results by showing all the partitions corresponding to alert status $A = 1111$ (all

stations active) and maximum delay 2. Fig. 2 shows the the upper bound (V_a) and lower bound (V_b) approximation along side the optimal value function. In the plot, each partition is separated by the dotted vertical lines. Fig. 3 shows the error in approximation for the upper bound based sub-optimal policy:

$$e_a(s) = V^*(s) - V_{\pi_a}(s), \quad \forall s$$

and the lower bound based sub-optimal policy:

$$e_b(s) = V^*(s) - V_{\pi_b}(s), \quad \forall s$$

where V_{π_a} and V_{π_b} are again the value functions associated with the suboptimal policies π_a and π_b respectively. We see that $e_b < e_a$ in most states as a direct consequence of the the sub-optimal policy π_b being *closer* to the optimal.

TABLE I
COMPARISON OF DIFFERENT LP FORMULATIONS

LP	#var	#con	time (min)	Error
Opt	1,645,855	2,333,675	-	-
UBLP	5,935	8,315	1.7	32.3%
LBLP	5,935	8,315	1.7	10.2%

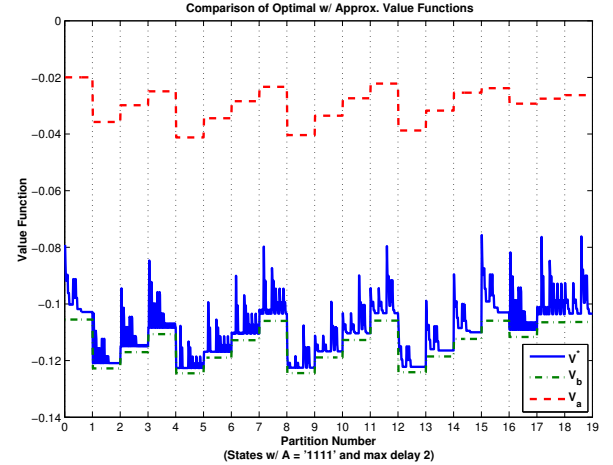


Fig. 2. Comparison of UB and LB approximate value functions with the Optimal

VI. CONCLUSIONS

A state partitioning based LP approach has been proposed to solve large scale stochastic control problems. This approach is convenient for controlled Markov chains where the transition probabilities between the partitioned meta-states are known. The approach is attractive in that it also provides bounds for deviation of the derived sub-optimal policies from the optimal. Finally, the approach has been demonstrated on a perimeter surveillance stochastic control problem. The original LP is computationally expensive for the patrol problem and the restricted LP approach provides sub-optimal solution in quick time. More importantly exact DP methods turn out to be intractable for other large scale problems frequently encountered in practise. Assuming the

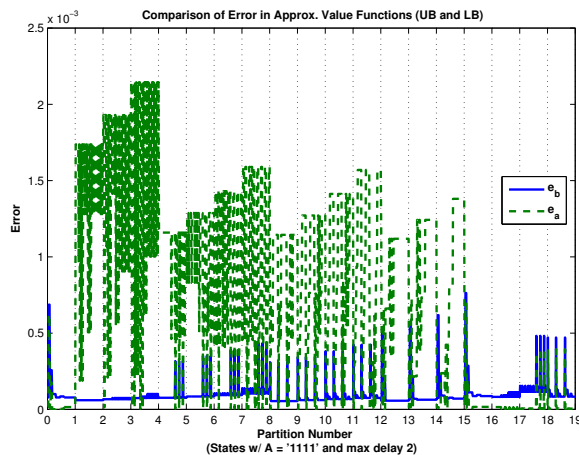


Fig. 3. Error in UB and LB sub-optimal policy based Value Functions

reward structure is amenable to aggregation, the proposed restricted LP approach should be a viable alternative for solving these problems.

REFERENCES

- [1] A. S. Manne, "Linear programming and sequential decisions," *Management Science*, vol. 6, pp. 259–267, 1960.
- [2] F. d' Epenoux, "A probabilistic production and inventory problem," *Management Science*, vol. 10, pp. 98–109, 1963.
- [3] A. Hordijk and L. C. M. Kallenberg, "Linear programming and markov decision chains," *Management Science*, vol. 25, no. 4, pp. 352–362, 1979.
- [4] P. J. Schweitzer and A. Seidmann, "Generalized polynomial approximations in markovian decision processes," *J. of Mathematical Analysis and Applications*, vol. 110, pp. 568–582, 1985.
- [5] M. Trick and S. Zin, "Spline approximation to value functions: A linear programming approach," *Macroeconomic Dynamics*, vol. 1, pp. 255–277, 1977.
- [6] J. R. Morrison and P. R. Kumar, "New linear program performance bounds for queueing networks," *J. of Optimization Theory and Applications*, vol. 100, no. 3, pp. 575–597, 1999.
- [7] W.-R. Heilman, "Solving stochastic dynamic programming problems by linear programming - an annotated bibliography," *Mathematical Methods of Operations Research*, vol. 22, no. 1, pp. 43–53, 1978.
- [8] G. Gordon, "Approximate solutions to markov decision processes," Ph.D. dissertation, Carnegie Mellon University, Pittsburg, PA, 1999.
- [9] D. P. De Farias and B. Van Roy, "The linear programming approach to approximate dynamic programming," *Operations Research*, vol. 51, no. 6, pp. 850–865, 2003.
- [10] B. Van Roy, "Performance loss bounds for approximate value iteration with state aggregation," *Mathematics of Operations Research*, vol. 31, no. 2, pp. 234–244, May 2006.
- [11] S. Axsäter, "State aggregation in dynamic programming: An application to scheduling of independent jobs on parallel processors," *Operations Research Letters*, vol. 2, pp. 171–176, 1983.
- [12] J. Bean, J. Birge, and R. Smith, "Aggregation in dynamic programming," *Operations Research*, vol. 35, pp. 215–220, 1987.
- [13] R. Mendelssohn, "An iterative aggregation procedure for markov decision processes," *Operations Research*, vol. 30, pp. 62–73, 1982.
- [14] D. Bertsekas and D. Castanon, "Adaptive aggregation for infinite horizon dynamic programming," *IEEE Trans. Automat. Control*, vol. 34, no. 6, pp. 589–598, 1989.
- [15] R. E. Bellman, *Dynamic Programming*. Princeton, NJ: Princeton University Press, 1957.
- [16] R. Howard, *Dynamic Programming and Markov Processes*. Cambridge, MA: MIT Press, 1960.
- [17] E. L. Porteus, "Bounds and transformations for discounted finite markov decision chains," *Operations Research*, vol. 33, pp. 761–784, 1975.
- [18] J. MacQueen, "A modified dynamic programming method for markovian decision problems," *J. of Mathematical Analysis and Applications*, vol. 14, pp. 38–43, 1966.
- [19] D. Gross, S. Rasmussen, P. Chandler, and G. Feitshans, "Cooperative operations in urban terrain (counter)," in *Defense and Security Symposium*. Orlando, FL: SPIE, Apr. 2006.
- [20] P. Chandler, J. Hansen, R. Holsapple, S. Darbha, and M. Pachter, "Optimal perimeter patrol alert servicing with poisson arrival rate," in *AIAA Guidance, Navigation and Control Conf.*, Chicago, IL, August 2009.
- [21] Y. Zhang, "Solving large-scale linear programs by interior-point methods under the matlab environment," Department of Mathematics and Statistics, University of Maryland, Baltimore, MD, Tech. Rep. TR96-01, 1995.