

Multiscale Adaptive Agent-Based Management of Storage-Enabled Photovoltaic Facilities

Rongxin Li, Peter Wang¹ and Geoff James²

Abstract. We propose a multiscale algorithm for autonomous agents to adaptively manage the operation of storage-enabled photovoltaic (PV) facilities based upon sequential decisions in a partially observable environment. The stochastic environment is learned and modeled by an approach called an ϵ -Machine, which operates on a set of *a priori* determined temporal scales, to give the agent an additional degree of freedom when optimizing its control decisions. We compare the performance of the proposed scheme with those of (1) control decisions based on a heuristic environment model rather than systematic learning, and (2) decisions made on a pre-determined scale. We argue that the systematic environment learning on multiple temporal scales makes the agent highly adaptable and, as a result, able to demonstrate a superior capability in managing the PV-storage facility according to a pre-determined objective. The particular application focused upon in the paper is in managing distributed PV facilities as potential replacement of conventional peaking power plants.

1 Introduction

1.1 Background

Photovoltaic (PV) generation of electricity constitutes an important renewable energy source. However, higher cost, intermittency and volatility have been identified as the main obstacles to more widespread uptake of solar and wind power generation. Conversely, it is expected that improving the profitability and reliability will result in greater penetration of renewable generation.

Electricity demand and price often vary greatly during each day. One of the disadvantages with renewable generation is that, in the absence of intervention, the output peaks are generally not aligned with either electricity demand peaks or wholesale price spikes. Improving this alignment not only helps with the economic aspects of renewable generation, but can also accelerate the displacement of conventional peaking power plants, which are often gas fired and emit large quantities of greenhouse gases.

The above improvement is made possible by energy storage, which adds a time-shifting capability to the production of renewable energy, allowing the shaping and planning of the latter. By taking into account the battery and weather models, intelligent storage planning can potentially cost-effectively shift production towards optimal alignment.

In this paper we examine the possibility of using mathematical modeling and operation optimization to tackle those issues. This research continues from our previous work reported in [3].

The generation and storage components of the proof-of-concept application addressed in this paper are shown in Figure 1. The application scenario involves using solar power for either trickle injection into the electricity grid or local non-critical use (e.g. hot water tanks, pool pumps) most of the time. The system presented in this paper is designed to have the maximum potential to deliver at demand or price peaks that occur at unpredictable times. The storage is aimed to be maintained at near the battery's maximum capacity without exceeding it, in preparation for unexpected opportunities.

At the same time, it is also highly desirable to minimize wasting energy resources, as increased utilization of available resources speeds up the financial return on renewable investment. In the presence of continuous strong irradiance the power potentially in excess of the sum of the greatest local consumption level at the time and the maximum battery and network absorption capacities (the latter being constrained by the network infrastructure) has to be dumped (e.g. to heat sinks). The probability of occurrence of such a situation, therefore, should be minimized.

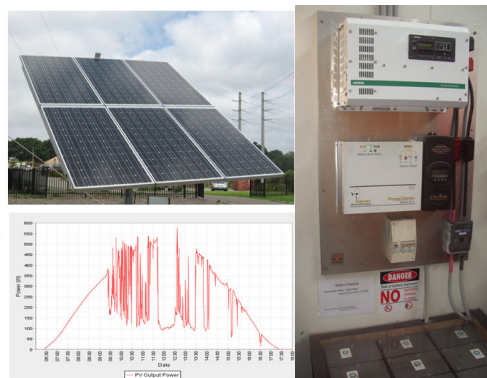


Figure 1. The generation and storage components of the proof-of-concept application addressed in this paper. Top left: PV modules that power a renewable peaking plant, with output over a typical fine day displayed on the bottom left; Right: a bank of batteries for scheduling and stabilizing the plant's output and an inverter for injecting the PV-generated power to the electricity grid.

The above goals require optimal planning by the autonomous agent within a stochastic environment. A decision-theoretic approach, Markov Decision Processes (MDPs), has therefore been chosen for this task. An MDP mathematically models planning problems where outcomes are partly under the control of the agent despite some uncontrollable randomness [8, 4]. In some applications, the autonomous agent is unable to acquire full and precise knowledge of the true state of the system. In such situations, an extended

¹ Information Engineering Laboratory, CSIRO ICT Centre, Australia.

² CSIRO Energy Technology, Australia.

MDP model referred to as the Partially Observable Markov Decision Process (POMDP) is more applicable, where decisions are made on the basis of a probabilistic distribution over the states of the system – known as a belief state, which is inferred by an autonomous agent from its unreliable or partial observations. The policy for an MDP or a POMDP prescribes the best action to be taken for any given belief state.

A limitation of the MDP or POMDP is that they can only operate on a system with a fully specified state space, including the states and the transition probabilities between them. This requirement often needs to be met by the use of an additional learning method. Our solution incorporates a learning approach called an ϵ -Machine. Different from the conventional learning methods, an ϵ -Machine captures a stochastic process with the least number of hidden Markovian states that can adequately describe the observed patterns, in the sense that the behavior exhibited in the empirical data can be statistically reproduced with that set of hidden states. These states are called the underlying process's causal states [12]. One of the main focuses of this paper is the integration of a causal-state-capturing ϵ -Machine with an operation optimizing POMDP.

2 Mathematical Formulations

2.1 ϵ -Machine

2.1.1 Original Formulation

An ϵ -Machine is a minimal stochastic sequence generator based on pattern-learning given an observed sequence. This sequence may be derived from historical data logs, as is the case with our application (the process is described in Section 4).

Formally, suppose that a discrete-time, discrete-valued Markovian process is represented by the bi-infinite sequence $\vec{S} = \dots S_{t-1} S_t S_{t+1} \dots$, where S_t is a random variable sampled at time t . $\vec{S}$ is split into a past series $\vec{S} = \dots S_{t-2} S_{t-1}$ and a future sequence $\vec{S} = S_t S_{t+1} \dots$. The conditional probability distribution of a future sequence is $P(\vec{S} | \vec{S} = \vec{s})$, where $\vec{s}$ is a past series of observations. The ϵ -Machine for $\vec{S}$ is defined as an ordered pair $\{\epsilon, \mathbf{T}\}$, where ϵ is the following function that maps from histories to sets of histories [11]:

$$\epsilon(\vec{s}) \equiv \left\{ \vec{s}' \mid P(\vec{S} \in F | \vec{S} = \vec{s}) = P(\vec{S} \in F | \vec{S} = \vec{s}'), \right. \\ \left. \forall F \in \mu(\vec{S}), \vec{s}' \in \vec{S} \right\}, \quad (1)$$

where F is a class of measurable future events, μ is the collection of all such classes, and $\mathbf{T}$ is the set of the transition matrices for the causal states of $\vec{S}$. As described in Section 1, the causal states are the minimal set of hidden Markovian states that can statistically reproduce the observations. $\epsilon(\cdot)$ defines sets of equivalent historical patterns. In practice, instead of bi-infinite sequences, a maximum length L_{max} is usually defined to restrict the lengths of $\vec{S}$ and $\vec{S}'$. More details of building and applying an ϵ -Machine are provided in Section 4. An important concept is the recurrent state. A recurrent state is a causal state, where for each transition to another causal state, there exists a series of transitions that will result in returning to the original state. An ϵ -Machine can essentially be described as a deterministic finite state machine that consists of only recurrent states and the associated transition probabilities. The recurrence constraint helps to ensure that only significant patterns in the observed sequence are learned by the ϵ -Machine.

2.1.2 Extension to Longer Scale Emitting Patterns

Given any significant value l , the number of states is be overwhelming as all possible time sequences up to that length need to be accounted for. Fortunately, the ϵ -Machine is able to select, via the above-mentioned test of recurrence, a relatively small of significant states.

In our current multi-scale setting, transition probabilities $P(\vec{c}_i^l | \vec{c}_j^l)$ for patterns of various temporal lengths $l \in L$ are computed, where $\vec{c}_i^l \in C^l$ and $\vec{c}_j^l \in C^l$ are a statistically significant pattern occurring in the past and in the future, respectively.

Due to the memory requirements for computing $P(\vec{c}_i^l | \vec{c}_j^l)$, an approximation method is used. $P(\vec{c}_i^l | \vec{c}_j^l)$ is instead computed by rolling forward the most likely unit state (a time series can be viewed as amalgamated unit states) for the length l . The computational detail is presented below. The impact of this approximation (which achieves a near-optimality) on the decision quality is currently being assessed.

2.2 Sequential Decisions on Multiple Scales

As mentioned above, the POMDP formulation is extended from MDP. An MDP is a stochastic control process defined by a tuple (S, A, T, R) , where S is the finite set of environment states; A is the set of possible actions that may be taken by the decision maker, $T : S \times A \times S \rightarrow [0, 1]$ is the probabilistic state-transition model, and $R : S \times A \rightarrow \mathbb{R}$ is the immediate reward or cost expected to be received after a transition. An underlying assumption is that the process is stationary and Markovian. The goal is to maximize or minimize an expected cumulative measure, usually a discounted sum of the immediate rewards or cost. The policy $\pi(s)$ for an MDP may be computed via algorithms such as value iteration or policy iteration, which prescribes the best action to be taken in a given state $s \in S$. With an additional finite set of observations O , a POMDP is defined as the tuple (S, A, T, R, O) . Upon observing $o \in O$ after taking action $a' \in A$, which is known to have the probability $p(o | s, a')$ if the environment is in state $s \in S$, the agent should update its belief about the current likelihood of s .

For our stated goals it is crucial to plan ahead using the right length of historical PV production data. As the Markovian order of the PV output as a stochastic process is unknown *a priori*, a multiscale strategy is employed to discover the optimal history length. The main environmental process, the PV panel's output, is assumed to be Markovian but possibly of a relatively high order. Based on historical data the ϵ -Machine learns all possible causal states up to a predetermined maximum length l_m . Although the environment process is assumed to be largely stationary, changes may happen across different seasons. Thus, the transition probabilities may need to be regularly updated.

Corresponding to the modeling scale l , each decision epoch is equivalent to the length l when mapped to the original scale of the time series. The state space for the POMDP is denoted as $S^l = S_s^l \times S_b^l$, where S_s^l is the set of the discretized PV outputs, and S_b^l is the vector of battery states. Note that as the battery process is a first order Markov chain (as discussed below), the computation of state transitions can be significantly simplified. To achieve this, we have used the simplified state space $\hat{S}^l = S_s^l \times S_b^l[l]$. At scale l the transition matrix is denoted as T^l . The solution to the POMDP $(\hat{S}^l, A, T^l, R, O)$ is denoted as $\pi^l(b, o_s^l)$, with o_s^l being the solar ob-

servation and b the belief about the battery's charge status (discussed below in more detail). Further denoting the corresponding value as $V\{\pi^l(b, o_s^l)\}$, we define the multiscale policy as

$$\pi = \underset{l}{\operatorname{argmax}} e^l V\{\pi^l(b, o_s^l)\}. \quad (2)$$

Solving POMDP in general for the infinite horizon is computationally intractable [6, 5]. To maintain computational tractability, a maximum likelihood heuristic described in [10] is adopted here. The approximation that results from this has been shown to be sufficiently accurate as the uncertainty regarding the battery's state is due to bounded observation noise, rather than unknown state features [10]. In this method, policies for the underlying discrete-state MDP are computed first.

Since the PV states are fully observable, the belief updating may be decomposed into two components, jointly described as (with the normalizing constant omitted):

$$b_{t+1}(s) \propto \delta(s_s - f_p(o_s)) p(o_b | s_b, a') \sum_{s'_b \in S_b^l[l]} \sum_{s'_s \in S_s} p(s_b | s'_b, s'_s, a') b_t(s'). \quad (3)$$

where $\delta(\cdot)$ is the Kronecker delta function, $s \in \hat{S}^l$, $s_b \in S_b^l[l]$, $s_s \in S_s^l$, $f_p(o_s)$ is the result of discretizing the PV observation o_s (as described in Section 2.1), o_b is the battery charge estimated from the noisy voltage observation. To be computationally feasible, a is restricted to a single type of action throughout the decision epoch. The policy to be adopted is:

$$\tilde{\pi}(b) = \pi(\arg \max_s \{b(s)\}). \quad (4)$$

3 System Description

3.1 Overview

On the basis of analyzing weather patterns at different temporal scales and modeling a given battery, stochastically optimizing operation decisions regarding the storage and release of energy are made sequentially to effectively shift the production to optimally compensate for sunlight availability under the battery constraint.

The system that we have developed is illustrated in Figure 2. This system can be parameter-tuned to address one or more of the following issues associated with renewable generation:

- Peak misalignment between generation and consumption, resulting in ineffective demand peak shaving and an inability to leverage wholesale pricing structure;
- Poor supply quality due to intermittency and volatility;
- Unsatisfactory running economics arising from the fact that wholesale electricity is mostly cheap commodity;
- Some battery operations are more expensive than the others.

3.2 Observations and Control Commands

3.2.1 Battery Observations

The capacity fading effect and the capacity recovery effect are two important aspects in battery operation [2]. The former is captured by Peukert's Law, which describes the battery capacity as a function of the discharge rate. These two effects are largely attributable to a common underlying mechanism: in an electrochemical cell, electroactive species becomes depleted near the electrodes as the battery discharges; however, during an idle period, the internal diffusion of

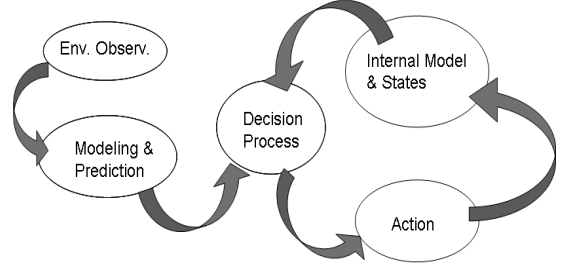


Figure 2. System Overview

the electroactive species becomes the dominant process in the absence of the discharge current [1]. The diffusion results in what is called the capacity recovery effect (CRE). It is a stochastic process, with the ensemble probability having an inverse relationship with the extent of the charge depletion [2, 1].

Together with the capacity fading effect, the existence of the CRE determines that the battery process is stochastic. Consequently, the battery's state transitions are modeled as a Markov chain. The battery's State-of-Charge (SoC) is indexed to constitute the states in the Markov chain. The transition probabilities take into account the capacity fading effect and stochastic recovery effect.

3.2.2 Uncertainty in Battery Observations

As described in Section 2.2, the state space for the POMDP is a product of two vectors. On the one that corresponds to the battery's SoC, the underlying state is only partially observable with significant uncertainty. This is because the battery's SoC is indirectly derived from voltage measurements, and there is a range of possible charge states that corresponds to each given voltmeter reading, the exact state being affected by other factors such as the temperature. However, this uncertainty is bounded within a range given as part of the battery's specifications. This means that the partial observability of the battery can be modeled as degradation by bounded additive noise. It is important to note that this partial observability does not arise from unknown state features.

3.2.3 PV Output Observations

On the other hand, the output of the solar module is fully observable. A practical aspect to note here is that, in order to maximize the generation potential of a PV module, its output ports are usually connected to a maximum power point tracker (MPPT), which controls the operation condition (namely, the voltage) for the PV. The output current of the MPPT can be accurately measured. The continuous readings are discretized into non-uniform bins in such a way that the number of data points falling into each bin is as constant as possible. This discretization method is often referred to as adaptive quantization.

3.2.4 Control Actions

The set of possible actions is known *a priori* to the control agent. To maintain computational tractability they are constrained to be the a single type of action throughout the decision epoch. Those actions correspond to distributing varying proportions of electricity to the battery for storage, to the electricity grid as trickle injection or to local appliances for immediate use. Any electricity exceeding the combined maximum battery intake and the maximum capability of the transmission infrastructure is automatically discarded. The action chosen by the controller agent is combined with the external factors to effect a system state transition, as illustrated in Figure 2.

4 Learning and Adaptive Control

In the construction phase of the ϵ -Machine, a series of observations must be made available. Using historical data logs, instantaneous readings at irregular intervals of the output of a PV panel are integrated and re-sampled at regular time intervals. This converts the current (in Amperes) to charge amounts (in Coulombs). These are deemed to be amounts available for grid injection, immediate local consumption or battery charging. For the system modeling, an algorithm called Causal State Splitting Reconstruction (CSSR) has been proposed to approximate the ϵ -Machine [12]. We employ a slight variation of the CSSR, which can be summarized in the four steps listed below, where length-restricted versions of $\vec{S}$ and $\vec{S}$ are denoted as $\vec{S}^L$ and $\vec{S}^L$, and a past observation is represented as $\vec{s}^L$, with L being the length:

1. Initialize the ϵ -Machine;
2. Homogenize the ϵ -Machine as defined by Eq. 1;
3. Determinize the ϵ -Machine;
4. Remove transient states from the ϵ -Machine.

The details of the above steps are briefly outlined as follows. In Step 1, through frequency counting, we compute an estimated probability distribution $\hat{P}^{(0)}(\vec{S}^1 | \vec{s}^0)$, where $\vec{s}^0$ represents the null sequence (thus the seemingly conditional probabilities are in fact independent probabilities). Note that the notion of the "null sequence" is used here merely for keeping the notation consistent with the other steps.

The subsequent step (Step 2) is multiple iterations with the main component directly following from Eq. 1. As L is increased by 1 in each iteration, a new causal state is created if a conditional probability $\hat{P}^{(L)}(\vec{S}^1 | \vec{s}^L)$ differs significantly (above a significance level α upon a hypothesis test) from all the other conditional probabilities given the existing causal states. Otherwise $\vec{s}^L$ is absorbed into the existing causal state that gives rise to the most similar conditional probability. This main component, along with its associated calculations, is repeated until the maximum history size L_{max} is reached. This is the process of gradually expanding the dependencies from marginal probabilities to conditional probabilities dependent on the previous L_{max} states.

In the next step (Step 3), each causal state is iteratively split into multiple states for each non-deterministic outgoing transition until all outgoing transitions are deterministic. A transition is deterministic if a transition out of one unit state (as explained above, a time series can be viewed as amalgamated unit states) leads to only one causal state. This is repeated for all causal states. The transition table is computed here. We now have a deterministic machine. The aim of Step 4 is to delete transient states. For each causal state, we iteratively check each outgoing transition for any paths back to the source causal state. If there is none, the causal state is a transient state and is deleted. This is again repeated for all causal states. These steps eventually produce an approximation to the ϵ -Machine.

4.1 Battery Modeling

The battery's operation is modeled as a discrete-time process. During each interval of operation, the charge loss from the discharging operation is $c_d = mQ_d$, where Q_d is a constant base amount, and $m \in [0, 1, \dots, M_d]$ depends on the discharge current. The recovery effect can be modeled as a Markov chain, as graphically illustrated in

[2], [1], [9] and [7]. A fully charged cell is assumed to have a maximum capacity of N charge units, where a charge unit is dependent upon both the battery discharge current and the time resolution. E represents the charge level when the battery reaches the discharged state. During idle intervals, the battery recovers a charge c_r with a probability $P_r(c)$, where c is its current charge level. Following [2], we have set $c_r = \frac{Q_d}{2}$. We model the probability as

$$P_r(c) = \begin{cases} 0 & c = N \text{ or } c = E \\ \frac{1}{N-c} & \text{otherwise} \end{cases} \quad (5)$$

To an external decision maker, the battery's charge status is only observable through indirect measurement such as the voltage. The true status cannot be deduced from such indirect measurement without some uncertainty. However, this uncertainty is bounded. Thus, we model the partial observability of the battery as degradation by bounded noise. That is, the mapping function from the voltmeter reading to the underlying battery state is degraded by uniform additive observation noise.

4.2 Modeling the Actions

The actions of the controller agent affect the battery states. By controlling variable resistances and shunt transistors, the agent makes a decision to attempt to charge all or part of the PV-generated electricity to the battery while diverting the remaining current to the non-critical local load. Any amount exceeding the maximum local consumption capability in the time interval, however, is automatically dumped. The first possible action approximately corresponds to an attempt to charge all of the PV output to the battery, with the consequence that the battery's charge level increases if $c < N$. The other possible actions result in charging the battery with less amounts or discharging the battery. Possible battery discharges within any given interval is as described in Section 4.1.

4.3 Utility Function

When the battery is in state s_b , the immediate utility function (*i.e.* reward) is defined as follows:

$$u(c) = \begin{cases} -U_u, & c \leq E \\ U_n, & c = N \\ -U_o, & c > N \\ \frac{U_n}{N-c+1}, & \text{otherwise} \end{cases} \quad (6)$$

where $c = f_b^{-1}(s_b)$ is the charge level, with $f_b^{-1}(\cdot)$ being an inverse mapping function; $U_u \geq 0$, $U_n > 0$ and $U_o \geq 0$ are constants. When the utility value is positive, it represents a reward; otherwise it is a penalty.

5 Experiments and Analysis

Further to our previous work that compared the impacts of different environment modeling strategies on the optimality of the planning [3], here we present some additional results. One of the comparison targets is the zeroth-order hold model, also known as the persistent environment model, where it is assumed that the environment state remains unchanged until the next observation becomes available. This is the optimal model when no prior knowledge is available about the process. In Figure 3 and Table 1, outcomes from normalized battery capacities resulting from random decisions, a POMDP based on the zeroth-order hold environment model and a POMDP based on the ϵ -Machine model are displayed. Figure 3 and Table 1

demonstrate that the learning-based modeling approach using the ϵ -Machine result in the most regular and stable patterns and enable the highest overall rewards.

Figure 4 shows two examples for visual assessment of the environment model constructed by the ϵ -Machine, where the model-generated process (green curve) is compared with the recorded PV process (red curve). The figure on the top panel is the reconstruction by the ϵ -Machine when the observed data were available and used by the construction process of the ϵ -Machine. By contrast, the observed data plotted figure on the bottom panel were not used during the construction of the ϵ -Machine. The large discrepancies at the bottom of the figures between the two curves is largely due to the quantization error.

Figure 5 provides a comparison of POMDPs operating at different temporal scales, as well as a multiscale solution. Visual inspection reveals that, for most of the time, the best scale has been successfully identified by the multiscale solution. Identification of the best scale for the problem is a significant benefit of using multiple scales. This solves the problem that the Markovian order of the environment process is unknown *a priori*, as mentioned earlier. Furthermore, the multiscale solution offers a more stable, higher average anytime-delivery capacity than any of the single scale solutions. The stability of the solution is manifested as the regularity of the pattern.

	Random Decisions	Zeroth Order Hold POMDP	POMDP based on ϵ Modeling
Process Knowledge	0	0	α
Intelligence in Decision Maker	0	1	1
Total Reward	2140232	5588844	7308421

Table 1. Comparison between methods: the random decision making method, POMDP with no knowledge of the PV process and optimal decision making through POMDP based on the ϵ -machine’s environment model. The total rewards were obtained by integrating over the trial period.

Conventional peaking power plants add significantly to infrastructure cost, are not economic to build and maintain and produce greenhouse gases. We have proposed a multiscale algorithm for autonomous agents to adaptively manage the operation of storage-enabled photovoltaic (PV) facilities by applying sequential decisions in a stochastic and noisy environment.

The environment is learned and modeled by an approach called an ϵ -Machine, which operates on a set of *a priori* determined temporal scales, to give the agent an additional degree of freedom when optimizing its control decisions. The operation of both the learning and the optimization on multiple scales solves the problem that the Markovian order of the environment process is unknown *a priori*. The systematic environment learning on multiple temporal scales makes the agent highly adaptable and able to optimally manage a PV-storage facility according to pre-determined objectives, such as a more stable and higher average anytime-delivery capacity.

Our focus has been on managing distributed PV facilities as potential replacement for conventional peaking power plants. Further comparison studies are currently underway using synthetic test patterns. A scheme using real-time observations and online management has also been implemented and is expected to produce further validation results in the near future.

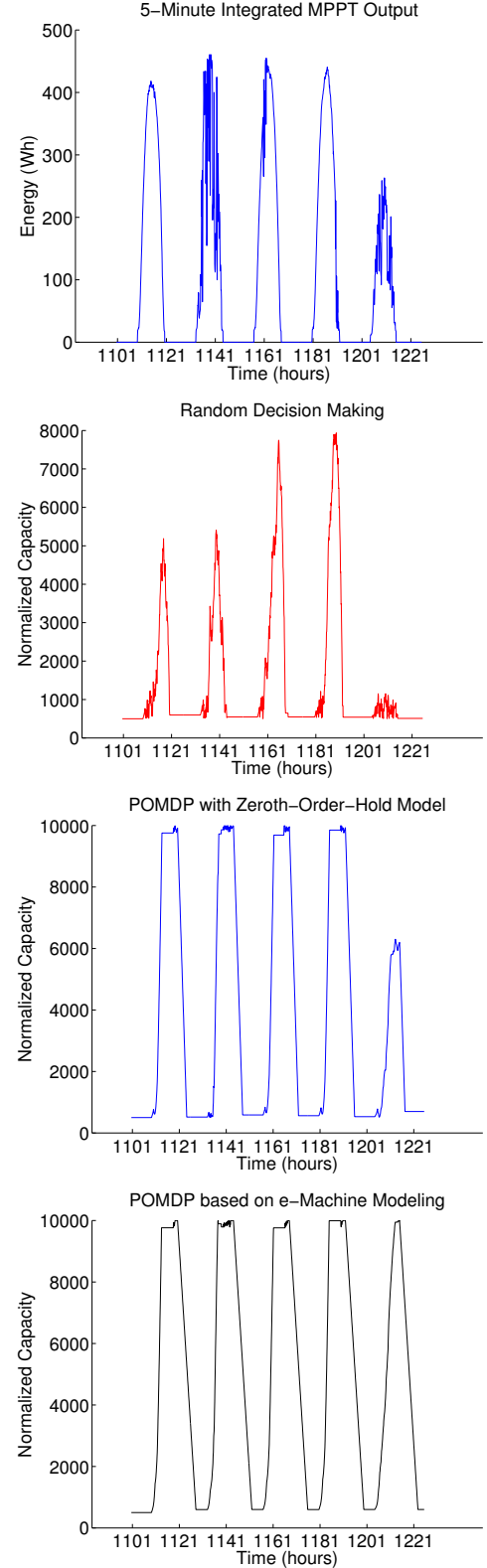


Figure 3. Given the MPPT output(top panel, see Section 3.2.3 for explanation), normalized battery capacities resulting from random decisions (second panel), a POMDP based on a zeroth-order hold environment model (third) and a POMDP based on the ϵ -Machine model (bottom).

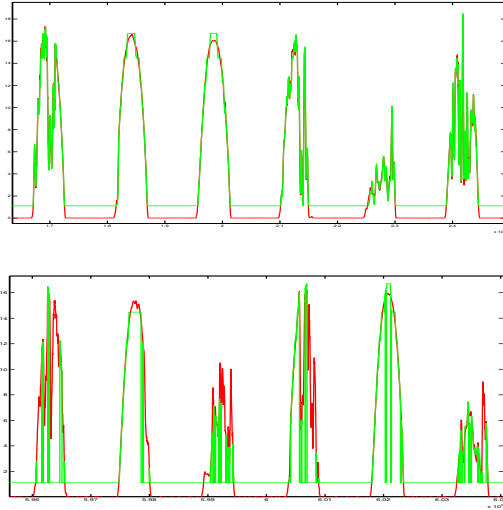


Figure 4. Visual assessment of the environment model constructed by the ϵ -Machine: Two sample outputs where the model-generated process (green curve) can be compared with the recorded PV process (red curve). The figure on the top panel is the reconstruction by the ϵ -Machine when the observed data were available and used by the construction process of the ϵ -Machine. By contrast, the observed data plotted figure on the bottom panel were not used during the construction of the ϵ -Machine. The large discrepancies at the bottom of the figures between the two curves is largely due to the quantization error.

REFERENCES

- [1] C.F. Chiasserini and R.R. Rao. A model for battery pulsed discharge with recovery effect. In *Wireless Communications and Networking Conference*, pages 636–639, 1999.
- [2] JongGyu Kim, John Paul M. Torregoza, InYeup Kong, and Won-Joo Hwang. Photovoltaic cell battery model for wireless sensor networks. *International Journal of Computer Science and Network Security*, 6(9B), 2006.
- [3] Rongxin Li and Peter Wang. Pattern learning and decision making in a photovoltaic system. In *7th International Conference on Simulated Evolution and Learning, SEAL 2008 Proceedings*, volume 5361 of *Lecture Notes in Computer Science*, pages 483–492. Springer, 2008.
- [4] Michael L. Littman, Thomas L. Dean, and Leslie P. Kaelbling. On the complexity of solving Markov Decision Problems. In *Proceedings of the Eleventh Annual Conference on Uncertainty in Artificial Intelligence (UAI-95)*, pages 394–402, Montreal, Québec, Canada, 1995.
- [5] Nicolas Meuleau, Kee-Eung Kim, Leslie P. Kaelbling, and Anthony R. Cassandra. Solving POMDPs by searching the space of finite policies. In *Proceedings of the Fifteenth Conf. on Uncertainty in Artificial Intelligence*, pages 417–426, 1999.
- [6] K. Murphy. A survey of POMDP solution techniques. Technical report, U.C. Berkeley, 2000.
- [7] D. Panigrahi T, D. Panigrahi, C. Chiasserini, S. Dey, A. Rao, R. Raghunathan, and K. Lahiri. Battery life estimation of mobile embedded systems. In *Fourteenth International Conference on VLSI Design*, pages 57–63, 2001.
- [8] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley-Interscience, April 1994.
- [9] V. Rao, G. Singhal, A. Kumar, and N. Navet. Battery model for embedded systems. In *18th International Conference on VLSI Design*, pages 57–63, 2005.
- [10] Nicholas Roy. *Finding Approximate POMDP Solutions Through Belief Compression*. PhD thesis, Carnegie Mellon University, 2003.
- [11] C. R. Shalizi. *Causal Architecture, Complexity and Self-Organization for Time Series and Cellular Automatas*. PhD thesis, University of Wisconsin at Madison, 2001.
- [12] Cosma R. Shalizi, Kristina L. Shalizi, and James P. Crutchfield. An algorithm for pattern discovery in time series. *Journal of Machine Learning Research*. To Appear.

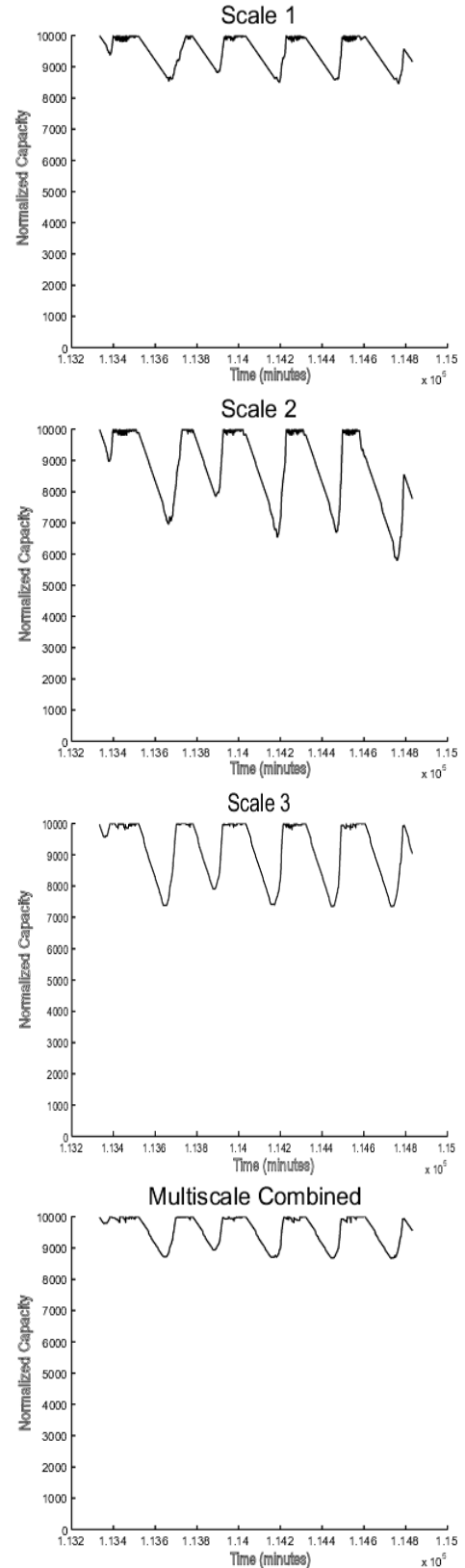


Figure 5. Normalized battery capacities resulting from two scales and a multiscale solution. Note that, for most of the time, the multiscale solution is able to identify the best scale. This solves the problem that the Markovian order of the environment process is unknown *a priori*. Furthermore, the multiscale solution offers a more stable, higher average anytime-delivery capacity than any of the single scale solutions.