Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence

Online Optimization with Dynamic Temporal Uncertainty: Incorporating Short Term Predictions for Renewable Integration in Intelligent Energy Systems

Vikas K. Garg
Toyota Technological Institute
at Chicago (TTI-C)
vkg@ttic.edu, montsgarg@gmail.com

T. S. Jayram
IBM Almaden
jayram@almaden.ibm.com

Balakrishnan Narayanaswamy
IBM Research, India
Murali.Balakrishnan@in.ibm.com

Abstract

Growing costs, environmental awareness and government directives have set the stage for an increase in the fraction of electricity supplied using intermittent renewable sources such as solar and wind energy. To compensate for the increased variability in supply and demand, we need algorithms for online energy resource allocation under temporal uncertainty of future consumption and availability. Recent advances in prediction algorithms offer hope that a reduction in future uncertainty, through short term predictions, will increase the worth of the renewables. Predictive information is then revealed incrementally in an online manner, leading to what we call *dynamic* temporal uncertainty. We demonstrate the non-triviality of this problem and provide online algorithms, both randomized and deterministic, to handle time varying uncertainty in future rewards for non-stationary MDPs in general and for energy resource allocation in particular. We derive theoretical upper and lower bounds that hold even for a finite horizon, and establish that, in the deterministic case, discounting future rewards can be used as a strategy to maximize the total (undiscounted) reward. We also corroborate the efficacy of our methodology using wind and demand traces.

Growing costs, environmental awareness and government directives have set the stage for an increase in the fraction of electricity supplied using renewable sources (Wiser and Barbose 2008). Distributed generation (Ackermann, Andersson, and Soder 2001), using renewable sources, is gaining prominence and perceived as vital in achieving cost and carbon reduction goals. Extracting value from a time varying and intermittent renewable energy resource requires intelligent optimization of markets (Bitar et al. 2011), generation (Bhuvaneswari et al. 2009), storage (Zhu et al. 2011) and loads (Kowli and Meyn 2011), and motivates this work.

Advanced techniques such as Weather Research and Forecasting (WRF) models (Monteiro et al. 2009), signal processing (Abdel-Karim, Small, and Ilic 2009), and machine learning (Sharma et al. 2011) can offer accurate predictions of solar and wind power availability (Contreras et al. 2003). With short term predictions, information about future time steps is refined as we get closer to them. This kind of *dynamic* uncertainty - where uncertainty changes as more information is revealed after every action - is widely prevalent. We demonstrate how online algorithms can be designed for such optimization and control problems (with dynamic uncertainty) that lie at the intersection of Model Predictive Control (MPC) (incorporate predictions) and Markov Decision Processes (MDPs) (model the state and action space). We also provide strong performance guarantees (both upper and lower bounds) for our algorithms.

While we focus on energy applications, our techniques are generic. We model a class of online resource allocation problems in terms of MDPs *with short term predictions* and develop non-obvious algorithms for them. In the process, we give a theoretical justification of MPC algorithms that have been widely used in the control theory. In particular, we

- study a number of online optimization problems that arise in smart grids in an MDP framework with arbitrary (non-stochastic, possibly adversarial) rewards,
- propose practicable algorithms for such problems and analyze the underlying MDP in a novel setting where short term predictions of future rewards (e.g., future renewable availability, demand and prices) are available,
- show how an algorithm that maximizes a time discounted form of the rewards (with appropriate discounting factor) can perform arbitrarily better than the obvious algorithm that maximizes the total (undiscounted) reward *even* when exact (i.e. noiseless) short term predictions are available[1],
- and provide simulations to substantiate that discounting can improve performance with short term predictions.

Prediction based (micro) storage management

Uncertainty in renewable energy supply and the growing imbalance between energy supply and demand have motived much recent work in algorithms for the online allocation of scarce intermittent energy supply to time varying demand at minimum cost. These problems arise for example in (i)

[1] In other words, we show that discounting can be used as a strategy to maximize the total (undiscounted) reward.

(micro) grid management (Katiraei et al. 2008), where local generation has to be scheduled and allocated optimally to local demand, (ii) buying and selling intermittent wind power to maximize profits (Bitar et al. 2011), (iii) demand response, where demand is to be curtailed or shifted to maximize utility (Kowli and Meyn 2011) and (iv) storage management, which involved optimal charging and discharging in response to time varying prices, demand and supply (Urgaonkar et al. 2011). While we focus on energy applications here, we emphasize that our results are applicable to general MDPs for online resource allocation *with short term predictions*. Our motivation is the ongoing project to establish a research microgrid at the Kuala Belalong Center in Brunei[2].

The online optimization problem that is of particular interest to us, and which subsumes many other smart grid resource allocation problems, is the problem of optimal energy storage management given short term predications of demand, prices and renewable power availability. In every time slot t suppose we, as a consumer, obtain some utility $A_t(d_t)$ for consuming electricity d_t. We also have access to a time varying amount of (free) renewable energy z_t and can purchase an additional amount of electricity p_t at a cost of c_t/unit on the electricity spot market. We choose to draw f_t from battery of size B_{max}, (note that f_t can be negative corresponding to charging the battery), so that battery state update is $x_{t+1} = x_t - f_t$ to satisfy a demand $d_t = z_t + f_t + p_t$. So the *offline* problem to be solved is

$$\underset{p_1,\ldots,p_T,f_1,\ldots,f_T}{\text{maximize}} \quad \sum_{t=1}^{T} A_t(d_t) - p_t c_t = \sum_{t=1}^{T} r(x_t, u_t, w_t) \tag{1}$$

$$\text{subject to} \quad \left.\begin{array}{r} d_t = z_t + f_t + p_t,\ 0 \le x_{t+1} \le B_{max} \\ -B_{ch} \le f_t \le B_{ch},\ x_{t+1} = x_t + f_t \end{array}\right\} \forall\, t$$

where B_{ch} is the maximum amount the battery can be charged or discharged by in a single time slot. The function (1) is called the *welfare* function and is the difference of the utility gained by the user by consuming electricity and the price paid for purchasing it. In general, utility (and reward) functions are assumed to be positive, concave and increasing satisfying a diminishing returns property (Li, Chen, and Low 2011) for computational reasons. Our techniques and model are more general and our results hold for any choice of utility function. We are interested in online algorithms for this problem where rewards and costs are revealed and refined *dynamically* through short term predictions.

Markov Decision Process Model

The smart grid applications we listed above, and many others online resource allocation problems, share some common characteristics allowing us to model the problem using an MDP. They have a notion of the state of the system $x_t \in \mathcal{X}$ in a time slot t and a set of actions (or decisions) available to the algorithm in that state $u_t \in \mathcal{U}$. In general the set of available actions could depend arbitrarily on the state. An action moves the system into a new state x_{t+1} in the next time step and depending on the state of nature $w_t \in \mathcal{W}$, gives the decision maker a reward $r(x_t, u_t, w_t)$. We assume all sets are finite. In the storage management problem $u_t = [p_t\ f_t]$ denotes the action taken and $w_t = [c_t\ A_t]$ denotes the disturbance while (1) gives the time varying rewards. We will use a **deterministic** MDP model where the state transition σ is defined as a mapping $(x, u) \mapsto x'$ where $x, x' \in \mathcal{X}$, $u \in \mathcal{U}$ and $w \in \mathcal{W}$. Our techniques can also be extended stochastic MDPs which we discuss towards the end of this paper. Rewards can be non-stationary but we restrict them to be non-negative; let $r_t(x, u, w)$ denote the reward at time t. We normalize the rewards to lie in the interval $[0, 1]$

In this paper, we are interested in online algorithms that use short term predictions, i.e. those that make decisions on actions u_t given knowledge of the next few states of nature which correspond to information about the disturbances $w_t, w_{t+1} \ldots$ and as a result the reward functions $r(\bullet, \bullet, w_t), r(\bullet, \bullet, w_{t+1}), \ldots$. In our energy applications for example, these values are often available given short term predictions of demand $d_t, \ldots, d_{t+L}$, prices $c_t, \ldots, c_{t+L}$ and customer utilities $A_t, \ldots, A_{t+L}$ for L steps, as discussed in the previous section. The optimal solution to this MDP for a horizon T is given by solving the following optimization problem $\underset{u_1,\ldots,u_T}{\text{maximize}} \sum_{t=1}^{T} r(x_t, u_t, w_t)$. Such an MDP can be used, for example, to represent exactly problem (1).

Dynamic temporal uncertainty for online algorithms with short term predictions

We assume that the disturbance information available to the algorithm for some future time step t, while it proceeds through steps $1, 2, \ldots, t-1$, is that it belongs to a *fixed* set $\mathcal{W}_t$; during step t, a fixed element $w_t \in W_{t,t}$ is revealed, but only after the action u_t is taken. We study *online* algorithms where the disturbances are *dynamically* evolving. At the start of time step t, nature reveals to the online algorithm a sequence of *active* disturbance sets $\mathcal{W}_{t,t}, \mathcal{W}_{t,t+1}, \ldots, \mathcal{W}_{t,T} \subseteq \mathcal{W}$, called the **disturbance profile** for time t. The set $W_{t,\hat{t}}$ represents *the allowed disturbances for time step $\hat{t}$ as seen by the algorithm at time t.* After an action is taken for this step, nature chooses a disturbance $w_t \in W_{t,t}$ called the **realized disturbance** for step t. We assume that the disturbance profiles satisfy a consistency property, in that the realized disturbance w_t is in all of them. As a result we can always take intersections with previous disturbance profiles, to have $\mathcal{W}_{1,t} \supseteq \mathcal{W}_{2,t} \supseteq \cdots \supseteq \mathcal{W}_{t,t}$, for all t. To summarize, the sequence of events, when the online algorithm is in state x_t at the start of time step t, is as follows: (i) the active disturbances $W_{t,\hat{t}}$ for all $t \le \hat{t} \le T$ are selected arbitrarily by an adversary and revealed to the online algorithm, (ii) the online algorithm takes some action u_t and moves to state $\sigma(x_t, u_t)$ and obtains a reward $r(x_t, u_t, w_t)$, (iii) time is incremented $t \leftarrow t + 1$. Thus, we consider the fully adversarial setting where future rewards can be chosen arbitrarily.

We adopt the customary approach of comparing the total reward obtained by the online algorithm ON to an *offline* algorithm with total reward OFF that has prescient

[2]http://ubdestate.blogspot.in/2009/06/kuala-belalong-field-studies-centre.html

information regarding the disturbance profile. A **strong offline** algorithm, inspired by (Regan and Boutilier 2011), is one that knows the realized disturbances w_t for every t beforehand. A **weak offline** algorithm, on the other hand, is similar to adaptive robust optimization (Ben-Tal, El Ghaoui, and Nemirovski 2009; Iancu 2010) in that it knows only the disturbance set $W_{t,t}$ for every t beforehand and has no knowledge of the realized disturbance. The **regret** of the online algorithm equals the maximum value of $OFF - ON$, whereas the **competitive ratio** equals the maximum value of OFF/ON over all choices of the disturbance parameters. In this paper, we provide bounds on the **average regret** defined as $\frac{OFF-ON}{T}$ over a time horizon T. Finally, we define the **diameter** or **delay** of an MDP as a fixed constant (for a given problem) ρ which is the maximum number of steps required for an algorithm to move from any specific state to another, similar to (Ortner 2010). For example, for the storage management algorithm $\rho = \frac{B_{max}}{B_{ch}}$.

Non-triviality of online optimization with short term predictions

We first briefly describe a simple example that indicates the **non-triviality** of regret minimization and achieving reasonable competitive ratio even with *perfect, noiseless* predictions of the future. We consider a situation where perfect forecasts of the rewards are available for the next $L+1$ time steps. Note that this corresponds to singleton sets $W_{t,\hat{t}}$ for $\hat{t} \leq t + L + 1$. A **natural algorithm**, given information at time t about the next $L + 1$ steps, would compute the best path that collects the most reward over the revealed horizon. Given this calculated path it then executes the first action. After the execution it is revealed information about one more step in the future, namely $t + L + 2$ at time $t + 1$. It then re-calculates the best path given this information and current state. This process is repeated in each time step. Note that this algorithm is completely natural and is essentially Model Predictive Control from control theory (Morari and Lee 1999), which is very widely used in practice.

The Game: Consider a state space with 4 states in sequence : $A - B - C - D$. Transitions are only allowed to stay in the same state e.g. $A \to A$, or to transition between neighbouring states i.e. $A \to B, B \to C, C \to D$ and in reverse. Suppose, the reward for transitioning into A in an even time step t is $(G + t\epsilon)$ and 0 in odd time steps, while the reward for transitioning into D in an odd time step t is $(G+t\epsilon)$ and 0 in even time steps, though this is not revealed to the online algorithm. Choosing $\epsilon = \frac{1}{T}$, say, ensures that rewards are bounded. In alternate time steps it is most profitable to transition to A and D, with no rewards for transitioning to the intermediate states B and C. Suppose that initially, at $t = 0$, the algorithm starts in state B. We consider the case where the current and next $L = 4$ states of nature are revealed before an action is taken.

Path followed by the natural algorithm: From B we are given a prediction that transitioning into A for the current and next 4 steps gives a reward $[0\ , G + \epsilon\ , 0\ , G + 3\epsilon\ , 0]$ while transitioning into D gives $[G\ , 0\ , G + 2\epsilon\ , 0, G + 4\epsilon]$. The best path available to the algorithm is $B \to C \to D \to D \to D$ gathering a reward of $2G + 6\epsilon$. It then executes the transition $B \to C$. Now given predictions for transition into A of $[G + \epsilon\ , 0\ , G + 3\epsilon\ , 0, G + 5\epsilon]$ and transition into D of $[0\ , G + 2\epsilon\ , 0, G + 4\epsilon, 0]$ the best path available to the algorithm is $C \to B \to A \to A \to A$ gathering a reward of $2G+8\epsilon$. We see that for any even lookahead L the algorithm repeats $B \to C \to B$ and collects no reward.

Competitive ratio of the natural algorithm: The optimal algorithm for the example above is for the algorithm to remain in state A or D throughout (depending on if T is odd or even) and collect a reward of approximately $\frac{T}{2}(G + \frac{T}{2}\epsilon)$. This results in a competitive ratio of ∞ *even for L large*.

While such counter examples are used in the analysis of greedy online algorithms, we have not seen them in the context of MPC. In the sequel we show how a simple modification of the natural algorithm can give us deterministic algorithms with better performance. We will develop a regularized (or discounted) algorithm below that 'believes' that 'a bird in hand is worth 2 in the bush', and will not fall into the same trap as the natural algorithm. *A primary contribution of our work is the analysis technique, which uses ideas from the potential function analysis of algorithms in a novel manner, that formalizes this intuition forming an novel bridge between online algorithms and MPC and MDP problems.*

Quantifying dynamic uncertainty

For ease of exposition, we will assume that the rewards are set to zero for time steps beyond the time horizon T, so that $r(\cdot)$ is well-defined for all positive integers t. We also set $\mathcal{W}_{t,\hat{t}}$ equal to some dummy set (say $\mathcal{W}$) for all $\hat{t} > T$. We consider a graph representation corresponding to the time unrolled (non-stationary) MDP (Madani 2002). We treat the state-action pair $(x_{\hat{t}}, u_{\hat{t}})$ as an *edge* $e_{\hat{t}}$ in the graph representation of the underlying MDP that can be taken during time step $\hat{t}$. Given the disturbance profile at time t and a future time step $\hat{t} \geq t$, each state-action pair $(x_{\hat{t}}, u_{\hat{t}})$ induces a set of rewards $I_{t,\hat{t}}(x_{\hat{t}}, u_{\hat{t}}) = I_{t,\hat{t}}(e_{\hat{t}}) = \{r(x_{\hat{t}}, u_{\hat{t}}, w_{\hat{t}}) : w_{\hat{t}} \in \mathcal{W}_{t,\hat{t}}\}$, each of which involves the common transition to state $\sigma(x_{\hat{t}}, u_{\hat{t}})$. The assumption that the transitions in the MDP are not affected by disturbances is quite restrictive and this may be relaxed by choosing a weaker offline algorithm to compare against, which we do not address here. Technically $I_{t,\hat{t}}(e_{\hat{t}})$ is a set but for our purposes, we may think of it as an interval whose *length* is $|I_{t,\hat{t}}(e_{\hat{t}})| = \max_{w_{\hat{t}}} r(x_{\hat{t}}, u_{\hat{t}}, w_{\hat{t}}) - \min_{w_{\hat{t}}} r(x_{\hat{t}}, u_{\hat{t}}, w_{\hat{t}})$. We define the **uncertainty function** $g(k) = \max_{t,\hat{t}:\hat{t}-t\leq k} \max_{x_{\hat{t}},u_{\hat{t}}} |I_{t,\hat{t}}(x_{\hat{t}}, u_{\hat{t}})|$. This is a non-decreasing function that encapsulates how large the uncertainty can get as we look further into the future. The key quantity in our bounds is the **effective uncertainty** $\eta_{g,\rho}(\gamma)$ of a ρ-delayed MDP with uncertainty function $g(\cdot)$, defined as a function of the discount factor $0 \leq \gamma < 1$.

$$\eta_{g,\rho}(\gamma) = g(0) + \frac{1 + \sum_{k=1}^{\infty} \gamma^k h(k)}{1 + \sum_{k=1}^{\infty} \gamma^k} \tag{2}$$

Here $h(k)$ equals 1, when $k < \rho$, and $g(k)$ otherwise. The first term $g(0)$ accounts for the uncertainty the online algorithm has about its current reward and is, essentially, unavoidable. The second term is much more interesting and

Algorithm 1 (Online Algorithm for Dynamic Uncertainty)

Setup: An MDP with initial state x_1, state transition function $\sigma(\cdot)$, and rewards $r_t(\cdot)$ for all t.
Input at time t: State x_t, reward intervals $I_{t,\hat{t}}(e_{\hat{t}})$ for all $\hat{t} \geq t$ and edges $e_{\hat{t}}$. Let $r_{t,\hat{t}}(e_{\hat{t}})$ denote the minimum value in $I_{t,\hat{t}}(e_{\hat{t}})$.
Output at time t: Action u_t.

Let $\hat{x}_t = x_t$ also denote the current state.
for each current action $\hat{u}_t$ and future sequence of actions $(\hat{u}_{t+1}, \hat{u}_{t+2}, \dots)$ **do**
  Let $\hat{e}_\tau$ denote the edge $(\hat{x}_\tau, \hat{u}_\tau)$ such that $\hat{x}_{\tau+1} = \sigma(\hat{x}_\tau, \hat{u}_\tau)$ for $\tau = t, t+1, \dots$
  Let $\hat{P}_t$ denote the path followed starting from $\hat{x}_{t+1}$ by taking this future sequence of actions
  Compute $\Delta_t(\hat{P}_t) = \sum_{k=1}^{\infty} \gamma^k \cdot r_{t,t+k}(\hat{e}_{t+k})$
  Compute the **anticipated** reward $r_{t,t}(\hat{e}_t) + \Delta_t(\hat{P}_t)$
end for
Compute (e_t, P_t) such that it maximizes the anticipated reward. Let $e_t = (x_t, u_t)$.
Take the action u_t to move to state x_{t+1}, and obtain the realized reward $\bar{r}_t(e_t) \in I_{t,t}(e_t)$

measures the *usable* information about future rewards available to the online algorithm relative to the worst case scenario where no such information is available. For example, information about the next ρ steps may be of limited value (due to unreachability), and decisions regarding future uncertainties, albeit important, may be deferred to a later time, and so should be discounted appropriately. Note that even with perfect short term predictions, the effective uncertainty is non-zero and the optimal discounting factor is some $\gamma < 1$, as emphasized in Theorem 2 below.

Discounting as a strategy for online optimization

We now describe an online algorithm, based on the notion of discounting future rewards, to obtain good regret bounds. The pseudocode for the algorithm is in Algorithm 1. The algorithm uses the minimum value of $I_{t,\hat{t}}(e_{\hat{t}})$ denoted by $r_{t,\hat{t}}(e_{\hat{t}})$ as the **estimated reward**; plausible alternatives do exist, but the analysis becomes more involved. We set the rewards to equal 0 beyond time step T. With this convention, we can state the regret bound for this algorithm.

Theorem 1. *For a ρ-delayed MDP with uncertainty function $g(\cdot)$ with bounded rewards, Algorithm 1 with discount factor γ achieves an average regret at most $\eta_{g,\rho}(\gamma)$ when compared to the strong offline algorithm.*

Proof. For an edge e_t and time t, let $\bar{r}_t(e_t)$ denote the **realized reward** given by the realized disturbance w_t. Consider the execution of the offline algorithm as tracing an infinite path $(a_1, a_2, \dots,)$ of edges starting from x_1. Let A_t denote the suffix $(a_{t+1}, a_{t+2}, \dots)$ of this path for any t. The total reward collected by the offline algorithm over the horizon T is $OFF = \sum_{t=1}^{T} \bar{r}_t(a_t)$. At time t, the online algorithm computes e_t and P_t such that the **anticipated reward** $r_{t,t}(e_t) + \Delta_t(P_t)$ is maximized. Thus, the anticipated reward of the chosen path will be better than for the path that joins the offline algorithm's path by traversing some edge e_t^* starting from x_t (since $\rho = 1$) and then following the path A_t. The the anticipated reward of this path equals $r_t(e_t^*) + \Delta_t(A_t)$, where $\Delta_t(A_t) = \sum_{k=1}^{\infty} \gamma^k r_{t,t+k}(a_{t+k})$.

$$r_{t,t}(e_t) + \Delta_t(P_t) \geq r_t(e_t^*) + \Delta_t(A_t) \geq \Delta_t(A_t). \quad (3)$$

In addition, the anticipated reward of the chosen path will be better than for the path P_{t-1} that was selected to maximize the anticipated reward during time step $t-1$. (For $t = 1$ set this reward equal to zero.) Indeed, this path starts from x_t and if $P_{t-1} = (e'_t, e'_{t+1}, \dots)$, then

$$r_t(e_t) + \Delta_t(P_t) \geq \sum_{k=0}^{\infty} \gamma^k r_{t,t+k}(e'_{t+k}) \geq \sum_{k=0}^{\infty} \gamma^k r_{t-1,t+k}(e'_{t+k}) = \tfrac{1}{\gamma} \cdot \Delta_{t-1}(P_{t-1}), \quad (4)$$

where the second inequality follows term-wise since $I_{t,t+k}(e_{t+k}) \subseteq I_{t-1,t+k}(e_{t+k})$. After multiplying (3) by $1-\gamma$, (4) by γ, and add the two inequalities for $t = 1, 2, \dots, T$. The sum telescopes, and since $\Delta_T(P_T)$ involves zero reward edges beyond the horizon T, we obtain:

$$\begin{aligned} \sum_{t=1}^{T} r_{t,t}(e_t) &\geq (1-\gamma) \sum_{t=1}^{T} \Delta_t(A_t) \quad (5) \\ &= (1-\gamma) \sum_{t=1}^{T} \sum_{k=1}^{\infty} \gamma^k r_{t,t+k}(a_{t+k}). \end{aligned}$$

For the edge e_t traversed by the online algorithm we have: $|r_{t,t}(e_t) - \bar{r}_t(e_t)| \leq g(0)$. For the edge a_{t+k} traversed by the offline algorithm at time $t+k$, we have: $|r_{t,t+k}(a_{t+k}) - \bar{r}_{t+k}(a_{t+k})| \leq g(k)$. Using these facts in the above equation, the total reward collected by the online algorithm,

$$\begin{aligned} ON &:= \sum_{t=1}^{T} \bar{r}_t(e_t) \geq -Tg(0) - (1-\gamma)T \sum_{k=1}^{\infty} \gamma^k g(k) \\ &+ \underbrace{(1-\gamma) \sum_{t=1}^{T} \sum_{k=1}^{\infty} \gamma^k \bar{r}_{t+k}(a_{t+k})}_{(A)}. \quad (6) \end{aligned}$$

We now evaluate expression (A) above, replacing $s = t+k$,:

$$\begin{aligned} (A) &= (1-\gamma) \sum_{s=1}^{T} \bar{r}_s(a_s) \sum_{k=1}^{s-1} \gamma^k = \sum_{s=1}^{T} \bar{r}_s(a_s)(\gamma - \gamma^s) \\ &= \gamma \cdot OFF - \sum_{s=1}^{T} \bar{r}_s(a_s)\gamma^s. \quad (7) \end{aligned}$$

Applying this bound in (6) and rearranging terms, we obtain:

$$\frac{OFF - ON}{T} \leq g(0) + (1-\gamma)\frac{OFF}{T} + (1-\gamma) \sum_{t=1}^{\infty} \gamma^k g(k). \quad (8)$$

Here, the second term in (7) is a geometric series that becomes negligible compared to T, for large enough T (When

padding we assumed that the rewards are 0 for the first q steps. Then $\gamma^q \ll 1/T$, say for $q = \log T$). Since, $OFF/T \leq 1$, the average regret is:

$$\frac{OFF - ON}{T} \leq g(0) + (1-\gamma)(1 + \sum_{t=1}^{\infty} \gamma^k g(k)) = \eta_{g,\rho}.$$

□

If ρ is unbounded then Theorem 1 is trivial and it is then possible to construct adversarial sequences for which the average regret of *any* online algorithm is unbounded.

Since the generality of the above proof may obscure the message, we consider a special case that is often of practical interest, when *perfect* predictions of the rewards are available for the current and next L time steps.

Theorem 2. *Consider the case of a ρ delayed MDP with **fixed lookahead** where the rewards have no uncertainty for $L+1$ steps, for some lookahead $L \geq \rho$, and no information is available beyond $L+1$ steps in the future i.e. $g(k) = 0$, for $k \leq L$, and 1 otherwise. For $\gamma = 1 - \frac{\log L}{L}$, the average regret is $\Theta(\frac{\log L}{L-\rho+1})$ and the competitive ratio is $1+\Theta(\frac{\log L}{L-\rho+1})$ compared to the strong offline algorithm.*

Here we see that as $L \to \infty$, the regret goes to 0, unlike the natural undiscounted MPC algorithm. In the next section we show that permitting ourselves some amount of randomization will allow us to prove better bounds. This serves two purposes: it gives a flavor of different algorithms and analyses that are possible, and it highlights the intuition that the deterministic discounting algorithm is in some sense a *derandomization* against possible futures.

Competitive ratio against the weak offline

In this section we present a *randomized* and *robust* online adaptive algorithm to handle dynamic uncertainty. This randomized algorithm is also applicable to the problem studied in the previous section, where it gives a better bound for *expected* reward. For simplicity, we assume that $\rho = 1$, though the proofs generalize. We consider the weak offline algorithm that is given a set of disturbances $\mathcal{W}_t^*, \forall\, t$ before $t = 1$, while the online algorithm is revealed the same disturbance set L steps in advance so that $\mathcal{W}_{t,t} = \mathcal{W}_{t-1,t} = \mathcal{W}_{t-2,t} = \ldots = \mathcal{W}_{t-L,t} = \mathcal{W}_t^*$ and $\mathcal{W}_{t-L-s,t} = \mathcal{W}\ \forall\ s \geq 1$. Note that, as before, an adversary can choose the sequence of disturbance sets arbitrarily. We denote the sequence of actions taken by the online algorithm with $u_1, \ldots, u_T$. Let the offline algorithm OFF we compare against have a policy consisting of an action u_t^* for every time step t and state x_t.

We design a 'lazy' online algorithm that chooses the path that optimizes the reward once every $L+1$ steps and follows this path for the next $L+1$ steps with no updates, reminiscent of the episodes of (Ortner 2010). Our online algorithm has a choice of $L+1$ possible starting times or 're-set' points $j = 1, \ldots, L+1$. In the $\rho = 1$ case, this corresponds to possibly 'giving up' rewards $t_0 = (L+1) + j$ in order to transition to the state that the optimal offline algorithm would be in at $t_0 + 1$. The algorithm makes this choice of 're-set' point j uniformly at random from 1 to $L+1$ without revealing this choice to the adversary, thus making it a randomized algorithm. At a particular time t_0 our online algorithm solves the following robust optimization problem for L time steps

$$\max_{u_{t_0}} \min_{w_{t_0} \in \mathcal{W}_{t_0,t_0}} \ldots \max_{u_{t_0+L}} \min_{w_{t_0+L} \in \mathcal{W}_{t_0,t_0+L}} \sum_{\tau=t_0}^{t_0+L} r(x_\tau, u_\tau, w_\tau)$$
$$\text{subject to :}\quad x_{\tau+1} = \sigma_t(x_\tau, u_\tau),\ \ \tau \in \{t_0, \ldots, t_0+L\}$$

The algorithm executes this sequence of $u_{t_0}, \ldots, u_{t_0+L}$. It then repeats this operation for the next batches at time $t+L, t+2L, \ldots$. Note that one policy evaluated by the online algorithm is to jump to x_{t+1}^* in step t and follow the same sequence of policies as the optimal offline algorithm for the remaining steps. As a consequence, for the alternative sequence of policies chosen by OFF, the algorithm can produce a 'witness' sequence of $\hat{w}_t, \ldots, \hat{w}_{t+L}$ such that

$$\sum_{\tau=t_0}^{t_0+L} r(x_\tau, u_\tau, w_\tau) \geq \sum_{\tau=t_0+1}^{t_0+L} r(x_\tau^*, u_\tau^*, \hat{w}_\tau) \qquad (9)$$

Essentially, the algorithm is giving up at most one reward in every sequence of $L+1$ rewards that the offline algorithm gets. Summing up (9) over $t_0, t_0+L, \ldots$ we have that

$$\begin{aligned} \sum_{t=1}^{T} r(x_t, u_t, w_t) \quad &\geq \quad \sum_{t=1}^{T} r(x_t^*, u_t^*, \hat{w}_t) \qquad (10) \\ &- \sum_{t=i*(L+1)+j} r(x_t^*, u_t^*, w^*) \end{aligned}$$

The first term in (10) is a lower bound on the rewards collected by the offline algorithm, for the w_t^*s (since the w_t^*s were chosen as the minimizers of the optimal offline problem). If the online algorithm uniformly chooses a starting point j in $1, \ldots, L+1$, then the second term in (10), on average, is bounded by $\frac{1}{L+1} OPT$, where OPT is the reward of the optimal offline algorithm, giving us the following:

Theorem 3. *For a ρ-delayed MDP with bounded rewards the randomized algorithm, in expectation over internal randomness, achieves a competitive ratio $1 + \Theta(\frac{1}{L+1})$ and regret $\Theta(\frac{1}{L+1})$ when compared to the weak offline algorithm.*

Simulations

Experimental set-up: We consider a home with local loads, storage and wind as in (1). Realistic load traces y_t were generated using the model built in (Richardson et al. 2010) from 1 year of data from 22 UK households. We collected wind speeds near our micro grid location in Brunei. For spot market prices, we used traces from the New England ISO. We assumed the demand must be satisfied, so that (1) becomes a cost minimization problem. For our experiments, we state results with costs as opposed to the (equivalent) reward model discussed earlier. Since the total cost has a free parameter (namely, the absolute amount of energy consumed), we scaled all the costs in the figures. We also set the storage size to $(1/3)^{rd}$ of the average daily consumption, and assumed it can be changed fully in 5 steps ($\rho = 5$).

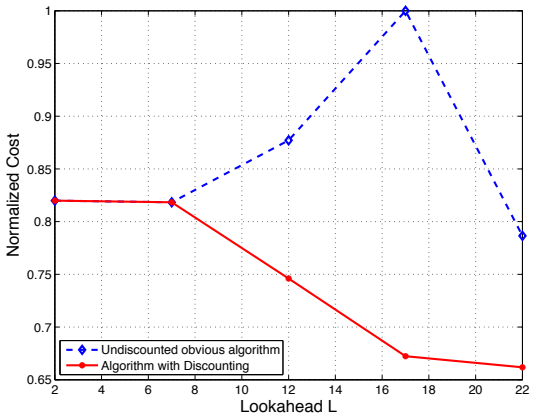

Figure 1: Benefits of discounting for storage management

The need for discounting: We now present the results of our experiments to further support our assertion that discounting is useful in real problems. In Figure 1, we compare the performance of the natural undiscounted algorithm with the discounting based strategy (for different settings of the lookahead) for the first 100 days of the year, assuming error free predictions of demand, prices and renewable supply. We see that, for the natural algorithm, increasing the lookahead into the future can actually *increase* the cost. We observed this phenomenon of reduced performance with increased lookahead was common to most of our experiments (including, sometimes, the discounting strategy). Thus, we emphasize that there is a scope for extracting value from predictions of the future even when they are *precise*.

Discussions and Conclusions

Near tightness of our bounds follows from constructing counterexample sequences. For instance, consider a reward matrix with the rows representing states and the columns time steps. Let $\rho = 1$, i.e. one reward is lost in any transition after the first time instant. The adversary generates the matrix with $\lceil \frac{1}{\epsilon} \rceil$ rows and $L+2$ columns (which can be repeated indefinitely). The first column contains all zeroes, the next L columns contain all ones, and one row in the last column contains a one while the rest contain zeroes. The optimal choice is to move during the first step to the row containing the one in column $L+2$, and to stay in that row for $L+1$ steps, collecting a total reward of $L+1$. Any (randomized) online strategy stands only an ϵ chance of choosing this row, and with probability at least $1-\epsilon$ must either miss one unit reward. The following result can be proved rigorously.

Theorem 4. *The competitive ratio of any online algorithm for arbitrary MDPs with perfect predictions of rewards for the next L time steps is larger than $1+\frac{1}{L}$. The competitive ratio of any deterministic online algorithm is larger than $1+\frac{1}{L}+\frac{1}{1+L^2}$.*

These results, along with Theorem 3 and Corollary 2, show that if randomization is possible and average case guarantees are sufficient, our randomized algorithm is essentially optimal in expectation with *expected* competitive ratio $1+\Theta(1/L)$. For deterministic algorithms, we obtain a lower bound of $1+\Omega(1/L+(1/L^2))$ and an upper bound of $1+O(\frac{\log L}{L})$. These results indicate a non-trivial separation between what is possible with randomized and deterministic algorithms; in particular, it may be possible to design better deterministic algorithms than what we presented here.

Computational considerations: Depending on the structure of the reward functions, the online algorithm has a complexity polynomial in L at each time step. For instance, if convex, the optimization problem being solved in Algorithm 1 resembles robust optimization over horizon L. (Ben-Tal, El Ghaoui, and Nemirovski 2009) discuss conditions that make the optimization problem computationally efficient.

MDPs with stochastic rewards and transitions: Our algorithm can be easily extended to account for stochasticity in transitions and in rewards. An equivalent notion of ρ-delay, for non-stationary MDPs corresponds to reaching an arbitrary distribution over states from any initial distribution, within ρ time steps. In particular, we can prove

Theorem 5. *If the online algorithm is revealed the distribution over transitions and rewards with lookahead L, then a version of Algorithm 1 has average regret $\Theta(\frac{\log L}{L-\rho+1})$ and expected competitive ratio $1+\Theta(\frac{\log L}{L-\rho+1})$ when compared with a weak offline algorithm that also only has access to the same distributions, albeit all before time $t=1$.*

Prior work: Uncertainty is known to exist in many real world applications (Nilim and Ghaoui 2005). Robust planning under uncertainty has been addressed by many papers (Jaksch, Ortner, and Auer 2010; Bartlett and Tewari 2009; Mannor, Mebel, and Xu 2012). Robust optimization techniques for MDPs have been proposed recently, e.g., (Regan and Boutilier 2011), but lack theoretical support. Some recent work has aimed at understanding optimization involving *parameter* uncertainty (under the aegis of stochastic optimization and robust optimization (Ben-Tal, El Ghaoui, and Nemirovski 2009)); however, in this paper, we explored the problems raised by *temporal* uncertainty, where knowledge of the future is limited. Another line of work strives to quantify the relative performance compared to a) the best alternative policy in the hindsight (Yu and Mannor 2009), and b) the best stationary policy over time (Even-Dar, Kakade, and Mansour 2009). Robust optimization of *Imprecise-reward MDPs* (IRMDPs) (Regan and Boutilier 2011), wherein an initially large feasible (uncertain) reward set shrinks as more information is acquired through online elicitation, has also been proposed. For deterministic MDPs, (Ortner 2010) showed ϵ-optimal performance on the finite horizon regret; however, the underlying assumption that successive rewards, for any fixed edge, are generated i.i.d. from a fixed but unknown probability distribution may not hold in practice. Contrastingly, we consider the case where external, possibly uncertain, *dynamic* predictions of the rewards are available for arbitrary, non-stationary MDPs.

In Theorem 2 and Theorem 5, the $L = T$ case corresponds to offline optimization or conventional robust optimization. **To the best of our knowledge, the setting $L < T$**

has been studied for the first time in our work. So, our work significantly builds on the robust and online optimization literature by accounting for the time varying/dynamic prediction structure in a non-stochastic world. Our approach is also reminiscent of MPC algorithms used in control theory (Morari and Lee 1999), and our analysis can be seen as providing some theoretical justification for such algorithms that discount rewards for tractability. However, unlike MPC, we demonstrate how discounting can be leveraged, as a strategy, to do well with respect to the total (undiscounted) reward.

References

Abdel-Karim, N.; Small, M.; and Ilic, M. 2009. Short term wind speed prediction by finite and infinite impulse response filters: A state space model representation using discrete markov process. In *IEEE PowerTech*, 1 –8.

Ackermann, T.; Andersson, G.; and Soder, L. 2001. Distributed generation: a definition. *Electric Power Systems Research* 57(3):195 – 204.

Bartlett, P., and Tewari, A. 2009. REGAL: A regularization based algorithm for reinforcement learning in weakly communicating MDPs. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, 35–42. AUAI Press.

Ben-Tal, A.; El Ghaoui, L.; and Nemirovski, A. 2009. *Robust optimization*. Princeton Univ Press.

Bhuvaneswari, R.; Edrington, C. S.; Cartes, D. A.; and Subramanian, S. 2009. Online economic environmental optimization of a microgrid using an improved fast evolutionary programming technique. In *North American Power Symposium (NAPS), 2009*, 1 –6.

Bitar, E.; Rajagopal, R.; Khargonekar, P.; Poolla, K.; and Varaiya, P. 2011. Bringing wind energy to market. *Submitted to IEEE Transactions on Power Systems*.

Contreras, J.; Espinola, R.; Nogales, F.; and Conejo, A. 2003. ARIMA models to predict next-day electricity prices. *IEEE Transactions on Power Systems* 18(3):1014–1020.

Even-Dar, E.; Kakade, S. M.; and Mansour, Y. 2009. Online Markov Decision Processes. *Math. Oper. Res.* 34(3):726–736.

Iancu, D. 2010. *Adaptive robust optimization with applications in inventory and revenue management*. Ph.D. Dissertation, Massachusetts Institute of Technology.

Jaksch, T.; Ortner, R.; and Auer, P. 2010. Near-optimal regret bounds for reinforcement learning. *The Journal of Machine Learning Research* 11:1563–1600.

Katiraei, F.; Iravani, R.; Hatziargyriou, N.; and Dimeas, A. 2008. Microgrids management. *IEEE Power and Energy Magazine* 6(3):54 –65.

Kowli, A., and Meyn, S. 2011. Supporting wind generation deployment with demand response. In *IEEE Power and Energy Society General Meeting*, 1 –8.

Li, N.; Chen, L.; and Low, S. 2011. Optimal demand response based on utility maximization in power networks. In *IEEE Power and Energy Society General Meeting*, 1 –8.

Madani, O. 2002. On policy iteration as a newton's method and polynomial policy iteration algorithms. In *Proceedings of the National Conference on Artificial Intelligence*, 273–278.

Mannor, S.; Mebel, O.; and Xu, H. 2012. Lightning does not strike twice: Robust MDPs with coupled uncertainty. *ICML*.

Monteiro, C.; Bessa, R.; Miranda, V.; Botterud, A.; Wang, J.; and Conzelmann, G. 2009. Wind power forecasting: State-of-the-art 2009. Technical report, Argonne National Laboratory, Decision and Information Sciences Division.

Morari, M., and Lee, J. 1999. Model predictive control: Past, present and future. *Comput. Chem. Eng.* 23(4):667 –682.

Nilim, A., and Ghaoui, L. 2005. Robust control of markov decision processes with uncertain transition matrices. *Operations Research* 780–798.

Ortner, R. 2010. Online regret bounds for markov decision processes with deterministic transitions. *Theoretical Computer Science* 411(29):2684–2695.

Regan, K., and Boutilier, C. 2011. Robust online optimization of reward-uncertain MDPs. In *Proceedings of IJCAI-11*, volume 141.

Richardson, I.; Thomson, M.; Infield, D.; and Clifford, C. 2010. Domestic electricity use: A high-resolution energy demand model. *Energy and Buildings* 42(10):1878–1887.

Sharma, N.; Sharma, P.; Irwin, D.; and Shenoy, P. 2011. Predicting solar generation from weather forecasts using machine learning. In *IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 528 –533.

Urgaonkar, R.; Urgaonkar, B.; Neely, M.; and Sivasubramaniam, A. 2011. Optimal power cost management using stored energy in data centers. In *Proceedings of the ACM SIGMETRICS*, 221–232. ACM.

Wiser, R., and Barbose, G. 2008. Renewables portfolio standards in the united states; a status report with data through 2007. Technical Report LBNL-154E, Lawrence Berkeley National Laboratory.

Yu, J., and Mannor, S. 2009. Online learning in markov decision processes with arbitrarily changing rewards and transitions. In *GameNets' 09*, 314–322. IEEE.

Zhu, T.; Mishra, A.; Irwin, D.; Sharma, N.; Shenoy, P.; and Towsley, D. 2011. The case for efficient renewable energy management for smart homes. *Proceedings of the Third Workshop on Embedded Sensing Systems for Energy-efficiency in Buildings (BuildSys)*.