Spring 2018

Learning-based short-time prediction of photovoltaic resources for pre-emptive excursion cancellation

Huaiqi Xie

Department: Electrical and Computer Engineering

LEARNING-BASED SHORT-TIME PREDICTION OF PHOTOVOLTAIC RESOURCES FOR PRE-EMPTIVE EXCURSION CANCELLATION

by

HUAIQI XIE

A DISSERTATION

Presented to the Graduate Faculty of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

DOCTOR OF PHILOSOPHY

in

ELECTRICAL ENGINEERING

2018

Approved by

Pourya Shamsi, Advisor
Mariesa L. Crow
Mehdi Ferdowsi
Jonathan W. Kimball
Bruce M. McMillin

PUBLICATION DISSERTATION OPTION

This dissertation consists of the following three articles which have been submitted for publication, or will be submitted for publication as follows:

Paper I: Pages 2-24 have been published to IEEE transactions on Smart Grid.

Paper II: Pages 25-37 have been published to Power & Energy Society General Meeting.

Paper III: Pages 38-64 will be submitted to IEEE transactions on Power Systems.

Paper IV: Pages 65-79 have been published to North American Power Symposium.

ABSTRACT

There is a growing interest in using renewable energy resources (RES) such as wind, solar, geothermal and biomass in power systems. The main incentives for using renewable energy resources include the growing interest in sustainable and clean generation as well as reduced fuel cost. However, the challenge with using wind and solar resources is their indeterminacy which leads to voltage and frequency excursions. In this dissertation, first, the economic dispatch (ED) problem for a community microgrid is studied which explores a community energy market. As a result of this work, the importance of modeling and predicting renewable resources is understood. Hence, a new algorithm based on dictionary learning for prediction of solar production is introduced. In this method, a dictionary is trained to carry various behaviors of the system. Prediction is performed by reconstructing the tail of the upcoming signal using this dictionary. To improve the accuracy of prediction, a new approach based on a novel clustering-based Markov Switched Autoregressive Model is proposed that is capable of predicting short-term solar production. This method extracts autoregressive features of the training data and partitions them into multiple clusters. Later, it uses the representative feature of each cluster to predict the upcoming solar production level. Additionally, a Markov jump chain is added to improve the robustness of this scheme to noise. Lastly, a method to utilize these prediction mechanisms in a preemptive model predictive control is explored. By incorporating the expected production levels, a model predictive controller is designed to preemptively cancel the upcoming excursions.

ACKNOWLEDGMENTS

First, my sincere respect and gratitude go to my advisor, Dr. Pourya Shamsi for his invaluable support and guidance on my research and study. The knowledge he has taught me, and the help he has offered to me with my work and life will continue to be encouraging and enduring in my career.

I am also extremely thankful to Dr. Mariesa Crow, Dr. Mehdi Ferdowsi, Dr. Jonathan Kimball and Dr. Bruce McMillin for their intellectual support, academic help and guidance on my research and Ph.D dissertation.

1I would like to express my thanks to all the colleagues and fellow friends at G2, G3 and G27 lab of Electrical Engineering department for their help on theoretical knowledge and simulation of this research.

Finally, I would also like to express my appreciation to my parents who have endowed me and given me constant support during these years. Their indispensable help and moral inspiration give me great power on my study and life.

TABLE OF CONTENTS

Page

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION

1. INTRODUCTION

Migration to microgrids provides new opportunities in energy planning and load flow management in Electric Power Systems (EPS). Economic dispatch (ED) problem in a power system is a well-known process and has been studied since the formation of power grids. In the first paper,we define a community microgrid as a microgrid that supports a community of residents. This microgrid does not have a single owner nor a central control system (it might have a central monitoring system). Within this microgrid, each node has full control over its local energy resources and can participate in microgrid energy planning based on its own personal benefits and without any obligations. After solving the economic dispatch problem, we realize that there are a lot of renewable energy resources in the microgrid and we need to know the power production information of these resources. So we proposed two methods to predict the upcoming solar power, the dictionary learning algorithm and a clustering-based Markov switch approach, which were presented in the second and third paper, respectively. Then a pre-emptive control scheme was proposed in the fourth paper that can incorporate any of the existing very short-term prediction methods to eliminate the power fluctuations in the system.

PAPER

I. ECONOMIC DISPATCH FOR AN AGENT-BASED COMMUNITY MICROGRID

Pourya Shamsi, Huaiqi Xie, Ayomide Longe, Jhi-Young Joo

ABSTRACT

In this paper an economic dispatch problem for a community microgrid is studied. In this microgrid, each agent pursues an economic dispatch for its personal resources. In addition, each agent is capable of trading electricity with other agents through a local energy market. In this paper, an energy market operating in the presence of the grid is introduced. The proposed market is mainly developed for an experimental community microgrid at Missouri University of Science and Technology (S&T) and can be applied to other distribution level microgrids. To develop the algorithm, first, the microgrid is modeled and a dynamic economic dispatch algorithm for each agent is developed. Afterwards, an algorithm for handling the market is introduced. Lastly, simulation results are provided to demonstrate the proposed community market and show the effectiveness of the market in reducing the operation costs of passive and active agents.

1. INTRODUCTION

Migration to microgrids provides new opportunities in energy planning and load flow management in Electric Power Systems (EPS). Economic dispatch (ED) problem in a power system is a well-known process and has been studied since the formation of power grids. Various algorithms for ED are available in the literature including static economic dispatch [1], dynamic economic dispatch [2]-[4], and dynamic economic dispatch with unit

commitment [2], [5]. In traditional microgrids, a central entity is responsible for monitoring, energy planning, and control of the microgrid [5]-[7]. Various research has studied aspects of distributed planning and markets in power systems [8]-[10]. Various market structures, game theoretic methods, and bidding policies have been applied to power systems [11]-[14]. Majority of electricity markets are competitive [15], [16]. In such markets, each participant provides a bid and the spot price is determined based on the ascending list of bids and the total demand. In many markets, auctions are closed and no information on submitted offers/bids are available to other agents. Even if offers/bids are openly announced, various techniques are required to gather information on inner states of competitors to generate a successful bid [17]. In this paper, a simple solution is provided that can be incorporated by residential agents in the energy planning and bidding mechanism.

Some common electricity markets are studied in [18]. Markets can be formed by independent agents and a utility or as a group of agents trading their resources [19]. In a simple auction market, operator clears the market by finding the intersection of the ascending supply and the demand [20], [21]. In this paper, the interest is on a close electricity market which is available to members of a community microgrid. In this market, bids are only submitted by the suppliers and not by the demand (demanding agents act passive). The focus is on the members of a local community who share their resources to minimize the total cost of acquiring their demand or to get profit from their excess resources. This process is also compatible with a demand responsive framework [22], [23] where the demand varies with the price. The main challenge in this market is the presence of the utility grid with a pre-determined rate for electricity. For this reason, if the clearing price of this market exceeds the regional price of electricity, then the grid will dominate the market. Hence, unlike a traditional market, in a distribution level community market, lower and upper bounds limit the spot price of the market.

In this paper, we define a community microgrid as a microgrid that supports a community of residents. This microgrid does not have a single owner nor a central control system (it might have a central monitoring system). Within this microgrid, each node has full control over its local energy resources and can participate in microgrid energy planning based on its own personal benefits and without any obligations (hence, the set of providers can vary with time). The incentive for the proposed definition is the structure of the community microgrid installed at Missouri University of Science and Technology (S&T) where the users can trade power without any interference from the utility grid. Although the algorithm does not depend on the size of the system, expansion of this algorithm to other communities has a fundamental requirement: There should be no utility meter inside the boundaries of the microgrid. The utility meter should be placed at the Point of Common Coupling (PCC). This is to prohibit the local electric cooperative from monitoring the flow of power within the microgrid. Otherwise, the price of selling and purchasing energy will be set by the electric cooperative. After introduction of this experimental community microgrid, a dynamic economic dispatch method for each agent is reviewed which will be used to derive the bids. Then the market is introduced and the overall algorithm is provided. Simulation results are provided to demonstrate the behavior of this system and cost reduction due to internal trades.

2. A COMMUNITY MICROGRID: GREEN COMMUNITY

The selected microgrid is based on *Solar Village* microgrid at Missouri S&T. This microgrid consists of four houses with their individual access to solar energy resources and storage systems. Also, a central 60kWh battery storage system with a 50kW bidirectional inverter and a 5kW Fuel Cell (FC) Distributed Energy Resource (DER) are shared among these houses and are managed by a central microgrid controller. The physical microgrid is shown in Fig. 1. The schematic of this system is illustrated in Fig. 2a.

Figure 1. Solar Village at Missouri S&T.

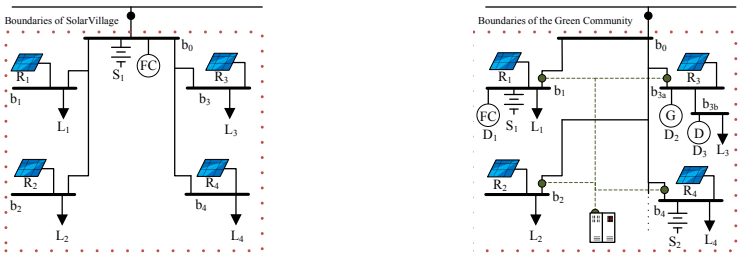


Figure 2. Schematics of (a) *Solar Village* phase I, (b) *Solar Village* phase II which is called the *Green Community*.

The overall microgrid is a property of S&T and the local utility provider, Rolla municipal utility, has no information on the power flow within this microgrid (which is part of S&T's agreement). Currently, the university is paying for the electricity usage of all tenants through the installed smart meter shown with a *black* circle in Fig. 2a. In the second phase of this project which is called the *Green Community*, several houses and local businesses will form a microgrid. This microgrid is shown in Fig. 2b. Currently, this system is under construction and we are interested in developing a market structure for energy trades within this community microgrid. In Fig. 2b, R_i, S_i, and D_i represent renewable energy resources, storage systems, and dispatchable generation systems, respectively. In this system, each house or business will pay for their individual electricity usage. However, this payment will be in the form of a cost share on the single electricity bill for the overall microgrid which is recorded by the utility meter at the PCC. Electricity usage of each house is recorded by the microgrid controller using multiple smart meters (shown in *green*).

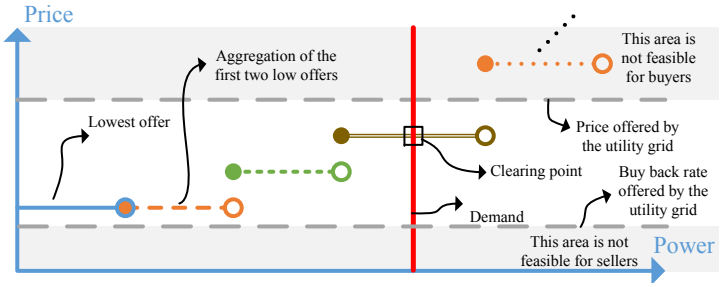


Figure 3. Aggregation of the bids.

2.1. COMMUNITY ENERGY MARKET

In this market, the goal is to find the spot price of electricity based on available bids on offered energy resources. At each time step (usually an hour), each agent will announce whether it demands energy or sells excess energy. Hence, the list of bidders will change at each time step (in this sense, the market is dynamic). If an agent is a buyer, it announces the amount to be purchased (buyers are passive). If an agent is a seller, it announces available power levels with their corresponding price. A seller can have multiple bids for its energy resources. A simple market clearing process is performed based on the intersection of the supply and the demand. Sorting of the bids is based on the ascending price rates of electricity. Hence, the market operator will aggregate the received bids as shown in Fig. 3. The market operator will find the spot price by intersecting the demand and the ascending plot of the bids. A difference between this market and an ordinary market is the presence of the utility grid. With respect to the power levels of the microgrid, utility grid has no limit in offering power at its set price. Therefore, if any offer is higher than the price of electricity from the grid, the offer is naturally neglected and the required demand is purchased from the grid. Also, there can be a case where large incentives are in place for utilization of distributed resources. Hence, grid can buy electricity at a higher price than what it sells. This scenario is not suitable for a microgrid with multiple nodes and one PCC as the sum of the power will pass through the PCC. So even if the sellers want to sell their energy to

the utility grid, they first need to supply the local demand. Therefore, first, they need to sell their electricity at a lower clearance price of the market, and then sell the excess energy to the grid at the higher rate of the incentives. In this scenario, users with large distributed resources will not benefit from being a member of the microgrid and they might seek their own connection to the grid. Fortunately, this is not the case for the microgrid located at Missouri S&T. In this region, the buy back rate is at most $0.04/kWh which is about half of the cost of purchasing electricity. Hence, sellers will profit if they sell power locally at a higher price than selling it back to the grid. There are two possible outcomes for this market.

2.1.1. There Is More Total Demand Than The Total Offer. In this case, to meet the demand, power has to be purchased from the grid. Hence, the intersection of the demand and the offer occurs on the price level of the grid. Therefore, in this case, the spot price will be equal to the price of the electricity from the utility grid and the bidders will receive this rate.

2.1.2. The Total Offer Is More Than The Demand. In this case, first, the market is cleared by meeting the local demand using the ascending price curve. Afterwards, the flow of power can be outwards at the point of PCC and the sellers can sell their power back to the utility grid. Usually this process occurs at a lower rate as it was mentioned that the average rate for our geographical location is $0.04/kWh. Hence, the sellers can decide whether it is profitable for them to sell power at this rate or not.

3. ECONOMIC DISPATCH FOR A SINGLE ENTITY

The problem of Economic Dispatch (ED) is to minimize the total cost of energy within a window of optimization. $\mathcal{D}_i = \{g_i, d_{i_1}, d_{i_2}, \cdots, b_{i_1}, b_{i_2}, \cdots\}$ is the set of dispatchable resources at node i (each agent can posses multiple resources of a same kind). In particular, P_{g_i} is the power flowing from the distribution network to agent i. $\bar{\mathcal{D}}_i = \{r_{i_1}, r_{i_2}, \cdots\}$ is the set of intermittent resource, and P_{l_i} is the load. $E_{i_j}(t)$ is the energy stored in the j-th battery

resource at node i. The economic dispatch problem for agent $i \in \{1, 2, ..., N\}$ is formulated as

$$\min_{P_k|k\in\mathcal{D}_i} C = \sum_{t=1}^{T_i} \sum_{k\in\mathcal{D}_i} C_k P_k(t) \tag{1a}$$

$$s.t. \sum_{k\in\mathcal{D}_i} P_k(t) + \sum_{k\in\bar{\mathcal{D}}_i} P_k(t) = P_{l_i}(t) \tag{1b}$$

$$P_k^{min} \leq P_k(t) \leq P_k^{max},\ k \in \mathcal{D}_i \tag{1c}$$

$$E_{i_j}^{min} \leq E_{i_j}(t) \leq E_{i_j}^{max},\ j \in \{1,\cdots, n\} \tag{1d}$$

$$E_{i_j}(t) = E_{i_j}(t-1) + P_{b_{i_j}}(t).\Delta t \tag{1e}$$

where T_i is the length of the dispatch window (optimization horizon). This value is usually selected to be 24-hours to support a day of dispatch. Larger values of this dispatch window results in a better sub-optimal solution at a higher computational costs. Δt is the time period between two consequent dispatch steps. n is the number of batteries at node i. Power balance equation is calculated in (1b). Each energy resource has power limitations which are considered in (1c). Problem (1) can be also solved using Dynamic Programming (DP). In this way, the problem can be reduced to subproblems which are solved independently. If a node i owns d dispatchable resources including $b < d$ battery storage systems, by using DP, a $T_i \times d/\Delta t$ dimensional problem will be reduced to $N_{stp_1} \times \cdots \times N_{stp_b} \times T_i/\Delta t$ problems of $(d - b)$ dimensions where N_{stp_j} is the number of steps selected for the dispatch of the j-th battery system. Although the motive for using DP instead of the above linear programing is not clear yet, it will be shown that using DP, each agent needs to only update a few nodes on the DP graph during two subsequent optimization cycles. Hence, on the long run, DP will impose a much lower computational burden. To use DP, the possible levels of energy in each battery system is discretized to a set of levels with a step size of E_{stp}. The optimization is performed every Δt (usually an hour). Therefore, the dispatch level of each battery is no longer an independent variable and is calculated as $P_{b_{i_k}}(t) = (E_{i_k}(t + 1) - E_{i_k}(t))/\Delta t$.

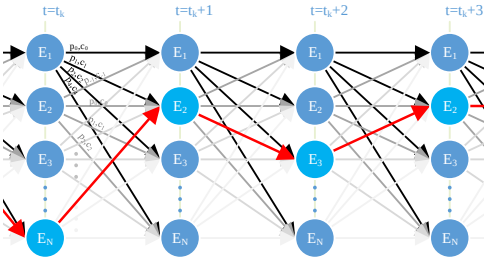


Figure 4. Dynamic programing graph of the ED problem.

Feasible dispatch levels for the battery should comply with (1d) and (1e), otherwise, the cost of transition from $E_{i_k}(t)$ to $E_{i_k}(t+1)$ is infinity. After solving each sub-problem, a graph of all possible transitions is formed. In this graph, nodes are possible energy levels in battery resources at a time t_k. Hence, set of graph columns are defined as $\mathcal{N} = \{\mathcal{N}_1, ..., \mathcal{N}_{T_i}\}$, $\mathcal{N}_k = \{E_{i_j}^{min}, E_{i_j}^{min} + E_{stp}^{k}, ..., E_{i_j}^{max}\}$ while the set of directed transitions (arcs) are defined as $\mathcal{W} = \{w_1, ..., w_{T_i}\}$ and $w_k \in \mathcal{N}_k \times \mathcal{N}_{k+1}$ $(k \in \{1, \cdots, T_i - 1\})$. Fig. 4 illustrates the transition graph for this DP by illustrating the directed graph $(\mathcal{N}, \mathcal{W})$. Based on the arc weights, the shortest path (lowest sum of weights) from the last column (i.e. $E_{end}^{T_i}$) to the starting energy level (i.e. E_{start}^{1}) is calculated using dynamic programming. This calculation will not only define the shortest path, but also will define the final energy level. We have assumed that the starting energy level is known which is the level at $t = t_0$. The energy level in the battery systems will create a dependency between the optimal solution of the dispatch at each time step. Hence, the economic dispatch problem is in fact an infinite horizon optimization problem where the starting point is known and the optimal path can be calculated using the extended Bellman method. However, there is no significant point in solving the solution for an infinite horizon case as the true stochastic variations of loads, intermittent resources, and policies selected by other agents are not known. Therefore, for long optimization windows, the covariance of stochastic process will become large and the optimization cannot provide any practical benefit compared to a smaller time window. For this reason, in many applications, the time window for a dynamic economic dispatch is

selected as an integer multiple of days such as T_i = 24h or 48h. Also, since the problem is now a sub-optimal solution of the original infinite horizon problem, it is sufficient to find the best solution without any concerns for upcoming windows. Therefore, by knowing the starting energy level for the battery system, one can find the shortest path to $t = t_0+T_i$ without enforcing any constraints on the final state of the battery. In the simulations provided, each agent will have a different optimization window.

4. ELECTRICITY MARKET IN A COMMUNITY MICROGRID

4.1. ANNOUNCING THE BIDS

The proposed method is mainly developed for linear cost functions. Recently a non-linear non-convex auction based method has been introduced which considers transmission losses [10]. In this work, a set of bids for various power levels is generated by each agent and transmitted to the neighboring agents. Due to non-convexity of the problem, each agent requires to provide a set of feasible operation points. Instead, in our method, the cost functions are linear and agents need to provide a list of bids including the rating of their resources and the price of each resource. If a resource has a non-linear but a convex cost function with a minimum located at zero (such as a second order function), then the agent can break its operation region into a set of linearized cost functions. Afterwards, the agent provide bids regarding the capacity of each linearized section and the corresponding price. In order to solve the ED problem, an agent needs the cost function of the grid C_g in (1). For positive acquires from the grid, this value is at most the price of electricity offered by the electric cooperative. However, this value can be lower as the clearance price of the market depends on available offers. Hence, an agent can have a price estimate of the grid for this time period as $\hat{C}_g(t)$. First, this agent can assume $\hat{C}_g(t)$ is equal to the price offered by the local electric cooperative. Eventually, this agent can train a price model based on the observations of the price at each market cycle. For instance, a simple learning mechanism

as $\hat{C}_g^{(i+1)}(t) = \hat{C}_g^i(t) + \gamma(C_g(t) - \hat{C}_g^i(t))$ where $\hat{C}_g^i(t)$ is the estimate of the spot price at time t during the i-th cycle of the market procedure. $C_g(t)$ is the clearing price of the market at the time t during the i-th cycle of operation of the market. γ is the learning (filtering) rate. This learning mechanism is based on a one dimensional recursive least squares which is proven to converge for economic systems with hidden information layers [24]. Although this simple method does not guarantee a boundary on the error, it can provide good estimations over the long run of the algorithm. In practice, agents can utilize more sophisticated estimation and learning mechanisms. At a time t, using the vector of price estimates for each optimization step and for a T_i window of time in the future, each agent solves the optimization problem and derives the optimal dispatch. Based on the dispatch, if $P_{g_i}(t) \geq 0$, then this agent is a buyer and acts passively in the market. This agent will only announce the required amount of power. If $P_{g_i} < 0$, then it is optimal for the agent to sell power back to the grid. Agent will generate the ascending cost plot of its resources. Lower cost resources will be used to supply the internal demand (i.e. $P_{l_i}(t)$). The remainder of the plot is announced to the market as a set of available capacity and the price of each capacity.

4.2. CLEARING THE MARKET

At this point, the market has received all the demands and offers for the time step t. If the demand is higher than the available capacity, then the remainder of the power has to come from the utility grid. Since the grid is an infinite capacity market (with respect to the nominal rating of the microgrid), the price of the grid will become the dominant price as the intersection of the demand and the bids occur on the price of the grid. Therefore, for the case where $\sum_k P_{g_k} > 0$, the clearing price of the market is price of the grid and every seller will receive this rate. If $\sum_k P_{g_k} < 0$, then there are more offers than the demand and the market can clear without considering the grid. As it was shown in Fig. 3, the clearance price of the market should remain between the price at which the utility grid sells power and the rate at which it buys back power. If the spot price is higher than the grid's

price, buyers would complain and will demand their individual connection to the utility grid. If the spot price is lower than the grid's buyback rate, then the sellers would seek their direct connection to the grid. Hence, to have a feasible and sustainable operation of the community microgrid, the spot price is bounded within these two margins. After setting the sport price, the dispatch is announced to the agents. At this time, the demand is fulfilled. However, some of the bids are not used. Based on the preferences of the remaining bidders, their capacity can be sold to the grid. However, this rate is the low buyback price rate of the grid and agents should verify if it is still in their benefit to do so. After clearing the market, the system will redo this process for the next time step. Here, it was assumed that the market is static and modifications of bids are not applicable. However, one can simply allow for modification of the bids and agents can compete further by modifying their bids. In this case, a maximum limit on the number of iterations is necessary to ensure a final settlement before the dispatch period begins.

4.3. POST MARKET PROCEDURES

At this time, the dispatch levels and the spot price of the electricity for the time step t are derived. Agents will use the information regarding the amount of power that was traded as well as the spot price to form an estimation for the similar time period in upcoming days. In a simple approach, each agent can track the spot price using a learning mechanism such as $\hat{C}_g^{(i+1)}(t) = \hat{C}_g^i(t) + \gamma_1(C_g(t) - \hat{C}_g^i(t))$ and track the demand level as $\hat{P}_g^{(i+1)}(t) = \hat{P}_g^i + \gamma_2(P_g^i(t) - \hat{P}_g^i(t))$. Tracking the demand is important for the sellers as the local demand is cleared at a higher rate than what the grid pays for electricity. So an agent needs to know how much power can be sold at a rate of the market and the remainder will be sold at the rate of the grid. The importance of the dynamic programing appears in the post market step. If an agent updates the price and demand estimates only for a similar time period, there is no change in the DP graph of upcoming hours. Therefore, the agent can simply update the DP graph by calculating the affected sub-problems without re-calculating

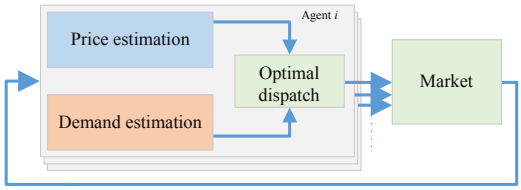


Figure 5. The proposed community economic dispatch scheme.

Algorithm 1 A community energy market

Initialize a 24h vector of $\hat{C}_g(t)$ and $\hat{P}_g(t)$ with a prior assumption based on the location. Set $t = 0$. While 1
for i **do**=1:#(Agents) ▷ Announcing bids
    Solve (1) for $t = \{t+1, t+2, \cdots, t+T_i\}$.
    if $P_{g_i}(t+1) > 0$ **then**
        Announce the total demand $P_{g_i}(t+1)$.
    else
        Announce the total offer $|P_{g_i}(t+1)|$.
        Breakdown resources used to form $P_{g_i}(t+1)$:
        Per resource, announce the capacity/price rate.
        Do not announce any resource with a price rate
        higher than grid's rate at $t+1$.
Place all bids in $O = [o]_j = [p_j, c_j]$ where p_j is the
capacity of the j-th bid and c_j is the corresponding rate.

the whole graph. Whether agents use DP or not, they need to recalculate their optimal dispatch for the upcoming hours and announce their bids for the next cycle of the market. The overall process for the proposed algorithm is shown in Fig. 5. It should be noted that the decision making policies and the estimation methods are not discussed in details in this paper. The main objective of this paper is to provide a market procedure for a microgrid in presence of the utility grid. The simple examples of recursive estimation methods provided can effectively handle the ED for residential agents as it is shown later in the simulation results. The summary of the proposed method can be described as Algorithm 1.

Algorithm 2 A community energy market - Continued

if Demand > total bids **then** ▷ Clearing the market
    Set $C_g(t+1)$ = grid's rate at time $t+1$.
    Use all bidders, buy the remaining demand from
    the utility grid.
else
    Sort O and find the intersection of the cumulative
    bids and the demand. Set this bid as $C_g(t+1)$.
    Limit: grid's buyback rate $\leq C_g(t+1)$.
    Pay the selected bidders at the rate $C_g(t+1)$.
    Remainder of the bids can be sold back to the utility
    grid at a rate $C'_g(t+1)$ = grid's buyback rate.
Enforce the dispatch. $t = t+1$.
Update the estimations: ▷ Post-market process
Each agent can track the settled $C_g(t)$ and update $\hat{C}_g(t)$.
Each agent can track the demand.
Each agent can track its estimated share of the market.

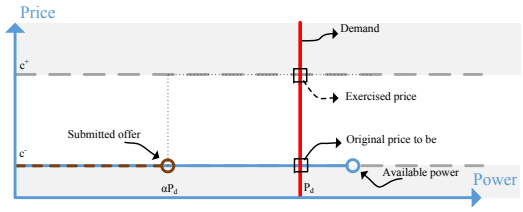


Figure 6. Forcing a higher sport price.

4.4. MARKET POWER EXERCISE

The proposed market seems to be vulnerable to market exercises. In this market, an agent can bid less to increase the spot price. As it was mentioned earlier, if sufficient resources are not available, then the spot price of the market will be equal to the price of the grid. In an extreme scenario, consider a microgrid with only one seller. As shown in Figure 6, this seller can support the demand P_d at the low price of c^-. The agent prefers a higher price of c^+. Hence, it provides a bid at αP_d. As a result, the spot price of the market will be the price of the grid which is c^+. If $\alpha \rightarrow 1^-$, this agent's income will increase from $c^- P_d$ to $c^+ P_d$. However, for this scenario to occur, agents should be aware of the demand and the offers. To prevent such practices, several conditions are added to the market. First, we

Table 1. Simulated resources/costs based on Fig. 2b.

Node	Resource (id.)	Power [kW] min.	Power [kW] max.	Energy [kWh] min.	Energy [kWh] max.	Price $/kWh
b_1	Solar PV (R_1)	0	1	-	-	0.00
	Battery (S_1)	-1	1	0.5	2	0.00
	Fuel Cell (D_1)	0	2	-	-	0.07
b_2	Solar PV (R_2)	0	0.8	-	-	0.00
b_3	Solar PV (R_3)	0	0.6	-	-	0.00
	Diesel gen. (D_2)	0	2	-	-	0.08
	Gas gen. (D_3)	0	1	-	-	0.06
b_4	Solar PV (R_1)	0	0.5	-	-	0.00
	Battery (S_2)	-2	2	1	4	0.00

prefer a non-iterative bidding where only one set of bids are collected for each cycle. Also, no information regarding the demand or offers are available to the sellers. Therefore, a seller does not know the demand of the microgrid or the offers provided by other sellers. Also, this market is specifically designed for a community microgrid. If any market exercises are observed, the home owners association of this community can fine the miss behaving agent or prohibit this agent from further participating in the market.

5. CASE STUDY

In this section, several case studies are provided for the microgrid shown in Fig. 2b. With respect to this figure, parameters of each load and resource are presented in Table 1. In this scenario, we assume that the price of buying energy from, and selling energy back to the grid are given by Fig. 7. To solve the problem for each agent, (1) was used. For each node of the DP graph, linear programing was used to find the cost of the transition based on the dispatch level of the battery. Lastly, the overall optimal path was found by finding the shortest path on th DP graph. Using a standard 4-core Intel 4-th generation i-7 laptop and MATLAB, the 48 hour optimal dispatch for agent b_1 was solved in 50ms. In a real-world implementation, each agent will solve the ED and will update his/her estimates of the market

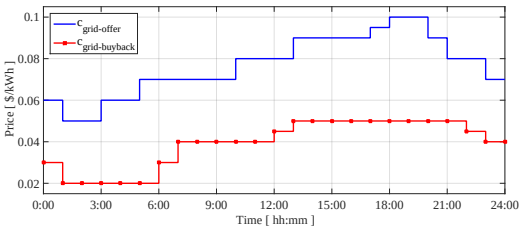


Figure 7. The 24h price of energy to and from the grid.

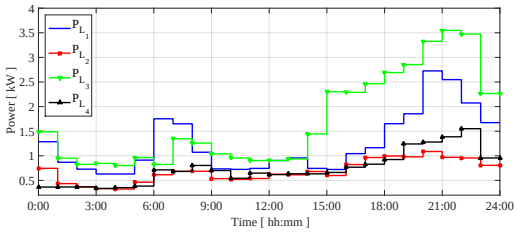


Figure 8. The 24h load profile of each bus.

behavior locally. However, we solved all the stages of Algorithm 1 using this MATLAB model. The overall processing time for 1 cycle of the market including ED of the 4 agents as well as the clearing process and post-market updates is 0.1s. It should be noted that agents b_2 and b_3 have an optimization window of 1 hour due to the lack of storage systems and agent b_1 has an optimization window $T_4 = 36$ hours to maintain generality. Load located at each bus are presented in Fig. 8. Solar production profile is depicted in Fig. 9. It should be noted that due to the close proximity of houses, their solar profile is similar and only varies in amplitude. Buses b_1 and b_4 have energy storage systems. Therefore, to perform an ED, these buses need to consider an optimization over a window of time. As it was mentioned before, the selection of the window itself is a trade-off between optimality and

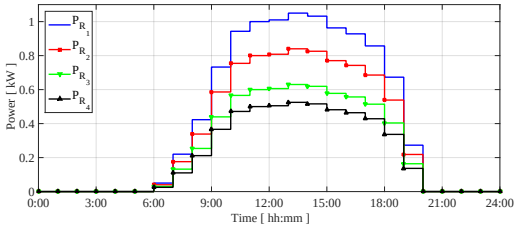


Figure 9. The 24h solar production profile of each bus.

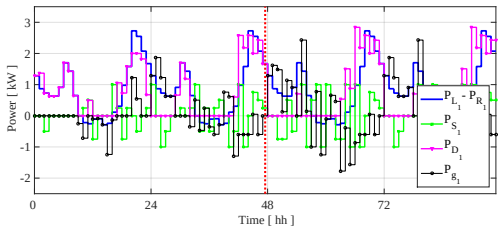


Figure 10. ED performed by b_1, hours 48-96 are the optimal dispatch in the future based on the knowledge at the present time ($t = 48h^-$).

computational complexity. In many low power applications, a 24 hour window is selected for ED. Without a loss of generality, b_1 selects its optimization window to be 48h and b_4 selects 36h. Also, the algorithm is started with an assumption that the price of electricity is \$0.07/kWh at each time step. Each bus assumes that it is possible to sell 1kWh to the market (at a higher rate than what the utility buys back at). These are starting assumptions and based on each agent's learning mechanism, the agent will soon find a better estimate of each of these parameters as is shown later. With the above assumptions, the system is simulated for 48 hours or equivalently, 48 market cycles for dispatch windows of 1 hour each. Currently, the market is not settled for the 48-th hour. Therefore, up to the hour 47, the price of the market is known and energy has been traded. We are looking at the time instance of hour 48 when each agent has calculated its optimal bid [demand] to [from] the market. At this time step, agent b_1 has calculated an ED with an optimization window of 48h in the future. Fig. 10 illustrates the dispatch performed by b_1 through time. The vertical red line denotes the present time. This figure illustrates the evolution of the optimal dispatch through time. It can be observed that as the number of market cycles increased, the ED solution is changing. During the first day, for 24h, there is no correct estimation of the price of the grid/market. Therefore, this agent is assuming \$0.07/kWh as the price of the electricity from the grid (which is a fair assumption throughout the U.S.). Also, this agent assumes that there is a chance of selling 1kW to the neighboring agents at any time. In the second day when there are prior knowledge of the trades which took place in the first day,

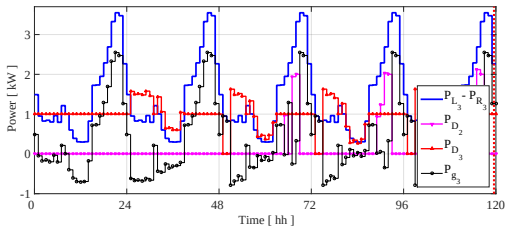


Figure 11. ED performed by b_2 during the first 5 days, ED for hours 1 thorough 119 has already been applied to the system and the solution for $t = 120h$ is awaiting execution.

this agent has a better understanding of the possible spot prices of the market at each market cycle. It can be observed that the battery storage system is optimally charged at times with lower cost of electricity and is sold to other agents during the peak usage times. The period between 48h to 96h shows the ED for a 48h window in the future. However, this agent will update this dispatch after every market cycle to maintain its optimality based on the settling price of the market and based on the microgrid demand. Similar to this agent, other agents dynamically solve the ED problem and participate in the market. The agent at node b_3 has no energy storage system. Hence, for this agent, there is no need to solve a dynamic ED in time and derivation of the ED for only one cycle in the future will suffice. Fig. 11 illustrates the dispatch for this agent. Based on the price of each resource provided in Table 1 and the starting assumption for the price of electricity to be sold to the grid, during the first day, this agent tends to use its gas generator to sell power to the microgrid. This resource is only \$0.06/kWh and can easily compete in the market. As more information is collected in the first day, on the second day, ED involves a significant dispatch for this gas resource. However, it can be observed that it takes one additional day for this agent to get a sufficiently accurate estimate for the settling price of the market to start using its diesel resource at a rate of \$0.08/kWh. To observe the evolution of the market and growth of the benefits for each agent, we consider the value function of the ED of agent b_1. At $t = 1$, this agent has no realistic estimate of the market. Hence, it is calculating the ED based on the prior assumptions and without high expectations of profits. Fig. 12 illustrates the

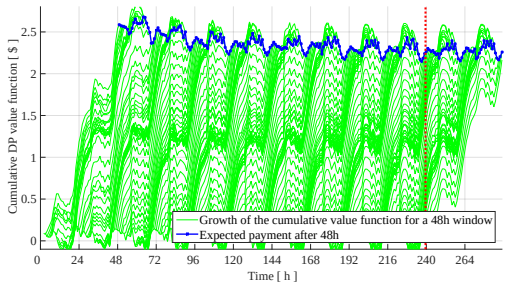


Figure 12. Reduction in the expected total cost of operation for a 48h optimization window of agent b_1.

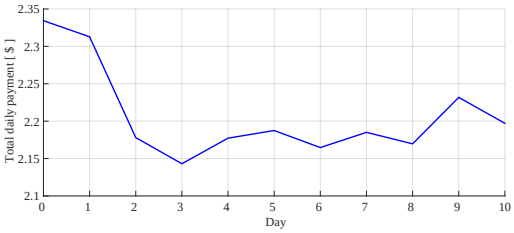


Figure 13. Daily cost of energy for b_3 for the first ten days of operation.

plot of the cumulative value functions for the dynamic ED of agent b_1 with its 48h dispatch window. Each green curve in this figure represents the growth of a the value function of the optimization period $t = t_o$ to $t = t_o + T_1$ where $T_1 = 48$h is the dynamic ED window for agent b_1. Each value function starts from zero and grows based on the expected cost of energy during the upcoming 48h. At $t = t_o + T_1$, the final value of the total expected cost of energy is achieved. These final points are connected using a blue line with small squares. It is observed that the expected final cost of a 48-hour operation is decreasing as the agent gains more knowledge about the operation of the market and can integrate more accurate pricing in its ED. In addition to profits for agents with storage systems, other agents can benefit from this market. For instance, Fig. 13 illustrates the daily cost of energy paid by agent b_3. It can be observed that the total is higher for this agent during the first two days of operation. However, as this agent acquires an estimate of the price/demand of the market, it can utilize its gas and diesel resources to reduce its costs of operation. Lastly, we observe the evolution of the spot price of the market. As it was mentioned before, this

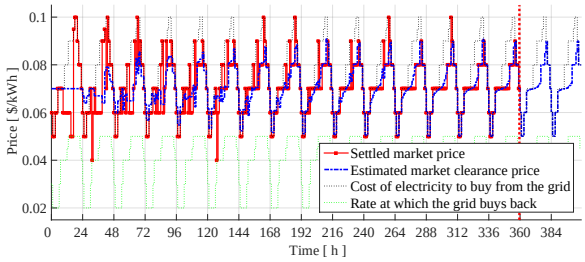


Figure 14. Evolution of the market spot price.

price is limited to the price of electricity from the grid and buy back price of the grid. On the first day, agents start with an assumption of \$0.07/kWh. However, as the market operates, new prices are settled and agents update their estimation. To reduce the number of days required, large learning factor for both cost estimation and demand estimation are used ($\gamma_1 = \gamma_2 = 0.3$). In Fig. 14, red line-dot illustrates the price settlement of the market for each cycle. It can be observed that during the daytime, the market price is settled to a lower value than the offer from the grid. This shows that the microgrid has enough capacity to support its demand and agents with distributed resources are able to compete in a local community market to sell their excess energy (due to the large number of solar resources). Also, since the spot price is lower than the regular price of electricity from the grid, agents who buy energy are benefiting as well. The blue line-dash curve illustrates the estimation of the price used by agent b_1. It is shown that agent b_1 is improving its price estimation and can better derive the optimal dynamic ED to dispatch its resources. For instance, for the 48 hours of dispatch after the current time $t = 360$h, this agent is utilizing the shown curve as the cost model for the market which is much more accurate than the stating constant assumption of \$0.07/kWh.

6. CONCLUSIONS

In this paper, a market for economic dispatch in a community microgrid was introduced. This market was based on a standard auction market with passive buyers where sellers provide bids by announcing their available capacity and its linear cost model. Market was cleared by intersecting the demand and the ascending list of offers. It was shown that in such community markets, agents can estimate the operation of the market and effectively dispatch their resources. Since the spot price of the market is always lower or equal to that of the grid and higher or equal to the buyback price of the grid, both sellers and buyers will always benefit from participating in this market.

REFERENCES

[1] S.-J. Ahn, S.-R. Nam, J.-H. Choi, and S.-I. Moon, March 2013, "Power scheduling of distributed generators for economic and stable operation of a microgrid," Smart Grid, IEEE Transactions on, vol. 4, no. 1, pp. 398-405.

[2] T. Nguyen and M. Crow, May 2012, "Optimization in energy and power management for renewable-diesel microgrids using dynamic programming algorithm," in Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2012 IEEE International Conference on, pp. 11-16.

[3] S. Bae and A. Kwasinski, Dec 2012, "Dynamic modeling and operation strategy for a microgrid with wind and photovoltaic resources," Smart Grid, IEEE Transactions on, vol. 3, no. 4, pp. 1867-1876.

[4] Y. Riffonneau, S. Bacha, F. Barruel, and S. Ploix, July 2011,"Optimal power flow management for grid connected pv systems with batteries," Sustainable Energy, IEEE Transactions on, vol. 2, no. 3, pp. 309-320.

[5] D. Olivares, C. Canizares, and M. Kazerani, July 2014,“A centralized energy management system for isolated microgrids,” Smart Grid, IEEE Transactions on, vol. 5, no. 4, pp. 1864-1875.

[6] S.-J. Ahn, S.-R. Nam, J.-H. Choi, and S.-I. Moon, March 2013,“Power scheduling of distributed generators for economic and stable operation of a microgrid, ” Smart Grid, IEEE Transactions on, vol. 4, no. 1, pp. 398-405.

[7] M. Hopkins, A. Pahwa, and T. Easton, June 2012, “Intelligent dispatch for distributed renewable resources,” Smart Grid, IEEE Transactions on, vol. 3, no. 2, pp. 1047-1054.

[8] S. Yang, S. Tan, and J.-X. Xu, Nov 2013, “Consensus based approach for economic dispatch problem in a smart grid,” Power Systems, IEEE Transactions on, vol. 28, no. 4, pp. 4416-4426.

[9] G. Binetti, A. Davoudi, F. Lewis, D. Naso, and B. Turchiano, July 2014,“Distributed consensus-based economic dispatch with transmission losses,” Power Systems, IEEE Transactions on, vol. 29, no. 4, pp. 1711-1720.

[10] G. Binetti, A. Davoudi, D. Naso, B. Turchiano, and F. Lewis, May 2014, “A distributed auction-based algorithm for the nonconvex economic dispatch problem,” Industrial Informatics, IEEE Transactions on, vol. 10, no. 2, pp. 1124-1132.

[11] P. Nguyen, W. Kling, and P. Ribeiro, March 2013, “A game theory strategy to integrate distributed agent-based functions in smart grids,” Smart Grid, IEEE Transactions on, vol. 4, no. 1, pp. 568-576.

[12] C. Colson and M. Nehrir, March 2013, “Comprehensive real-time microgrid power management and control with distributed agents,” Smart Grid, IEEE Transactions on, vol. 4, no. 1, pp. 617-627.

[13] Y. Shoham and K. Leyton-Brown. Cambridge University Press, 2008, Multiagentsystems: Algorithmic, game-theoretic, and logical foundations.

[14] Y. Wang, W. Saad, Z. Han, H. Poor, and T. Basar, May 2014, "A game-theoretic approach to energy trading in the smart grid," Smart Grid, IEEE Transactions on, vol. 5, no. 3, pp. 1439-1450.

[15] A. K. David and F. Wen. IEEE, 2000, "Strategic bidding in competitive electricity markets: a literature survey," in Power Engineering Society Summer Meeting, vol. 4, pp. 2168-2173.

[16] G. Jimenez-Estevez, R. Palma-Behnke, R. Torres-Avila, and L. Vargas, Nov 2007, "A competitive market integration model for distributed generation," Power Systems, IEEE Transactions on, vol. 22, no. 4, pp. 2161-2169.

[17] F. Wen and A. K. David, 2001, "Optimal bidding strategies and modeling of imperfect information among competitive generators," Power Systems, IEEE Transactions on, vol. 16, no. 1, pp. 15-21.

[18] D. M. Biagi, 2004, "Transaction automation on the internet: open electronic markets, private electronic markets and supply network solutions," Int. J. of Electronic Business, vol. 2, no. 6, pp. 674-685.

[19] L. Xiao, N. Mandayam, and H. Vincent Poor, Jan 2015, "Prospect theoretic analysis of energy exchange among microgrids," Smart Grid, IEEE Transactions on, vol. 6, no. 1, pp. 63-72.

[20] F. Zhao, P. Luh, J. Yan, G. Stern, and S.-C. Chang, Feb 2010, "Bid cost minimization versus payment cost minimization: A game theoretic study of electricity auctions," Power Systems, IEEE Transactions on, vol. 25, no. 1, pp. 181-194.

[21] J. Contreras, O. Candiles, J. de la Fuente, and T. Gomez, Feb 2001, "Auction design in day-ahead electricity markets," Power Systems, IEEE Transactions on, vol. 16, no. 1, pp. 88-96.

[22] H. Mohsenian-Rad and A. Davoudi, Sept 2014, "Towards building an optimal demand response framework for dc distribution networks," Smart Grid, IEEE Transactions on, vol. 5, no. 5, pp. 2626-2634.

[23] M. Ilic, L. Xie, and J.-Y. Joo, Nov 2011, "Efficient coordination of wind power and price-responsive demand part i: Theoretical foundations," Power Systems, IEEE Transactions on, vol. 26, no. 4, pp. 1875-1884.

[24] A. Marcet and T. J. Sargent, 1989, "Convergence of least-squares learning in environments with hidden state variables and private information," Journal of Political Economy, vol. 97, no. 6, pp. pp. 1306-1322. [Online]. Available: http://www.jstor.org/stable/1833240

II. DICTIONARY LEARNING FOR SHORT-TERM PREDICTION OF SOLAR PV PRODUCTION

Pourya Shamsi, Mahdi Marsousi, Huaiqi Xie, William Fries, Chelsea Shaffer

ABSTRACT

Prediction of power generated from renewable energy resources such as solar photo-voltaic (PV) is a crucial task for stabilization of grids with high renewable penetration levels. Short-term prediction of these resources allow for preemptive regulation of injected power fluctuations. In this paper, a new algorithm based on dictionary learning for prediction of solar power fluctuations is introduced. This algorithm is effective on systems with structural regularities. In this method, a dictionary is trained to carry various behaviors of the system. Prediction is performed by reconstructing the tail of the upcoming signal using this dictionary. After introduction of the proposed algorithm, experimental results are provided to evaluate the prediction mechanism.

1. INTRODUCTION

Incorporation of wind and solar energy resources will lead to massive injections of power fluctuations to power grids. Unlike conventional steam based generators, wind and solar PV resources are prone to sub-second dynamics which are caused by variations in wind speed or shading by clouds. Unfortunately, steam based resources are not fast enough to cope with such power fluctuations and hence, overall power grid will observe frequency or voltage perturbations.

Stochastic frequency/voltage perturbations have been observed by states with high penetrations of solar resources. In an extreme case, Germany has suffered from a countrywide frequency perturbations which has led to recalls of more than $300,000$ solar PV

inverters. In Gemrany, power fluctuations due to solar resources could shift the frequency of the grid to 50.2Hz. At this point, various resources were detecting islanding and were disconnecting from the grid which could lead to further instabilities. This problem is known as "*Germany 50.2Hz*" [1].

Commonly used short-term prediction methods include autoregressive (AR), moving average (MA), ARMA, autoregressive integral moving average (ARIMA), Kalman filters, neural networks, and fuzzy neural networks [2]. In traditional methods, coefficients are fixed and no dynamical update is available. In modern approach, coefficients can undergo continuous training (mainly for neural networks) or an optimization (for regression methods). For instance, multivariate regression in [3] is capable of optimizing a prediction coefficient B such that $y = xB$ where x is the input data and y is the output. However, this optimization is performed on a set of training data and is not updated regularly. A recent advancement in such modeling is sparse multivariate regression [4]. Another closely related research involves formation of the multivariate model on conditional random fields (graphical multivariate Gaussian) [5], [6]. One of the most recent research in this field benefits from probabilistic modeling of B [7]. In these methods, prediction coefficients are selected using an optimization problem and a set of training data.

A natural generalization to a prediction (or regression) algorithm is to learn every possible behavior of the system and store it in a dictionary. Using this dictionary, instead of finding prediction coefficients using optimization on a large set of training data which demands high computational complexity, prediction coefficients can be selected as a combination of these behavioral information. Hence, in each step of prediction, based on the last set of observed data, a few atoms of this dictionary is selected to predict upcoming signals. Such learning mechanism can adapt to environmental conditions for a given location and find structural regularities in formation of clouds.

In this research, a new method for prediction of the tail of a signal is introduced. This method is based on dictionary learning. First, dictionary learning is introduced. Later, the proposed algorithm is implemented on a dictionary learning framework. Examples are provided to demonstrate effectiveness of the proposed methods.

2. INTRODUCTION TO DICTIONARY LEARNING

Dictionary based sparse coding has been widely studied in the fields of image processing, classification, compression, and denoising [8]-[11]. In *Sparse coding*, an input image x is represented by a sparse coding matrix α using atoms of a dictionary $\mathcal{D}$ such that $x = \mathcal{D}\alpha + \zeta$ where ζ is the noise in reconstruction of x, and α is the derived from an optimization problem. Given an input image, neither the dictionary nor the representation is known. If $\mathcal{D}$ is also derived using an optimization algorithm, the problem is called *Dictionary learning*.

In general, α and $\mathcal{D}$ are calculated by solving $\{\alpha, \mathcal{D}\} = \arg\min_{\alpha,\mathcal{D}} \parallel x - \mathcal{D}\alpha \parallel_F$ s.t. $\parallel \alpha_{:j} \parallel_0 \leq K, \quad \parallel d_k \parallel_2 = 1, \quad \forall k, j$ where $\parallel A \parallel_F^2 = \mathrm{Tr}(AA^H)$ is the norm in the sense of Frobenius (or Hilbert-Schmidt on Hilbert spaces) for a given matrix A and $\parallel A \parallel_0 = \sum 1_{a_{ij} \neq 0}$ is the sparsity measure (counting the number of non-zero elements in A). Dictionary atom $d_k = d_{:k}$ is the k-th column of dictionary $\mathcal{D}$ and $\alpha_{:j}$ is the j-th column of coding matrix α. K is the maximum sparsity of interest. Normalization of dictionary atoms increases the performance of coding process. The above optimization problem is not convex. A good approach in solving such problems is an alternating approach where once $\mathcal{D}$ is optimized with α constant and then α is optimized with $\mathcal{D}$ constant. This method is known as *Coordinate Descent*.

Assuming a dictionary is available, a representation of the input signal can be constructed by a pursuit algorithm. The main objective of such algorithm is to find the most accurate representation of the signal with constraints on the number of used dictionary atoms. Various coding algorithms are available such as basis pursuit [12], Matching Pursuit

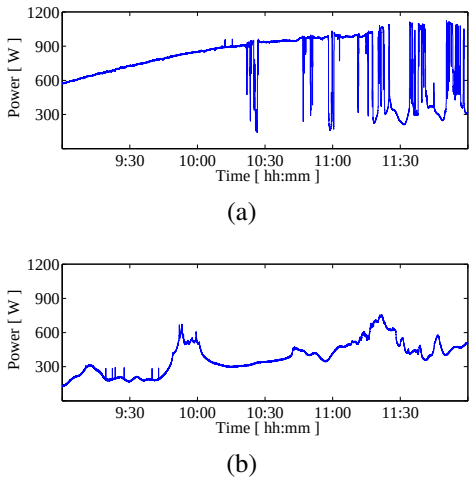


Figure 1. (a) Generated power at the zip-code 65409 on Oct. 18th, (b) generated power at the same location on Oct. 23rd, 2014.

(MP), and Orthogonal MP (OMP) [8]-[10]. Selection and update of the dictionary is another aspect of dictionary learning methods. Many algorithms have been considered for training of a dictionary. K-SVD (K Singular Value Decompositions) is an effective algorithm for training dictionaries after generation of a representation matrix [8], [13], [14].

The proposed prediction method is not limited to a specific coding or training algorithm and other algorithms such as a recently developed adaptive learning for simultaneous selection of the size, representation error, and the sparsity level of a dictionary (DLENE) can be used [11]. Recently, dictionary learning has been used for identification of the load and missing links in a communication network [15], [16]. This work is an extension of estimating missing patches of an image, to the graph of a communication network.

3. DICTIONARY LEARNING FOR PREDICTION OF RECURRENT TIME SERIES

It should be reminded that dictionary learning is most successful for applications with natural recurrence in signals such as image compression or denoising. Solar energy fluctuations depend on the status of clouds and the resulting shading. *Cloud specification* has

been a scientific tool for atmospheric studies since 1800. Shapes of clouds are categorized into *genera* (or types) such as Cumulus, Stratus, Cirrus, etc. Each genus is categorized into *species* which are then categorized into *varieties*. Based on weather conditions at a given time and location, probability of a sudden change in the cloud type is very small. Using this knowledge as a prior, one can observe a natural recurrence in cloud patterns for a region at a given time and hence, tune a dictionary based on these patterns. An example of this recurrence is observed in generated power from the solar resource shown in Fig. 1a and 1b. These signals are gathered from the same location but with different weather conditions. One can note that shapes and durations of transients are different between the two plots, but within each plot, these transients follow similar patterns.

The proposed method has four steps. The first step which is performed only once for each given weather condition includes formation of the dictionary. Second step involves continuous updates of the dictionary, third step is training the updated dictionary and the last step includes the prediction process.

3.1. FORMULATION OF THE DICTIONARY

This step is simple, however, it requires a large set of solar production training data at a given weather condition (two to three days of data for each weather condition). If the size of training sequence x is $n \times k$, then at least $n \times m$ data is required where $m < k$ and $x = \mathcal{D}\alpha$ where α is a $m \times k$ coding matrix.

First, collected data is clustered into a $n \times k$ matrix x such that consequent data form columns of x. Hence, $x = [x_{ij}]$, $x_{ij} = s_{(jn+i-nk-n-1)}$ where s_t is the data at time t. The newest data is s_{-1} located at x_{nk} while the oldest data in x is $v_{(-nk)}$ located at x_{11}. An over-complete candid dictionary $\hat{\mathcal{D}}$ is generated by randomly selecting $(m-1)$ columns of the training data x. Dictionary $\mathcal{D}$ is formed as $d_1 = [1/\sqrt{n} \cdots 1/\sqrt{n}]^T = [1/\sqrt{n}; \cdots; 1/\sqrt{n}]$

and

$$d_i = (\hat{d}_{i-1} - \bar{\hat{d}}_{i-1})/\| (\hat{d}_{i-1} - \bar{\hat{d}}_{i-1}) \|_2, \quad \forall_{2<i<m} \tag{1}$$

where $\bar{\hat{d}}_{i-1}$ is the mean of $\hat{d}_{i-1}$ which is the $(i-1)$-th atom of the candid dictionary $\hat{\mathcal{D}}$. This process will normalize the dictionary and eliminate the mean of each atom. Only the first atom of the dictionary, d_1 has a non-zero mean.

3.2. DICTIONARY UPDATE

The proposed prediction process will be executed continuously. Hence, the information in the training vector is constantly aging and new training samples are required. After arrival of each n samples, samples are placed in a vector $x^{new} = [s_{-n}; ...; s_{-1}] \in \mathbb{R}^{n\times 1}$. This vector replaces the oldest vector of information in x (x has k columns which corresponds to k vectors of information). Also, sparse code of x^{new} is calculated as $\alpha_{x^{new}} = \text{OMP}(\mathcal{D}, x^{new})$ and is used to update α based on new arrivals. OMP is orthogonal matching pursuit algorithm which is an iterative greedy algorithm. First, the residual vector is set to $x_r = x^{new}$. In each iteration, the atom which corresponds to the largest residual energy is found as $\arg\max_d |\mathcal{D}^T x_r|$ and the corresponding atom d is added to the dictionary of representatives, $\mathcal{D}_r$, and the new residual is calculated as $x_r = (\mathbf{I} - \mathcal{D}_r \mathcal{D}_r^+)x$. $\mathcal{D}_r^+$ is the Moore-Penrose pseudoinverse of $\mathcal{D}_r$. OMP continues for K iterations to generate the coding vector $\alpha_{x^{new}}$ with a sparsity of K. Coding vector α generated in this part is solely used for training the dictionary and not for prediction.

3.3. DICTIONARY TRAINING

In this section, dictionary training based on K-SVD is introduced [8]. Other training methods can be incorporated as well. Training of the dictionary is independent of the prediction routine. Based on experimental prediction results for solar production, training

the dictionary at one minute intervals provides accurate results. First, α is readily available from the pursuit algorithm as it was mentioned previously (for instance, using OMP). K-SVD training starts with selection of an atom d_j which is column j of $\mathcal{D}$. The inner product $\mathcal{D}\alpha$ will lead to multiplication of d_j to row $\alpha_{j:}$. Hence, if d_j and the corresponding row $\alpha_{j:}$ are eliminated, error in representation will be

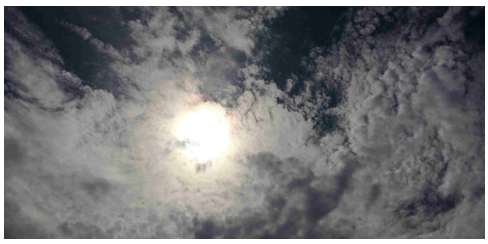
$$\| E_j \|_F = \| x - \sum_{i \neq j} d_i \alpha_{i:} \|_F \tag{2}$$

hence, by solving $\arg\min_{\{d_j^{new}, \alpha_{j:}^{new}\}} \| E_j - d_j \alpha_{j:} \|_F$ the trained dictionary atom is acquired. It should be noted that this solution does not guarantee original sparsity of α. Hence, to ensure accurate results, this solution should only be acquired under a constraint that if $\alpha_{jh} = 0$ then $\alpha_{jh}^{new} = 0$. Intuitively, if an index did not have a value prior to this update, same index should not acquire any value after the update. Therefore, unnecessary indices are eliminated from E_j to form a corrected $E_j^{corrected}$. Under this assumption, the solution is derived using Singular Value Decomposition (SVD) of the error term E_j such that $E_j^{corrected} = U\Sigma V^T$ is the single largest SVD of $E_j^{corrected}$ and U generates the update of d_j while ΣV is the update to the sparse row vector $\alpha_{j:}$.

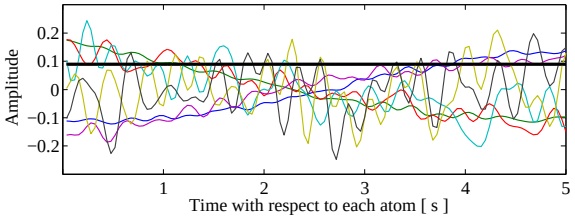
3.4. PROPOSED PREDICTION ALGORITHM

The prediction routine is a separate process such that $\mathcal{D}_T = T\mathcal{D} = [\mathbf{I}_{n_p \times n_p} \ \mathbf{0}_{n_p \times (n-n_p)}]\mathcal{D}$ is a truncated dictionary of the size $n_p \times m$ by ignoring the last $n - n_p$ rows ($n_p < n$). $\mathbf{I}$ and $\mathbf{0}$ denote identity and zero matrices, respectively. Similarly, the complement to this truncation is $\mathcal{D}_{\bar{T}} = \bar{T}\mathcal{D} = [\mathbf{0}_{(n-n_p) \times n_p} \ \mathbf{1}_{(n-n_p) \times (n-n_p)}]\mathcal{D}$. In the simplest form of prediction, after arrival of n_p new samples, x is generated as $x = [s_{-n_p}; \ldots; s_{-1}]$. Coder for prediction, α, is derived using

$$\alpha = \arg\min \ \{ \| x - T\mathcal{D}\alpha \|_F \text{ s.t. } \| \alpha \|_0 \leq K \} \tag{3}$$



(a)



(b)

Figure 2. (a) Cloud formations captured at the zip-code 65409 on Oct. 19th, 2013 (b) 8 mostly used dictionary atoms for prediction of generation power fluctuations.

an alternative approach is to compute α using a convex optimization

$$\alpha = \arg\min \ \{\| x - T\mathcal{D}\alpha \|_F + \lambda_1 \| \alpha \|_p\} \tag{4}$$

where $p > 0$. In fact $p = 1$ can lead to better robustness against outliers and $p = 2$ leads to better boundedness. $p > 0$ does not guarantee sparsity.

Solution to the first optimization can be calculated as $\alpha = \text{OMP}(T\mathcal{D}, x)$. Due to fewer number of rows, $\mathcal{D}_T$ is much more complete than $\mathcal{D}$. Therefore, this algorithm is effective if the prediction ratio $r = (n - n_p)/n$ is a small number. In fact, experimental results show a high accuracy when prediction ratio $(n - n_p)/n < 5\%$. This prediction algorithm lies upon a proposed lemma that

Lemma 1. If a given signal can be described using certain atoms of a trained dictionary, the expected value of the remainder of that signal is the remainder of those select atoms.

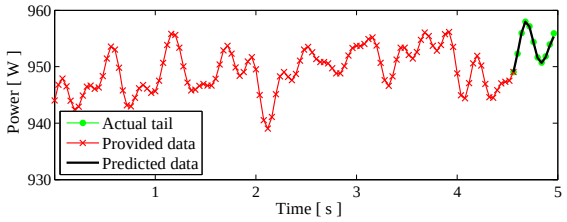
Therefore, $\mathsf{E}[y|\alpha, \mathcal{D}] = \mathcal{D}_{\bar{T}}\alpha = \bar{T}\mathcal{D}\alpha$ to generate the tail of those select dictionary atoms. Hence, y is the output prediction vector of length $(n-n_p)$. The strength of this method compared to many of the existing methods is that this method is capable of predicting a vector of the tail of a signal with multiple entries in **one step**. Also, this method is updating the prediction matrix for every prediction step. Unlike multivariate methods, this update does not require complex calculation since a dictionary has already been trained. Therefore, each step requires only an update to the coding vector α.

3.5. COMPARISION WITH OTHER METHODS

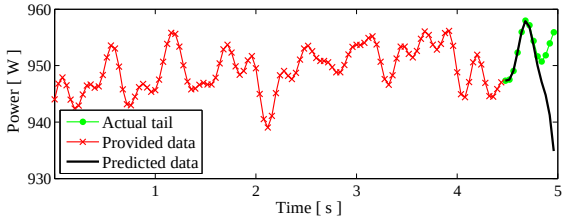
Other prediction methods which are based on an optimization are available such as [4]. In these methods, prediction coefficients are either calculated once using a set of training data or is calculated in every cycle. In the first approach, the prediction cannot provide accurate results as the system (cloud formations) undergoes continuous changes. In the second approach, the optimization is only performed on the last set of observed data which does not contain large amount of information. In the proposed approach, a large amount of information is gathered in a dictionary. In each prediction cycle, based on the last set of observed data, only a few of these behavioral vectors are selected to predict upcoming signals. Hence, this process is combining large amount of information with low computational complexity for regular updates to prediction coefficients.

4. EXPERIMENTAL RESULTS

For this test, a sampling rate of 25 samples per second (SMPS) is selected. Length of each dictionary atom is 5 seconds. Therefore, dictionary has $n = 25 \times 5 = 125$ rows. Dictionary is trained using 2 hours of data for zip-code 65409 on Oct. 19th, 2014. 600 atoms for the dictionary is considered to generate an over-complete dictionary (i.e. $n = 125 < k = 600$). Cloud formations for this location on the date of experiment are shown



(a)

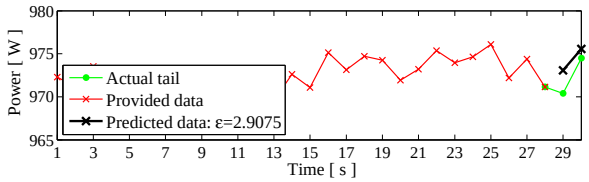


(b)

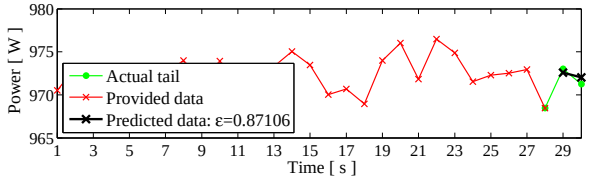
Figure 3. (a) successful prediction of 0.4 seconds of data, (b) unsuccessful prediction of 0.5 seconds of data.

in Fig. 2a. Fig. 2b illustrates 8 of the most dominant dictionary atoms for the following example. In the first example, a prediction of 0.4 seconds is made after observation of 4.6 seconds of data. This prediction is shown in Fig. 3a. It is observed that a successful multi-sample prediction of upcoming signal is generated. However, if the prediction window is increased to 0.5s, the prediction is not longer accurate for later samples. This case is shown in Fig. 3b. Although the proposed prediction method can generate a multi-sample prediction vector, a proper balance between the size of the dictionary, harmonic contents of the input signal, vector lengths, and prediction ratio is required.

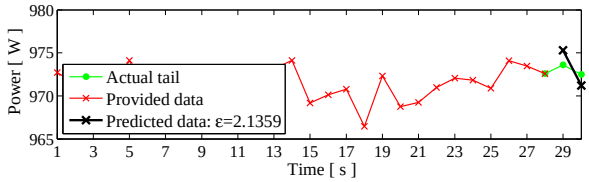
In another case study, sampling rate is 1 SMPS and the length of prediction is 2 seconds after 28 seconds of data. For this scenario, dictionary has 100 atoms. Due to the low data rate, signals are not as smooth as the previous example. However, lower computational requirements allow for longer prediction window of 2 seconds. A measure for prediction error is defined as the norm of the residual such that $\varepsilon = \| x_{actual} - \bar{T}\mathcal{D}\alpha \|_2$.



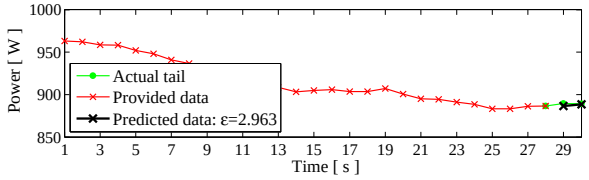
(a)



(b)



(c)



(d)

Figure 4. Four predictions of 2 seconds after 28 seconds of observed signal.

Four examples are shown in Fig. 8, 9, 10, and 4d. For each plot, ε denotes the prediction error. For least-square prediction (performed on the same problem), ε is 3.9, 1.4, 2.5, and 27.9, respectively. Therefore, the proposed method demonstrates better accuracy.

5. CONCLUSION

In this paper, a new prediction scheme based on dictionary learning was introduced. This algorithm is based on sparse coding technique used in image processing. By assuming structural regularities in solar production, up-coming generation levels were predicted. The proposed method was experimentally evaluated using a solar PV resource. In conclusion, this algorithm is effective for prediction of power fluctuations in solar production.

REFERENCES

[1] "Conclusion of the study on the "50.2 hz shutdown of pv systems in the low voltage distribution network"," VDE Association for Electrical, Electronic & Information Technologies, Report. [Online]. Available: http://www.vde.com/en/fnn/pages/50-2-hz-study.aspx

[2] T. Ouyang, X. Zha, and L. Qin, "A survey of wind power ramp forecasting," Energy and Power Engineering, vol. 5, no. 04, p. 368, 2013.

[3] H. Fan and Q. Song, "A sparse kernel algorithm for online time series data prediction," Expert Systems with Applications, vol. 40, no. 6, pp. 2174-2181, 2013.

[4] A. J. Rothman, E. Levina, and J. Zhu, "Sparse multivariate regression with covariance estimation," Journal of Computational and Graphical Statistics, vol. 19, no. 4, pp. 947-962, 2010.

[5] C. Sutton and A. McCallum, "An introduction to conditional random fields," arXiv preprint arXiv:1011.4088, 2010.

[6] —, "An introduction to conditional random fields," Machine Learning, vol. 4, no. 4, pp. 267-373, 2011.

[7] J. Chiquet, T. Mary-Huard, and S. Robin, âĂIJStructured regularization for conditional gaussian graphical models,âĂİ arXiv preprint arXiv:1403.6168, 2014.

[8] M. Aharon, M. Elad, and A. Bruckstein, "k-svd: An algorithm for designing overcomplete dictionaries for sparse representation," Signal Processing, IEEE Transactions on, vol. 54, no. 11, pp. 4311-4322, 2006.

[9] Y. C. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in Signals, Systems and Computers, 1993. 1993 Conference Record of The Twenty Seventh Asilomar Conference on. IEEE, 1993, pp. 40-44.

[10] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," Information Theory, IEEE Transactions on, vol. 53, no. 12, pp. 4655-4666, 2007.

[11] M. Marsousi, K. Abhari, P. Babyn, and J. Alirezaie, "An adaptive approach to learn overcomplete dictionaries with efficient numbers of elements," Signal Processing, IEEE Transactions on, vol. 62, no. 12, pp. 3272-3283, June 2014.

[12] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," SIAM journal on scientific computing, vol. 20, no. 1, pp. 33-61, 1998.

[13] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient implementation of the k-svd algorithm using batch orthogonal matching pursuit," CS Technion, p. 40, 2008.

[14] Z. Jiang, Z. Lin, and L. S. Davis, "Learning a discriminative dictionary for sparsecoding via label consistent k-svd," in Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on. IEEE, 2011, pp. 1697-1704.

[15] P. A. Forero, K. Rajawat, and G. B. Giannakis, "Semi-supervised dictionary learning for network-wide link load prediction," in Cognitive Information Processing (CIP), 2012 3rd International Workshop on. IEEE, 2012, pp. 1-5.

[16] P. Forero, K. Rajawat, and G. Giannakis, "Prediction of partially observed dynamical processes over networks via dictionary learning," Signal Processing, IEEE Transactions on, vol. 62, no. 13, pp. 3305-3320, July 2014.

III. CLUSTERING-BASED MARKOV SWITCHING AUTO-REGRESSIVE MODEL FOR SHORT-TERM PREDICTION OF SOLAR PRODUCTION

Huaiqi Xie, Pourya Shamsi

ABSTRACT

In this paper, a new approach to short- and very short-term prediction of excursions from solar energy resources is proposed. This approach is based on a new Clustering-based Markov Switched Autoregressive (AR) Model (CMSAM) that is capable of predicting the solar production excursion patterns under various cloud formations using a set of reference feature vectors identified through clustering. This method partitions the features extracted from the training data into multiple clusters and later uses the representative features of each cluster to predict the upcoming solar production levels. This method is capable of predicting upcoming production from seconds to minutes allowing a backup mechanism to ramp up if necessary. The main contribution of this paper is the introduction of a clustering-based mechanism in combination with a Markov jump process to identify the active cluster while boosting the immunity to noise.

1. INTRODUCTION

Prevalence, of solar energy is limited by the ability of the grid to cope with solar induced excursions [1]. Such excursions can lead to local voltage fluctuations as well as a grid-level frequency instability if penetration levels exceed 25% [2]. One challenge in using back-up generators is the time needed to ramp-up. In a few seconds, fast-acting reserve generators can ramp-up and compensate for the reduced levels of solar production caused by a passing cloud. However, to achieve this without suffering a frequency excursion, a prediction mechanism that is capable of predicting few seconds of upcoming excursion

patterns is needed. Such algorithm can be used as a feed-forward controller in a frequency excursion cancellation scheme. Therefore, in this paper, a new method for predicting very short-term excursions in solar production is introduced.

There are many prediction methods used in recent years to predict the expected hourly solar production which are mainly developed for the electricity market. Some include autoregressive model [3]-[5], neural networks [6]-[8], fuzzy logic [9], [10], and support vector machines [11], [12]. Also, imaged-based support vector machines have been recently explored in [13]. Weather-based methods have been used to incorporate the weather data in the prediction of mean solar production [14]-[17]. Most of these methods are designed to predict long-term behavior of a solar farm which ranges from daily expected values to monthly averages based on the historic data observed. However, such methods cannot be used to predict the short-term time series dynamics of the solar resource. In a more dynamic aware approach, some researchers have considered the average production of the previous hour to fine-tune the prediction of the upcoming hour. A recent work has considered a switched mechanism in a neural-network-based process to improve the accuracy of day-ahead predictions [18]. [19] combines a heuristic method with a neural network to form an optimization based approached for short-term prediction of excursions and a similar approach has considered a neural-network with particle swarm optimization but for long-term forecasting of wind energy [20], [21] uses a windowed Gaussian process to handle the time-varying characteristics of a wind resource.

The term short-term prediction of solar excursions is often expressed for time-periods ranging from an hour to a day and very short-term deals with seconds to an hour. Very short-term dynamics of a solar panel depends on the on-going cloud formations above the panel. The pattern of production observed is directly related to the cloud type. However, cloud formations vary over time and the same regression model cannot predict the observed dynamics. Fortunately, there is a limited number of cloud genera and hence, an algorithm can capture the expected pattern of each genus to predict the excursion patterns. Based

on this reasoning, researchers have started to explore pattern-based methods to predict wind and solar resources. [22] trains several reference patterns and uses them to predict upcoming wind production data. Recently, a method based on dictionary-learning was introduced that generates an over-complete dictionary of patterns to be used for prediction of upcoming generation levels through a linear interpolation of its dictionary patterns using sparse coding [23]. The dictionary learning algorithm has been widely used in image processing. However, a fundamental challenge with this algorithm is that it relies on a dictionary of patterns to be used. Hence, it requires a large database of observed patterns which will increase the computational burden during the coding process. Furthermore, it is not very accurate as it does not use an underlying transfer function of the system and instead, uses a linear interpolation of patterns. To address this issue, this paper is interested in a dictionary of underlying transfer responses instead. In the discrete-time domain, this appear as a regression model. Hence, in this paper, switching between several reference regression vectors is of interest. Piece-Wise Affine (PWA) model is a candidate algorithm that offers such behavior. PWA trains multiple reference feature vectors (instead of reference patterns) to use in the prediction process.

PWA is obtained by partitioning the state and the input set into a finite number of polyhedral regions, and by considering linear/affine systems sharing the same continuous state in each region [24]. These systems can approximate nonlinear dynamics with arbitrary accuracy and they are sufficiently expressive to model a large number of physical processes. PWA model has been used for identification of unknown (and nonlinear) systems [25] as well as in control processes [26]. PWA is based on extracting features from the observed pattern that can be used for identification or estimation of the sub-models. Feature-based methods deliver more accurate results compared to pattern-based methods. Instead of representing a pattern, they represent the sub-system that generates a particular pattern and hence, are similar to transfer functions. For example, [27] abstracts features from the data to form a predictive deep Boltzmann machine for an hour-ahead prediction of wind resources.

PWA alone has a drawback. Identification of the active sub-model is a challenge. Wrong identification of the active sub-model can result in inferior predictions. This paper will use PWA as the core and enhances it with two interacting optimization objectives for selecting the active sub-model to improve robustness to noise and accuracy.

The closest work to the proposed algorithm in this paper is Markov-Switching Regression (MSR) or Markov-Switching Auto-regressive Model (MSAM). The Markov-switching model was introduced to econometrics by [28] and was applied to model nonlinear dynamics in time series through a PWA while solving the sub-mode selection process as a Markov jump chain. Examples include modeling business cycles [29], GDP growth, interest, and inflation rates [30]-[32]. In MSAM, the decision to select a reference regression vector is based on a Markov jump chain. [33] has used a MSAM scheme to predict solar irradiance by maximizing the probability of the validity of a certain feature vector, based on the last observed data as well as the last active feature vector. Hence, the jump is solely decided based on a Markov jump chain. In a more advanced solution, [34] uses a maximum likelihood approach that maximizes the probability of selecting the correct feature vector based on the recently observed data. Similarly, [35] uses a conjugate prior distribution in the form of a Dirichlet distribution for the jump probabilities and also a conditional posterior density kernel for the regime at which the observe wind time series will evolve.

In this paper, we are expanding the MSAM technique to a Clustering-based Markov-Switching Auto-regressive Model (CMSAM) for prediction of excursion patterns in a solar energy resource. First, the justification behind the utilization of a jump regression process is introduced. Later, methods for feature extraction and clustering of the training data are explored. Identification of the reference sub-models and also the proposed Markov jump-chain for robustness to noise are presented using an optimization process. Lastly, the prediction process is derived and experimental analysis are provided to evaluate the proposed algorithm and compare it with some of the conventional methods. The specific contributions of this paper include: i) introducing a new clustering-based Markov-switching

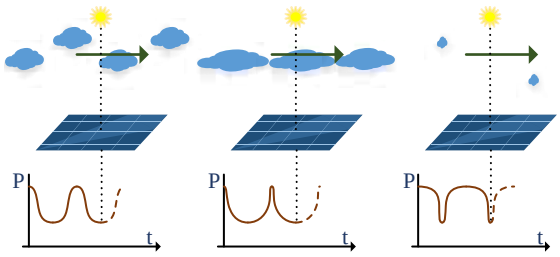


Figure 1. Different short-term dynamical response of a PV system as a function of cloud formations.

AR and proposing an optimization process to balance between the clustering objective versus the Markovian-jump objective and ii) using CMSAM for prediction of excursions in solar production.

2. CLUSTERING-BASED MARKOV-SWITCHING AUTO-REGRESSIVE MODEL FOR HYBRID PREDICTION OF SOLAR PRODUCTION

In this section, PWA in the form of a CMSAM will be used to predict the short term dynamics observed from a solar photo-voltaic (PV) resource. A solar PV system can be conceptualized as a hybrid dynamical system where different cloud genera lead to different short-term dynamical responses. For instance, Figure 1 illustrates the response of a PV system to three different cloud formations. Hence, a SAM model can be utilized to predict the behavior of such system. To do so, first, a CMSAM space has to be trained based on the empirical data observed over a variety of cloud formations for a given location. Later, a prediction scheme utilizes this trained SAM model to predict the upcoming solar production levels based on the last set of observed data.

The state space representation of CMSAM is defined as

$$x_{t+1} = \Phi_{\psi_{t+1}} \mathbf{x}_{t:t-k+1} + e_{t+1} \tag{1}$$

$$\psi_{t+1} = f(\mathbf{x}_{t:t-\tau+1}, u_t) \tag{2}$$

where $\mathbf{x}_{t:t-k+1} = [x_t \; x_{t-1} \cdots \; x_{t-k+1}]^T$ is the vector of observed data and $\Phi = [\varphi \; \varphi_{-1} \cdots \; \varphi_{-k+1}]$ is the vector of AR coefficients with a length k where φ_{-i} is the AR coefficient corresponding to the time shift $-i$. e_{t+1} is the error of predicting x_{t+1} and is equal to $x_{t+1} - \hat{x}_{t+1}$ where $\hat{x}_{t+1} = \mathsf{E}[x_{t+1}|\mathbf{x}_{t:t-k+1}] = \Phi_{\hat{\psi}_{t+1}}\mathbf{x}_{t:t-k+1}$ is the predicted value. ψ_t is the switching state and identifies the current dynamical mode of the system with its estimated value denoted as $\hat{\psi}_t$. This state is a function of some unknown inputs and x_t as $\psi_t = f(\mathbf{x}_{t:t-\tau+1}, u_t)$. $f(\cdot)$ maps its input to a countable set of $\{1, 2, \cdots, \zeta\}$ where ζ is the number of discrete states of the hybrid system. However, since the unknown inputs (i.e. movement of clouds) are not known, $\hat{\psi}_t$ is estimated using only the observed data and a model selecting mechanism in the form of $\hat{\psi}_t = \hat{f}(\mathbf{x}_{t:t-\tau+1})$. It should be noted that the standard PWA model is formed based on affine planes and hence, if utilized for time-series, is in the form of $x_{t+1} = [\Phi_{\hat{\psi}_t} \; \Phi^0_{\hat{\psi}_t}][\mathbf{x}_{t:t-k+1} \; \beta]^T + e_{t+1}$ where β is a constant number (intercept) and $\Phi^0_{\hat{\psi}_t}$ is its AR coefficient. The proposed algorithm in this paper is based on a locally Wide Sense Stationary (WSS) process with zero mean for each local vector. Hence, it is already known that the expected value of β is zero. Additionally, selection of the number of discrete states, ζ, is an optimization problem. Assuming that the length of the AR vector, k, is known, (1) defines a hybrid Infinite Impulse Response (h-IIR) discrete time system. For each ψ_k, the hybrid model is reduced to a single sub-model. In this method, first, the prediction algorithm needs to train each individual sub-model j, Φ_j, using the available empirical data. Later, a method to identify and estimate the active sub-model is needed. This method acts as the function $\hat{f}(\cdot)$. Lastly, fine tuning of $\hat{f}(\cdot)$ is performed by introducing a Markov jump chain to this estimator.

2.1. TRAINING THE PROPOSED SAM

To achieve a successful AR model, the underlying data has to be WSS. However, solar PV production data is not WSS. First, the mean of this data is a function of time and varies throughout the day. Additionally, the second moment of this data depends on the

peak production possible and also varies throughout the day. Hence, the largest variations are observed during the mid-day hours when the production can peak. These variations are smaller in early morning or evening. Hence, the solar production level, x_t, is modeled as $x_t = g(t) + h(t)$ where $g(t)$ is a non-WSS process emulating the daily variations in the production data while $h(t)$ is a WSS process emulating the random excursions from the expected variations $g(t)$.

$g(t)$ is often modeled using a time varying beta distribution. Many literature is available on prediction of the long-term variations (i.e. $g(t)$). However, in this paper, the interest is to perform a short-term prediction of the solar production as a function of the most recently observed data. Hence, here a method is presented to deliver a locally WSS signal based on the short-term observed data.

A time length of τ is considered for preparing sections of data into locally WSS processes. τ is considered to be in the range of seconds to several minutes for our short-term prediction application. Additionally, τ has to be larger than the length of the AR vectors, k in (1). For instance, if the feature selection method is based on Yule-Walker, it is recommended to have $\tau > 10k$. If the observed solar production data set is $\mathbf{P} = \{x_0, x_{-1}, \cdots\}$ where x_0 represents the most recent observed data point at time t, then a local training vector with a length of τ and a time delay of j is constructed as

$$v_j = [x_{-j}, x_{-j-1}, \cdots, x_{-j-\tau+1}]^T \tag{3}$$

where $j \in \{0, 1, 2, \cdots, N - \tau\}$ where N is the number of data points available and is often in the order of millions. The total number of v_j's is $N - \tau + 1$. Figure 2 illustrates the relation between the empirical data and v_j.

If the length of v_j's are small compared to the variations in solar production (i.e. τ is in the order of minutes), then it can be assumed that the main process $g(t)$ is constant for the length of v_j. Hence, the mean of v_j is the expected mean for each individual samples

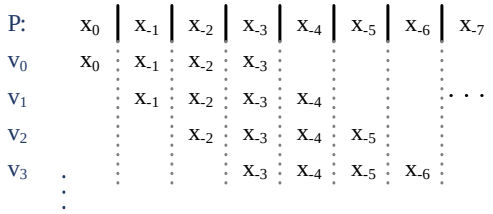


Figure 2. Training vectors v_j for $\tau = 4$.

within this vector. Also, the process covariance is independent of time and fixed for all samples within this local vector. In this case, normalized training vectors are defined as $\tilde{v}_j = (v_j - \bar{v}_j)/\sigma_{v_j}$ where $\bar{v}_j$ is the standard mean of v_j and σ_{v_j} is the biased sample variance for zero-mean data points in v_j.

For each $\tilde{v}_j$, an AR feature is extracted. using one of the available methods such as Burg, Yule-Walker, Ordinary Least Squares (OLS), etc. Some of the common feature extraction methods are explored later. However, the proposed algorithm does not depend on the particular feature extraction method used as long as the same method is used throughout the process. The regression vector (i.e. feature vector) corresponding to $\tilde{v}_j$ is Φ_{v_j}. So at this point, a set of $N - \tau + 1$ regression coefficients each with a length k is at hand. These coefficients should be clustered into ζ clusters where $C = \{\Phi_1, \Phi_2, \cdots, \Phi_\zeta\}$ is the set of centroids of each cluster. It should be noted that each centroid is in fact the reference feature vector of each cluster. The total number of clusters, ζ can be optimized using a block coordinate descent approach [36] to provide a balance between the accuracy and the computational burden. This parameter is a user-selected parameter and does not impact the proposed algorithm. Later, results with different ζs are provided. The clustering is achieved by solving an optimization in the form of

$$\min_{C,\mathcal{S}} \sum_{i=1}^{\zeta} \sum_{\Phi_{v_j} \in \mathcal{S}_i} \| \Phi_{v_j} - \Phi_i \|_2 \tag{4}$$

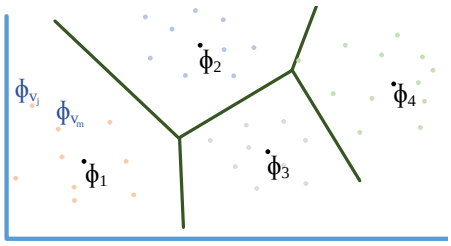


Figure 3. An example of feature vector clustering with four clusters.

$\mathcal{S} = \{\mathcal{S}_1, \cdots, \mathcal{S}_\zeta\}$ is the set of clusters where $\mathcal{S}_i$ contains the members in cluster i. If k-means approach is used, the centroid for each cluster, Φ_i is calculated as the mean of the members in that cluster such that $\Phi_i = \sum(\Phi_{v_j} | v_j \in \mathcal{S}_i)/|\mathcal{S}_i|$.

By clustering the large number of candidate feature vectors (regression coefficients) into only ζ representative centroids which are the representative regression coefficients as well, the prediction of the solar production will be performed using the hybrid auto-regression model in (1). Figure 3 illustrates the relationship between all of the extracted features, Φ_{v_i}'s, and ζ centroids to represent ζ distinct sub-systems to be used as the core of the hybrid regression model.

2.1.1. K-means Versus K-medoids. Although k-means demonstrates good results, analytically, it is not possible to prove that the mean of each cluster will follow all of the properties of the feature extraction method used. For instance, a feature extraction method might always result in a feature vector that satisfies $\| \Phi_{v_i} \| = 1$ for each $\tilde{v}_i$. However, if many features are averaged to $\Phi_j = \sum(\Phi_{v_i})/n$, then the same property cannot be guaranteed for Φ_j. To this end, using k-medoids is analytically more accurate. Unlike k-means, k-medoids does not average the members to find the centroid. It uses a single member that represents the least distance error to all the other members as the center. Now, since this particular member was directly extracted using the feature extraction method of interest, it will contain all of the properties of that method and hence, is an analytically correct feature vector for that

method. To this end, the proposed algorithm utilizes k-medoids. Unfortunately, k-medoids is much slower than k-means. Our trials have shown that both methods generate similar results and therefore, in practice, one can utilize k-means.

2.1.2. Derivation Of Regression Coefficients. The missing aspect of the above prediction algorithm is the derivation of feature vectors (regression coefficients). There are various methods to calculate autoregressive coefficient Φ for a set of data such that $x_{t+1} = \Phi\mathbf{x}$. Below, some of these methods are reviewed.

i) *Ordinary Least Squares*: OLE has been widely used for linear regression in statistical analysis. Given a set of data with a length of τ, to derive a regression coefficient of length k, the data is formulated as a multivariate regression process where the time delays are treated as different input variables to this process such that $\mathbf{x}_{t+1:-} = \varphi\mathbf{x}_{t:-} + \varphi_{-1}\mathbf{x}_{t-1:-}\cdots$ where $\mathbf{x}_{t+p:-}$ is the time shifted training data by p. To do so, the vector $v_j = [x_{-j}, x_{-j-1},\cdots, x_{-j-k+1}]^T$ is broken into sections with a length of k such that

$$Z = \begin{bmatrix} x_{-j-1} & x_{-j-2} & x_{-j-3} & \cdots \\ x_{-j-2} & x_{-j-3} & x_{-j-4} & \cdots \\ x_{-j-3} & x_{-j-4} & x_{-j-5} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{5}$$

is a $k \times k$ sample matrix and $\mathbf{z} = [x_{-j}, x_{-j-1},\cdots, x_{-j-(k-1)}]^T$ is the output vector. Now, based on multivariate OLS, $\Phi_{v_j} = \mathbf{z}Z^T(ZZ^T)^{-1}$ which is the vector of k regression coefficients to be used in the clustering process.

ii) *Yule-Walker Method*: Yule-Walker is a standard method for deriving regression coefficients. This method results in better regression coefficients. However, to derive a good feature vector, it requires a much larger data set compared to other methods. Hence, $\tau \gg k$ is a recommended condition for using this method.

iii) *Burg Method*: Burg method is based on Levinson-Durbin and is useful where not enough data is available to generate accurate regression coefficients using Yule-Walker. Burg method requires a smaller amount of data to derive a reasonably accurate feature vector. Hence, if one is interested in reducing the length of τ, Burg can replace Yule-Walker. However, the results are less accurate.

In this paper, Yule-Walker method is used as the main method of feature extraction.

2.2. SHORT-TERM PREDICTION USING SAM

The SAM model described in (1) with a total of ζ regression vectors trained in (4) is used for short-term prediction of a switched system by first finding the best representative active subsystem $i \in \{1,\cdots,\zeta\}$, and then using linear regression for prediction of upcoming data as $\Phi_i \mathbf{x}$.

First, in order to identify the best sub-model i, a vector of observed data prior to the prediction point is needed. This vector of length τ_p also has to be locally WSS. Hence, the prediction routine starts by selecting the vector $v^p = \mathbf{x}_{t:t-\tau_p+1} = [x_t, x_{t-1},\cdots, x_{t-\tau_p+1}]^T$ with its mean as $\bar{v}^p$ and its biased variance for its zero mean representation as σ_{v^p}. By defining $\hat{v}^p = (v^p - \bar{v}^p)/\sigma_{v^p}$, the candid feature vector (regression coefficient) is estimated using the same feature extraction method used for training to derive $\hat{\Phi}$. Yule-Walker, Burg, OLS, etc. methods do not result in the same structure for the feature vectors. Hence, the same method used in the training process has to be used for the prediction vector selection as well.

The best representative sub-model is selected as

$$\min_{\kappa_t} \ (\hat{\Phi} - \Phi_{\kappa_t}) W (\hat{\Phi} - \Phi_{\kappa_t})^T \tag{6}$$

$$s.t. \ \ \kappa_t \in \{1,\cdots,\zeta\}$$

where κ_t is the index of the optimal sub-model at time t based on the information available in $\hat{v}^p$. W is the weight matrix to increase the influence of coefficients corresponding to the most recent data if needed (i.e. coefficients for x_t, x_{t-1}, ...). In the simplest form, W can be a unity matrix of proper size. By selecting the best Φ_{κ_t}, the expected value of the upcoming data is estimated as

$$\hat{x}_{t+1} = \mathsf{E}[x_{t+1}|\mathbf{x}_{t:t-\tau_p+1}, C, \kappa_t] = \Phi_{\kappa_t}\mathbf{x}_{t:t-\tau_p+1} \tag{7}$$

which is the final component in developing (1) as a short-term prediction method for solar PV production. The most important assumption in deriving (7) is that the feature vector representing x_t through $x_{t-\tau_p+1}$ is the same feature vector representing x_{t+1} through $x_{t-\tau_p}$ (i.e. $\mathsf{E}[\kappa_{t+1}|\kappa_t] = \kappa_t$). This assumption is valid if the rate of switching between sub-systems is smaller than the sampling rate. In a PV prediction application, this assumption is valid. The rate of prediction is in the order of seconds while the variations in the cloud types occur within minutes to hours.

3. MARKOV CHAINS FOR ADDED NOISE IMMUNITY

In this section, a Markov jump chain is added to the optimization (6) in order to improve its immunity to noise. The feature extraction methods such as Yule-Walker and Burg have inherent immunity to noise. Additionally, Thousands of features are averaged within a cluster to select the representative feature of each cluster Φ_i. Hence, these reference feature vectors are immune to noise within each individual sample. However, the prediction process is solely based on a single observation vector $\hat{v}^p$ to derive $\hat{\Phi}$ and find the active sub-model using (6). One can notice a fundamental flaw with this process. If $\hat{v}^p$ contains large amounts of measurement noise or interference from irregular clouds, the estimation of the active sub-model will not be accurate.

In this section, we introduce a jump chain to enforce some persistence on keeping the active sub-model instead of constantly jumping between sub-models. Before the analytical modeling, the concept behind this Markov chain comes from the fact that the migration of the active cloud genus to a new genus takes a while. During the training process, the probability of migrating between each sub-model (which represents a group of cloud formations) to another sub-model is captured empirically. Using these probabilities a Markov chain is derived as

$$\mathbf{s}_t = P\mathbf{s}_{t-1} \tag{8}$$

where $\mathbf{s}_t = [\Pr(\kappa_t = 1), \cdots, \Pr(\kappa_t = \zeta)]$ is the state vector of this chain where $\Pr(\kappa_t = i)$ is the probability of the having the i-th sub-model active at time t (i.e. being at state i). $P = [p_{ij}]$ is the transition probability matrix with $p_{\kappa_{t-1}\kappa_t}$ is the probability of going from sub-model κ_{t-1} at time $t-1$ to sub-model κ_t at time t. This probabilistic vector is then added as a cost to (6) so that in addition to the distance between the estimated feature vector $\hat{\Phi}$, the probability of changing the state is also influential on the decision process. The new optimization is derived as

$$\begin{aligned} &\min_{\kappa_t} \ (\hat{\Phi} - \Phi_{\kappa_t})W(\hat{\Phi} - \Phi_{\kappa_t})^T(1 - p_{\kappa_{t-1}\kappa_t})^{\alpha} \\ &s.t. \ \ \kappa_t \in \{1, \cdots, \zeta\}, \ P \end{aligned} \tag{9}$$

where α controls the strength the Markov jump chain versus the distance cost model. $\alpha = 0$ is equivalent to eliminating the Markov jump matrix from the optimization. Considering the fact that jump probabilities between clusters are lower than 1, this paper suggests that α should be tuned between 0 and 1 and values larger than 1 should be avoided. With this addition, the SAM in (1) will evolve into a more robust Markov Switched Autoregressive

Model (CMSAM) of

$$\mathsf{E}[x_{t+1}|\mathbf{x}_{t:t-k+1},\ C,\ \kappa_{t-1},\ P] = \Phi_{\kappa_t}\mathbf{x}_{t:t-k+1} \tag{10}$$
$$\kappa_t = \arg\min[(\hat{\Phi} - \Phi_{\kappa_t})W(\hat{\Phi} - \Phi_{\kappa_t})^T(1 - p_{\kappa_{t-1}\kappa_t})^\alpha]$$

In order to evaluate the importance of adding the proposed Markov chain to the previous SAM model, an example is provided. In this example, three randomly generated stable discrete time systems are considered for analysis. All systems have a sampling time of 1s. The first system is a Finite Impulse-Response (FIR) system in the form of;

$$a_t = x_t + 0.5x_{t-1} - 0.3x_{t-3} + 0.6x_{t-6} - 0.4x_{t-10} \tag{11}$$

The other two sub-models are Imfinite Impulse-Reponse (IIR) systems as

$$b_t = -0.5b_{t-1} + 0.2b_{t-2} - 0.5x_{t-1} + x_{t-2} \tag{12}$$
$$\begin{aligned} c_t = {} & -0.5c_{t-1} + 0.4c_{t-2} + 0.03c_{t-3} - 0.12c_{t-4} \\ & + 0.04c_{t-5} + 0.01c_{t-6} + x_{t-3} - 2x_{t-4} - 0.5x_{t-6} \end{aligned} \tag{13}$$

A white noise source is connected to each system. The output of the hybrid system is selected at random with uniform jump probabilities and the jump occurs at a rate of 1 minute. A 100,000 point training data set is measured from this system and fed to the training process and clustering described in (4). τ is selected to be 200s and $k = 29$.

At this point, the system is trained and three sub-models are identified. Now, as the prediction data is coming in, the sub-model identification process, (8), is initiated with $\tau_p = 60$. Figure 4 illustrates the identified sub-models versus the actual sub-model generating the test data. As it can be observed, there is a time delay between the actual jump moment of the hybrid system, and identification of the correct active sub-model. The reason for this is the amount of time required to collect sufficient samples to generate a

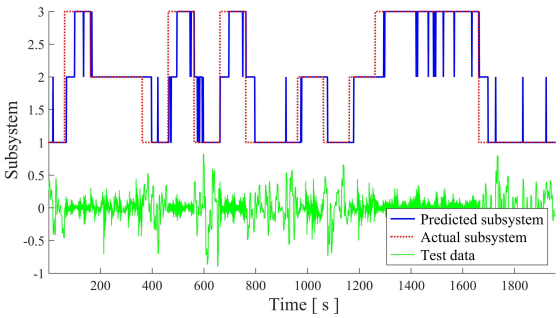


Figure 4. Identification of the correct sub-model in a hybrid system with three sub-models without the use of the proposed Markov chain.

reasonably accurate regression model of the active sub-model. Please note that τ_p is 60 seconds. Therefore, to identify the sub-model correctly, at least a large portion of this vector should be updated by the new samples from the new sub-system. This time delay is inherent to the identification process and will not pose a major issue in our application where the migration between the active sub-systems (i.e. formations of the clouds) is occurring at a much lower rate than the sampling time.

An issue observed in Figure 4 is that in some occasions, even though the system was already in the correct sub-model, for prediction sets, it mistakenly uses a wrong sub-model to predict. For instance, between 1400 to 1600 seconds, the system has mistakenly used the second sub-model instead of the correct one which is the third. This problem occurs as the sample noise influences the feature vector extraction and clustering processes. To mitigate this issue, the proposed Markov-jump-based SAM will be incorporated. Using the original 100,000 point set of training data, a Markov jump matrix, P, is trained. Later, this matrix is used in CMSAM proposed in (9) to improve the sub-model identification process and deliver more accurate results. Figure 5 illustrates the validation of CMSAM using the same set of prediction data from before. It can be observed that the wrongful jumps to other sub-models have been eliminated.

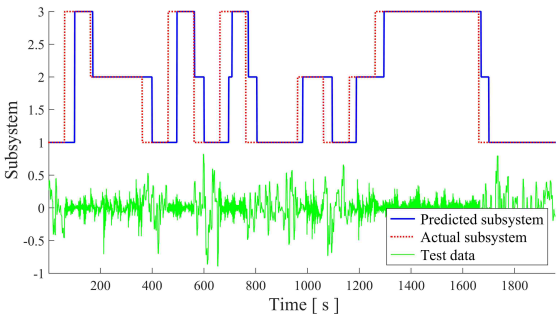


Figure 5. Identification of the correct using CMSAM.

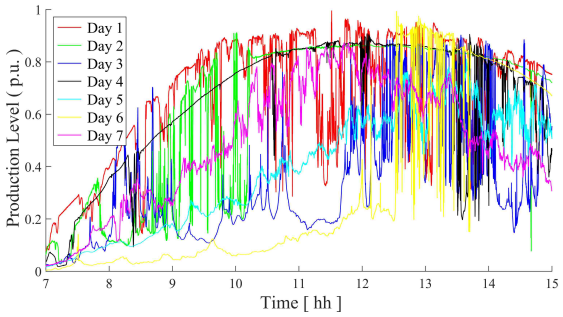


Figure 6. Set of training data containing 7 days.

4. EXPERIMENTAL RESULTS

In this section, experimental results are provided to evaluate the performance of CMSAM and compare it with some of the common prediction methods. All empirical data used in this section are collected from a solar panel installed in Missouri, USA at the zip-code of 65409 operating in October-November time frame to achieve a wide range of excursion patterns.

To show the effectiveness of this method, a multi-day example is preferred as the cloud sub-models vary between different days. Hence, a 7-day training scenario is considered in this paper. Each selected day has a different cloud pattern ranging from no cloud to very cloudy. Figure 6 depicts the collected solar production levels during these 7 days.

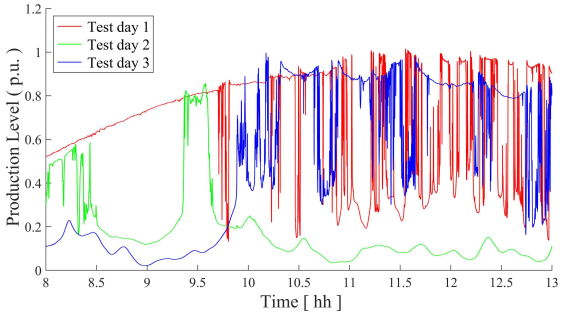


Figure 7. Set of testing data containing 3 days which have not been used for training.

At this point, a 5 cluster CMSAM is trained using the selected set of training data. The training uses $\tau = 240$ seconds to train local regression vectors of length $k = 59$ to predict 1 second of upcoming production level. Yule-Walker is selected as the feature extraction method and k-medoids with 5 replications is selected as the clustering algorithm. To test the performance of CMSAM, 3 random days not used for training have been selected. The production level collected during these days are depicted in Figure 7. To evaluate the test days, a set of observed data with the length of $\tau_p = 90$ seconds is fed to the process to predict 1 second. Figure 8 illustrates the results for *test day 1*. In this figure, at each second, the predicted data is plotted against the empirical data captured from the PV system. As expected, no large difference is observable in this macro figure. However, the main purpose of this figure is to illustrate the slow change in the reference cluster used for prediction. This figure shows that the optimization process does not constantly switch between clusters and in fact, for each time period, a particular vector of reference regression coefficients Φ_i performs better unless a sudden excursion occurs.

To observe the actual difference between the prediction and the empirical data, Figure 9 illustrates the same plot focused around 11:41am. At this time, there is a sudden excursion and this jump can evaluate the performance of CMSAM in switching between clusters to achieve a good prediction. By observing this figure one can note that cluster 3 behaves like the persistence method. This cluster will just predict the upcoming data to

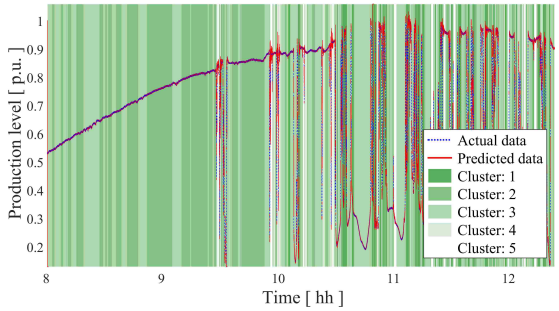


Figure 8. Prediction results for the test day 1; the background coloring illustrates the index of the reference cluster used at each moment.

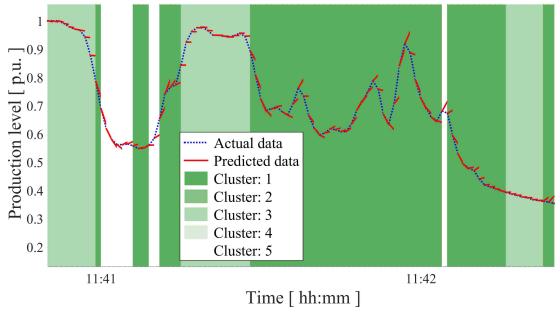


Figure 9. Prediction results for the test day 1; a zoomed-in plot for the time 11:41am to illustrate the difference between the captured data and the prediction.

be almost equal to the previously observed data. Other clusters predict different patterns. For instance, at 11:41, there is a sudden excursion of about -0.5 pu and the algorithm is predicting the upcoming data with a good accuracy.

It should be noted that the computational burden to predict one upcoming second is not significant for a modern processor. On an Intel i-7 based computer, each sample takes less than 0.01s to be processed. That provides a protective mechanism almost 1 full second to act (for instance, to ramp up a gas generator to cope with an excursion).

To compare these results with a conventional auto-regressive methods, Figure 10 depicts the results for the same minute using a Yule-Walker based auto-regressive model which is similar to CMSAM with only 1 cluster. It can be observed that the prediction is

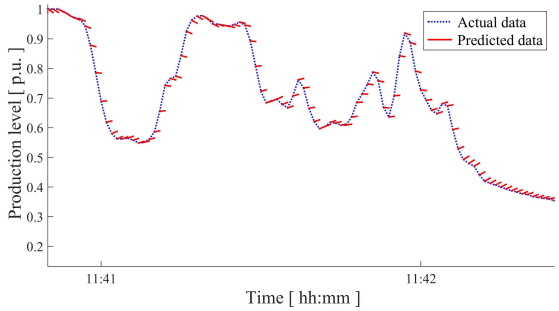


Figure 10. Prediction results for the test day 1; a zoomed-in plot for the time 11:41am predicted using a conventional auto-regressive method.

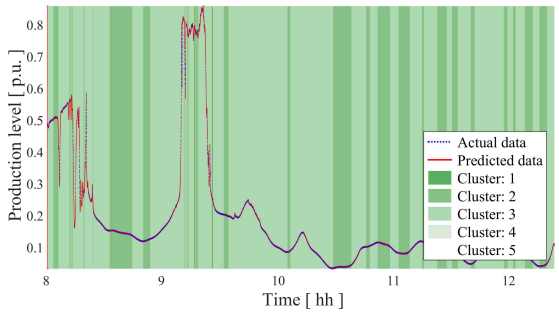


Figure 11. Prediction results for the test day 2; a different combination of clusters has been used by the optimization algorithm during this day.

very close to the persistence method and resembles a low-pass filter behavior. In particular, comparing the moment of 11:41am between Figure 9 and Figure 10 suggests that CMSAM performs better in predicting such excursions.

The same test has been performed on the *test day 2* and the results are illustrated in Figure 10. It can be observed that the utilization of clusters is different during this day. In particular, most of the day has been predicted using only 2 clusters.

In the next scenario, the number of clusters has been increased to 15. All the other parameters are kept the same as before. With a higher number of clusters, the Markov jump matrix added in (9) plays a more important role in the selection of the active cluster. To demonstrate how this matrix looks like for the 7-day training data used in this section, Figure 12 illustrates a color map of this Matrix. As expected, probabilities are dense on the

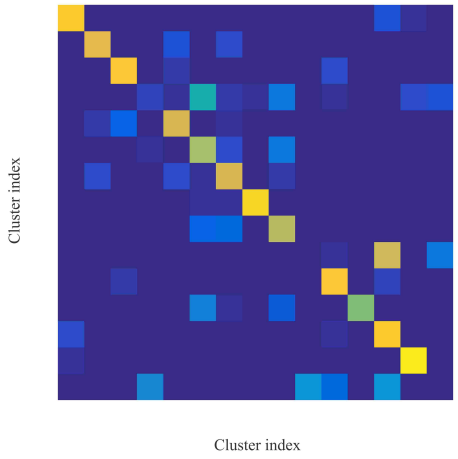


Figure 12. A visualization of the jump probability matrix for a 15 cluster CMSAM.

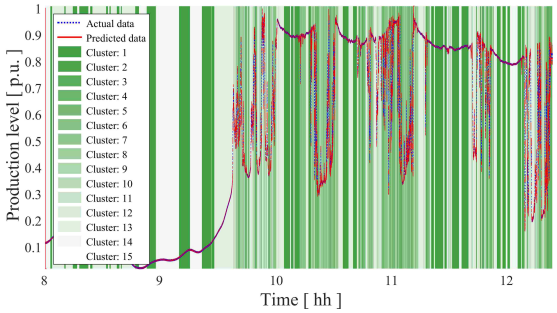


Figure 13. Prediction results for the test day 3; prediction using a 15 cluster CMSAM.

diagonal axis which shows that when the cloud formations enter a particular cluster, they tend to remain there for a while. However, there are some exceptions as observed in Figure 12 which are not intuitively expected such as clusters 4, 10, and 15 which are transitional clusters. In particular, if the system enters cluster 15, it will jump to cluster 10 with a high probability and during the next cycle, with a high probability it will jump to cluster 13. Such behaviors are not intuitive and are captured by analyzing the training data. A conventional neural network or regression algorithm cannot represent such dynamics.

Table 1. Comparison between CMSAM and conventional methods.

Method	Day 1	Day 2	Day 3	Overall
Persistence method	21%	7.7%	14%	15%
Auto-regression $k = 59$	1.33%	0.39%	1.21%	1.06%
Neural network 2 layers, input layer= 59	1.3%	0.33%	1.18%	1.03%
5 cluster CMSAM $k = 59$, $\alpha = 0$, $\tau_p = 90$	1.25%	0.31%	1.05%	0.96%
5 cluster CMSAM $k = 59$, $\alpha = 0.1$, $\tau_p = 90$	1.2%	0.30%	1.05%	0.94%
5 cluster CMSAM $k = 59$, $\alpha = 0.3$, $\tau_p = 90$	1.3%	0.30%	1.07%	0.99%
15 cluster CMSAM $k = 59$, $\alpha = 0.1$, $\tau_p = 90$	1.16%	0.29%	0.97%	0.89%

Figure 13 illustrates the prediction of *test day 3* using the 15 cluster CMSAM. With a higher number of clusters, it can be visually observed that during excursions, there are sudden changes in the clusters used. This shows that each section of the pattern is best represented with a different subsystem and hence, a single regression model (i.e. a conventional auto-regressive model) cannot predict the data effectively.

Lastly, a comparison between the CMSAM results and several of the conventional approaches are provided in Table 1. To compare the results, the average of the percentage Root Mean Square Error (%RMSE) of each predicted second is calculated over the test data. RMSE is a good measure of evaluating the accuracy of a prediction algorithm. In this table, the average of the %RMSE is calculated for each test day and also for the overall 3-day test data. The first method of interest is the common persistence method. Many technical reports suggest that for a very short-term prediction, the best approach is to just use the last observed data. Table 1 illustrates that this method, is in fact by far the least accurate method of prediction. Auto-regressive and neural networks both provide good results. However, CMSAM leads to better results. The main reason is that the conventional methods train a single predictor based on the training data. However, CMSAM trains

multiple predictors which are called the clusters. Hence, it can represent more patterns. In this table, first the 5 cluster CMSAM is analyzed under several α factors (refer to (9)). With $\alpha = 0$, the jump chain is ignored and the results are calculated solely based on the distance of the prediction feature vector and the reference feature vectors of each cluster. When α is increased to 0.1, the accuracy of the prediction improves which is due to the added robustness to noise and random variations. However, if α is increased to 0.3, the accuracy is reduced which is due to the increased inertia in keeping the previous cluster and preventing the optimizer from migrating to other clusters easily. Hence, there is an optimal α depending on the scenario. Although the 15 cluster example is more accurate, the improvement in the performance comes with the higher computational burden. Hence, adding the numbers of clusters should be weight against the added computational burden. In conclusion, for this section, the 5 cluster CMSAM with $\alpha = 0.1$ provides good results with a lower computational burden compared to the 15 cluster case.

5. CONCLUSION

This paper presented a new switched auto-regressive model for very-short time prediction of solar production data. This algorithm is suitable for predicting seconds to minutes of upcoming solar production levels. The proposed algorithm trains multiple reference regression feature vectors and will chose one based on the last set of observed data so that it can use the best representing sub-system for an ongoing cloud formation. Also, a novel Markov jump chain was added to this prediction method to improve its accuracy and reduce the impact of noise on the sub-system selection process. Lastly, various experimental results were provided to evaluate the performance of this algorithm and compare it with some of the conventional methods.

REFERENCES

[1] J. Cochran, P. Denholm, B. Speer, and M. Miller, 2015, "Grid integration and the carrying capacity of the us grid to incorporate variable renewable energy,"National Renewable Energy Laboratory (NREL), Golden, CO (United States), Tech. Rep.

[2] P. Denholm, K. Clark, and M. OâĂŹConnell, 2016, "On the path to sunshot. emerging issues and challenges in integrating high levels of solar into the electrical generation and transmission system,"National Renewable Energy Lab.(NREL), Golden, CO (United States), Tech. Rep.

[3] X.-H. Yang and Y.-Q. Li, 2011, "Dna optimization threshold autoregressive prediction model and its application in ice condition time series,"Mathematical Problems in Engineering, vol. 2012.

[4] A. Gorcin, H. Celebi, K. A. Qaraqe, and H. Arslan. IEEE, 2011, "An autoregressive approach for spectrum occupancy modeling and prediction based on synchronous measurements," in 2011 IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications, pp. 705-709.

[5] —. IEEE, 2011, "An autoregressive approach for spectrum occupancy modeling and prediction based on synchronous measurements," in 2011 IEEE 22nd International Symposium on Personal, Indoor and Mobile Radio Communications, pp. 705-709.

[6] J. Yi, Q. Wang, D. Zhao, and J. T. Wen, 2007, "Bp neural network predictionbased variable-period sampling approach for networked control systems," Applied Mathematics and Computation, vol. 185, no. 2, pp. 976-988.

[7] S. Kerdpiboon, W. L. Kerr, and S. Devahastin, 2006, "Neural network prediction of physical property changes of dried carrot as a function of fractal dimension and moisture content," Food research international, vol. 39, no. 10, pp. 1110-1118.

[8] Y. Chen, B. Yang, and J. Dong, 2006, "Time-series prediction using a local linear wavelet neural network," Neurocomputing, vol. 69, no. 4, pp. 449-465.

[9] O. Acaroglu, L. Ozdemir, and B. Asbury, 2008, “A fuzzy logic model to predict specific energy requirement for tbm performance prediction,” Tunnelling and Underground Space Technology, vol. 23, no. 5, pp. 600-608.

[10] A. Altunkaynak, M. OÂÍ zger, and M. CÂÿ akmakci, 2005, “Water consumption prediction of istanbul city by using fuzzy logic approach,” Water Resources Management, vol. 19, no. 5, pp. 641-654.

[11] J. H. Min and Y.-C. Lee, 2005, “Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters,”Expert systems with applications, vol. 28, no. 4, pp. 603-614.

[12] J. Zeng and W. Qiao, 2013, “Short-term solar power prediction using a support vector machine,” Renewable Energy, vol. 52, pp. 118-127.

[13] H. S. Jang, K. Y. Bae, H. S. Park, and D. K. Sung, July 2016, “Solar power prediction based on satellite images and support vector machine,” IEEE Transactions on Sustainable Energy, vol. 7, no. 3, pp. 1255-1263.

[14] J. R. Andrade and R. J. Bessa, 2017, “Improving renewable energy forecasting with a grid of numerical weather predictions,” IEEE Transactions on Sustainable Energy, vol. PP, no. 99, pp. 1-1.

[15] P. Mathiesen, J. M. Brown, and J. Kleissl, April 2013, “Geostrophic wind dependent probabilistic irradiance forecasts for coastal california,” IEEE Transactions on Sustainable Energy, vol. 4, no. 2, pp. 510-518.

[16] Q. Xu, D. He, N. Zhang, C. Kang, Q. Xia, J. Bai, and J. Huang, Oct 2015, “A short-term wind power forecasting approach with adjustment of numerical weather prediction input by data mining,” IEEE Transactions on Sustainable Energy, vol. 6, no. 4, pp. 1283-1291.

[17] W. P. Mahoney, K. Parks, G. Wiener, Y. Liu, W. L. Myers, J. Sun, L. D. Monache, T. Hopson, D. Johnson, and S. E. Haupt, Oct 2012, "A wind power forecasting system to optimize grid integration," IEEE Transactions on Sustainable Energy, vol. 3, no. 4, pp. 670-682.

[18] Y. Jiang, H. Long, Z. Zhang, and Z. Song, 2017, "Day-ahead prediction of bi-hourly solar radiance with a markov switch approach,"IEEE Transactions on Sustainable Energy, vol. PP, no. 99, pp. 1-1.

[19] A. Asrari, T. X. Wu, and B. Ramos, April 2017, "A hybrid algorithm for shortterm solar power prediction: Sunshine state case study," IEEE Transactions on Sustainable Energy, vol. 8, no. 2, pp. 582-591.

[20] N. Amjady, F. Keynia, and H. Zareipour, July 2011, "Wind power prediction by a new forecast engine composed of modified hybrid neural network and enhanced particle swarm optimization," IEEE Transactions on Sustainable Energy, vol. 2, no. 3, pp. 265-276.

[21] J. Yan, K. Li, E. W. Bai, J. Deng, and A. M. Foley, Jan 2016, "Hybrid probabilistic wind power forecasting using temporally local gaussian process," IEEE Transactions on Sustainable Energy, vol. 7, no. 1, pp. 87-95.

[22] Q. Hu, P. Su, D. Yu, and J. Liu, July 2014, "Pattern-based wind speed prediction based on generalized principal component analysis," IEEE Transactions on Sustainable Energy, vol. 5, no. 3, pp. 866-874.

[23] P. Shamsi, M. Marsousi, H. Xie, W. Fries, and C. Shaffer. IEEE, 2015, "Dictionary learning for short-term prediction of solar pv production," in 2015 IEEE Power & Energy Society General Meeting, pp. 1-5.

[24] E. Elhamifar, S. A. Burden, and S. S. Sastry, 2014, "Adaptive piecewiseâĂŞaffine inverse modeling of hybrid dynamical systems," IFAC Proceedings Volumes, vol. 47, no. 3, pp. 10 844-10 849.

[25] M. Vasak, D. Klanjcic, and N. Peric, Oct 2006, "Piecewise affine identification of mimo processes," in 2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, pp. 1493-1498.

[26] M. Momeni, H. M. Kelk, and H. Talebi, 2015, "Rotating switching surface control of series-resonant converter based on a piecewise affine model," IEEE Transactions on Power Electronics, vol. 30, no. 3, pp. 1762-1772.

[27] C. Y. Zhang, C. L. P. Chen, M. Gan, and L. Chen, Oct 2015, "Predictive deep boltzmann machine for multiperiod wind speed forecasting," IEEE Transactions on Sustainable Energy, vol. 6, no. 4, pp. 1416-1425.

[28] S. M. Goldfeld and R. E. Quandt, 1973, "A markov model for switching regressions," Journal of econometrics, vol. 1, no. 1, pp. 3-15.

[29] J. D. Hamilton, 1989, "A new approach to the economic analysis of nonstationary time series and the business cycle," Econometrica, vol. 57, no. 2, pp. 357-384.

[30] C.-J. Kim and C. R. Nelson, 1999, "Has the us economy become more stable? a bayesian approach based on a markov-switching model of the business cycle," The review of Economics and Statistics, vol. 81, no. 4, pp. 608-616.

[31] R. Garcia, P. Perron et al. Universite de Montreal, Departement de sciences economiques, 1991, An analysis of the real interest rate under regime shifts.

[32] M. Evans and P. Wachtel, 1993, "Inflation regimes and the sources of inflation uncertainty," Journal of Money, Credit and Banking, vol. 25, no. 3, pp. 475-511.

[33] A. Shakya, S. Michael, C. Saunders, D. Armstrong, P. Pandey, S. Chalise, and R. Tonkoski, 2017, "Solar irradiance forecasting in remote microgrids using markov switching model," IEEE Transactions on Sustainable Energy, vol. 8, no. 3, pp. 895-905.

[34] P. Pinson and H. Madsen. IEEE, 2008, “Probabilistic forecasting of wind power at the minute time-scale with markov-switching autoregressive models,” in Probabilistic Methods Applied to Power Systems, 2008. PMAPSâĂŹ08. Proceedings of the 10th International Conference on, pp. 1-8.

[35] Z. Song, Y. Jiang, and Z. Zhang, 2014, “Short-term wind speed forecasting with markov-switching model,” Applied Energy, vol. 130, pp. 103-112.

[36] Y. Xu and W. Yin, 2013, “A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion,” SIAM Journal on imaging sciences, vol. 6, no. 3, pp. 1758-1789

IV. PREEMPTIVE CONTROL: A PARADIGM IN SUPPORTING HIGH RENEWABLE PENETRATION LEVELS

Pourya Shamsi, Huaiqi Xie

ABSTRACT

This paper investigates a preemptive approach to cope with solar induced grid voltage intermittencies. To do so, this paper proposes a combination of a predictor and a voltage controller to eliminate the delays associated with conventional sample based voltage regulation. Unlike model predictive controllers, the proposed preemptive approach focuses on the input disturbances and does not neglect them as zero mean wiener processes. Additionally, due to the utilization of input predictors, the controller is no longer bound to causal response to the sampled data and can preemptively compensate for upcoming events. After introducing the proposed controller, simulation results are provided to compare the effectiveness of this approach in comparison with the existing voltage regulator schemes.

1. INTRODUCTION

Penetration levels of intermittent energy resources have been limited by the grid's capability in coping with their induced voltage and frequency excursions. Renewable energy resources such as solar systems can suffer from sudden power fluctuations as much as 100% of their production level in sub-second events. Such excursions can induce significant transients on the voltage of the distribution levels, and in the large scale, on the frequency of the power system. As a result, grid operators suffer from the intermittencies of such resources. For instance San Diego is in the front line of observing solar penetration levels of above 25%. San Diego Gas and Electric has reported large voltage excursions on their distribution feeders as shown in Figure 1. Such fluctuations are not healthy for the system

and are outside of ANSI voltage boundaries. Economic dispatch methods deal with the average expected generation from intermittent resources and do not deal with short term excursions of these energy systems [1], [2].

One approach to deal with the induced voltage and frequency excursions is to utilize a fast acting power source to compensate for the fluctuations of solar/wind resource productions. Ultra-Capacitor (UC) based active filters and battery storage systems have been investigated for mitigation of such excursions. Although battery storage systems have large capital investment and real-estate requirements, they allow for long-term mitigation of solar production fluctuations. Therefore, currently these systems are preferred over UC based filters. For instance, [3] has investigated a fleet of battery storage systems to accommodate load and solar production fluctuations.

Currently, various grid operators have started to integrate battery storage based voltage regulators with their distribution level feeders. Such systems are able to perform load profile management and energy planning as their main functionality while providing a voltage regulation scheme as an ancillary service.

Battery units are capable of regulating the voltage by controlling the reactive power in high voltage distribution networks. In lower voltage networks, due to higher resistive components of the network, voltage regulation is best achieved by compensating for the power fluctuations induced by intermittent resources. In the existing systems, a controller samples the voltage of the feeder and dispatches the battery storage system. Our aim is to improve this controller by migrating from a sample based mechanism to a preemptive strategy.

In model predictive control, first, the behavior of the system is predicted using the known model of the system and then, a suitable control input is generated accordingly. For instance, [4] has developed a model predictive method for changing transformer taps in distribution networks with high solar penetration. [5] investigates the utilization of model predictive controllers for energy management in a distribution system with various

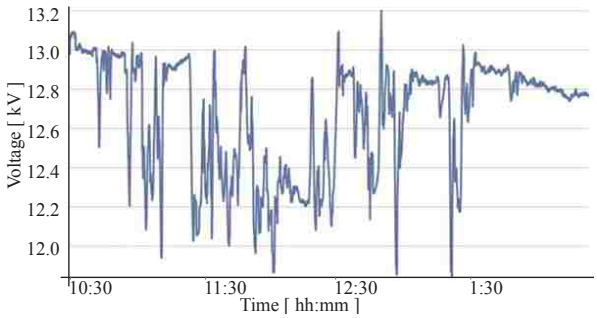


Figure 1. Line voltage of a distribution feeder with 25% penetration of solar resources.

distributed generation systems. However, model predictive methods develop a model for the plant and tend to ignore the disturbances from the inputs. For instance, in a grid voltage regulator scheme, the model predictive controller consists of a model to emulate the behavior of the distribution network. Using this model, the controller is able to calculate a (sub-) optimal dispatch command and track the desired behavior.

However, our goal is to take a step further and predict the production levels of renewable resources and preemptively inject the mismatch power even if no voltage mismatch is sensed yet. This approach is different from the traditional control approaches where an error signal is fed into the controller.

To achieve this, we need to incorporate a very short-term prediction mechanism. Neural networks have been used for long-term prediction of loads and intermittencies in power systems. Many grid operators prefer two level neural networks over other prediction methods due to the simplicity of training and utilization. However, neural networks do not perform as well as Auto Regressive (AR) models [6] and Support Vector Machines (SVM) [7] for short-term prediction of renewable resources. For instance, [8] has utilized a hybrid SVM-AR model for short-term prediction of solar excursions. In addition to traditional methods, new learning based methods have been utilized for very short term prediction of solar intermittencies as well. For instance, [9] has introduced a dictionary learning method to predict solar excursions using a linear combination of trained reference patterns using sparse coding.

Our proposed algorithm can incorporate any of the existing very short-term prediction methods. hence, we will utilize a basic AR model for this paper. This paper will first establish the overall control scheme and then will develop a test scenario to compare the proposed control scheme with a non-preemptive voltage regulation method.

2. THE PROPOSED PREEMPTIVE CONTROL SCHEME

Without the loss of generality, a model for a discrete time linear distribution network seen by a voltage regulator is in the form of

$$x(n+1) = A(n)x(n) + B(n)u(n) + \sum C_i(n)\omega_i(n) \tag{1}$$

where $x(n)$ is the vector of state variables, $u(n)$ is the vector of control inputs managed by the voltage regulator, and $\omega_i(n)$, $0 \leq i \leq h$ are the vectors of m other inputs to the system including the resources and loads. In a stochastic model predictive control (MPC) approach, it is assumed that the expected behaviors of all inputs are extracted and added to the model of the system such that

$$z(n+1) = A'(n)z(n) + B'(n)u(n) + \sum C'_i(n)\tilde{\omega}'_i(n) \tag{2}$$

and

$$\begin{aligned} \min \quad & J = \sum C(\hat{z}(n), u(n)) \\ \text{s.t.} \quad & \hat{z}(n+1) = \hat{A}'(n)\hat{z}(n) + \hat{B}'(n)u(n) \\ & \hat{z}(n) = \hat{a}'(n)\hat{z}(n) + \hat{b}'(n)\bar{z}(n) \end{aligned} \tag{3}$$

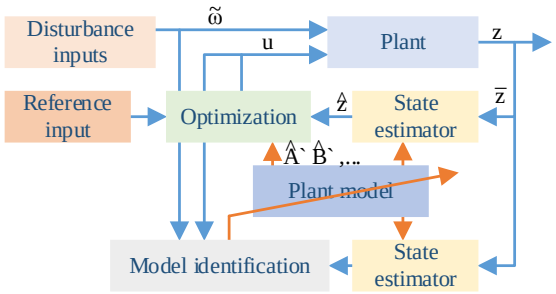


Figure 2. A generic stochastic model predictive scheme.

where $z(n) = [x(n) \; ; \; s(n)]$ is the augmented state vector which contains the additional state variables, $s(n)$, required to emulate the expected behavior of other inputs. Using this augmentation, it is assumed that auxiliary inputs, $\omega_i(n)$'s, are now reduced to zero mean, wiener processes which can then be ignored for the optimization process [10].

To cope with stochasticity in measurements, state estimators are incorporated to estimate the true state of the system $\hat{z}(n)$ based on the measured samples $\bar{z}(n)$. An example of such estimators is the Kalman filter. Also, to cope with variations of the model, adaptive methods and system identification techniques are added to this control scheme to emulate the true model of the system $A'(n)$ and $B'(n)$ using an estimated model of the system $\hat{A}'(n)$, $\hat{B}'(n)$, $\hat{a}'(n)$, and $\hat{b}'(n)$. A schematic for such a model predictive control is shown in Figure 2.

Sampled based controllers such as the MPC introduced above are bounded to the dynamics of the system. If an input varies, such controllers do not compensate for the induced variation until the disturbance goes through the dynamics of the system, reaches a sampling point, is sampled, and after the associated delays, arrives at the controller [11]. Hence, traditional sample based controllers cannot eliminate the induced excursions in a power system as the renewable energy resources are constantly inducing unknown power fluctuations to the system and the controllers can eliminate such fluctuations only after the fluctuations have passed through the system and have impacted the state variables of interest.

To cope with this problem, we are proposing a preemptive control approach. A preemptive control scheme is based on a preemptive behavior meaning that the controller will react and compensate for an induced disturbance prior to sampling and even prior to the arrival of the event. This approach is different from a predictive approach as the prediction is extended beyond the plant and is now covering the dynamics of the inputs. Hence, we are proposing a semi-non-causal structure. We are looking into the future of the induced disturbances by eliminating the traditional assumption of "zero mean wiener processes" and providing dynamic models for such stochastic inputs using a prediction mechanism.

Hence, the proposed preemptive controller is different from an MPC as it does not ignore the dynamics of auxiliary inputs, ω_i's. The main goal of the preemptive control is to ensure that the behaviors of such inputs are predicted. Hence, preemptive control is not a controller but a control scheme consisting of any suitable predictor in combination with a controller. For instance, AR models can be used to predict the behavior of each input as

$$\omega_i(n) = \sum \alpha_i(j)\omega_i(n-j) + \tilde{\omega}_i(n) \tag{4}$$

where $\alpha_i(j)$ is the j-th AR coefficient ($1 \leq j \leq m$) and $\tilde{\omega}_i(n)$ is now a true zero mean Gaussian noise. Such methods can be recursively used to predict k upcoming inputs (or methods such as [9] can be used that can predict k samples in a single prediction step). These k samples can later be used in the MPC optimization block to calculate the optimal input. For instance, if a receding horizon approach with k-samples into the future is of interest, the k-predictions of the auxiliary inputs will allow a more detailed prediction of the behavior of the system. It should be noted that in the existing MPC schemes, these inputs are considered to be zero during the receding horizon optimization. A diagram for a preemptive controller scheme over an MPC is shown in Figure 3.

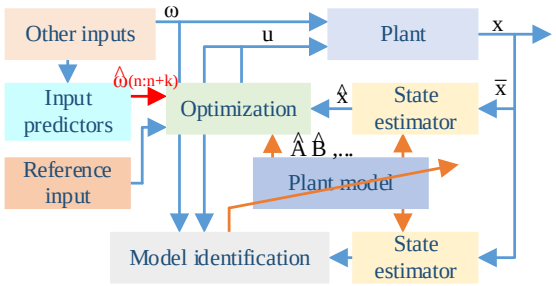


Figure 3. A preemptive-control extension of a traditional stochastic MPC.

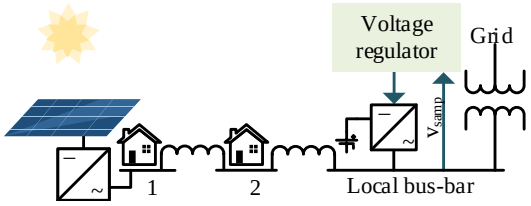


Figure 4. A conventional voltage regulation scheme in a distribution level network.

3. APPLICATION OF THE PREEMPTIVE CONTROL TO A DISTRIBUTION LEVEL FEEDER

In this section, we continue the introduction of the preemptive control using an example. Consider a low voltage distribution level network. Home level solar energy resources are interfacing such networks that are not designed to support bi-directional power flow. Additionally, such networks are weak and have large impedances. Hence, sudden variations in their power flow can induce large voltage fluctuations across such networks. Figure 1 was captured over such networks.

A distribution level network supporting a solar resource is shown in Figure 4. In such systems, the excursion in the production level of the solar resource will pass through the power converter and the distribution network. At this point, these excursions will translate to a voltage fluctuation that can be sampled by the sampling system. This data will pass through the dynamics of the sampling circuitry, communication circuitry, and lastly, the

dynamics of the controller and battery to become an injected compensation to the system. As a result, even the best controller cannot cancel the delays induced by the communication systems and the phase shifts applied by the dynamics of each component within this system.

In oppose to this mechanism, a preemptive controller will bypass the majority of the dynamics and delays mentioned above by directly sampling and predicting the solar irradiance.

The geographical area within a small distribution network is negligible compared to the area covered by cloud shading effects. Hence, a small irradiance sensor or a small milli-watt solar panel will observe the same shading dynamics as all of the solar systems installed within this distribution network (we ignore the millisecond delays between shading observations as we are considering time periods in the order of tens of milliseconds to seconds in the prediction process).

Hence, by placing a solar irradiance sensor on the controller, the controller can dynamically tune an AR predictor and predict the expected upcoming fluctuations in solar irradiance. Using this information and knowing the model of the system, the controller can anticipate the upcoming voltage/power fluctuation and inject a compensatory power to cancel such effects.

This preemptive behavior will benefit the system on the long-run as it is expected that the predictor mechanisms are tuned and are operating in a satisfactory fashion. However, the controller cannot guarantee that every fluctuation will be canceled. Also, the controller might wrongfully predict an upcoming fluctuation and preemptively inject a compensating power without an actual fluctuation ever arriving. However, such incidents will occur at a rate which is far less than the rate of excursions in a non-preemptively controlled system (otherwise, the predictor is not suitable and another prediction method should be used). Our proposed preemptive control scheme for the power system under study is shown in Figure 5.

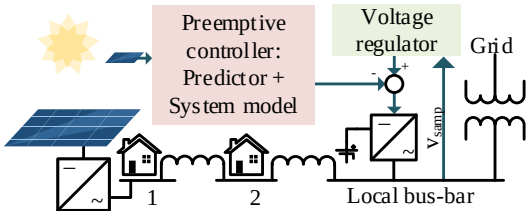


Figure 5. A preemptive voltage regulation scheme in a distribution level network.

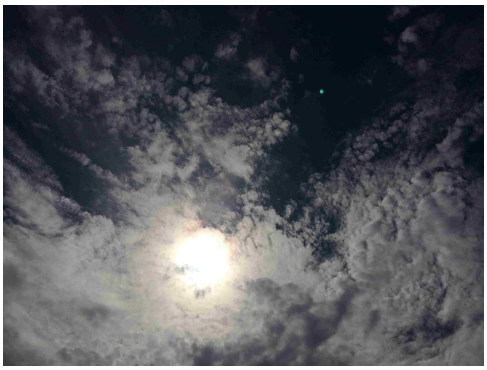


Figure 6. Cloud formations on Oct. 19th, 2013, 12:10pm, at 65409.

4. CASE STUDY

In this section, we compare a non-compensated and compensated systems of Figure 4, with the preemptively compensated system of Figure 5. To this end, we assume that houses 1 and 2 are consuming 5kW and 4kW, respectively. Also, for the solar panel, we use a real-world data captured from our 10-kW solar resource located at the zip-code of 65409. Cloud formations in this location at the time of this scenario are shown in Figure 6.

First, we need to develop a prediction scheme for the data collected from this location. As the data is arriving, we are continuously predicting 1 second into the future. To do so, we use a 29-th order AR model to predict 1 future sample. A regressive model can be developed by calculating the AR coefficients using least squares, maximum-likelihood, Yule-Walker, or Burg methods. It is shown that with a small training data-set, Burg method can develop a more accurate AR model compared to the other methods. Hence, we use Burg as the model generation and update mechanism. Based on the data collected every

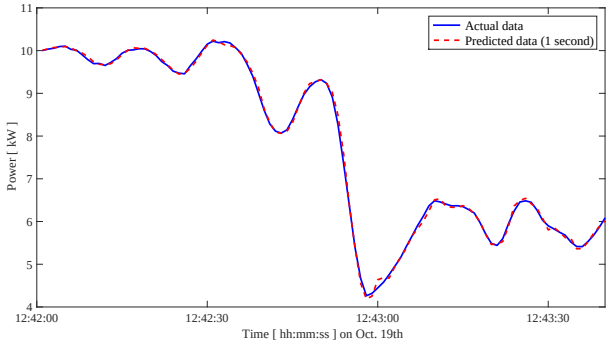


Figure 7. Comparison between the 1-second into the future predicted data and the measured data streams.

5 minutes, we run the Burg algorithm to develop a 24-coefficient AR model. Using this model, at each time step of 1 second, we can predict the upcoming solar irradiance level based on the last 24 samples captured.

Figure 7 illustrates the comparison between the actual data and the predicted data. In this figure, the AR model is used to predict the upcoming solar production level for the following second of our 10-kW solar installation. After the arrival of the new measurement, this measurement is replaced with the predicted data and the following second will be predicted. To show the effectiveness of the prediction, the stream of predicted data are plotted against the measured data. It can be observed that the AR model is performing well and is predicting the solar production level 1 second into the future.

Now that we have a prediction mechanism, we start simulating the system. First, we need to set the base scenario where the system has no compensation mechanism. In this scenario, both houses are connected with line impedances of $0.08 + \jmath 0.075\ \Omega$. Using the data collected from our solar resource, the system is simulated in Simulink + PLECS and the results are shown in Figure 8.

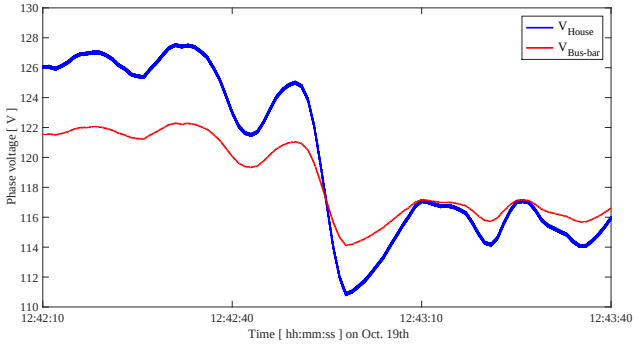


Figure 8. Simulation results for a non-compensated system suffering from power excursions shown in Figure 7.

It can be observed that the voltage of the system goes through large excursions particularly at the location of the house 1. This is expected as the solar system induces large variations in the power flow. However, these variations propagate to the location of the main bus and can impact the neighboring units.

To cope with such excursions, current system operators have started to install battery based voltage regulators (companies such as BYD manufacture container based battery systems for installation in distribution level networks). In this example, we have considered a traditional PI controller which is often used in such systems to be the voltage regulator. By sampling the voltage of the main bus, this controller dispatches a compensatory power from the battery system to regulate the main bus voltage at 120 V (in reality, this value is 12.4 kV, our intention was to use 120 V for all plots to provide the ease of comparison). Results of this scenario are shown in Figure 9.

It can be observed that not only the voltage of the main bus is improved, but also the voltage of house 1 has a lower boundary of fluctuations. However, a sample based controller cannot cancel the excursions completely. Such controllers rely on the measurements to cope with a fluctuation. Hence, the measurements have to be observed first. Also, after observation, communication time delays and battery converter dynamics prohibit an instantaneous cancellation of such excursions. Additionally, increasing the

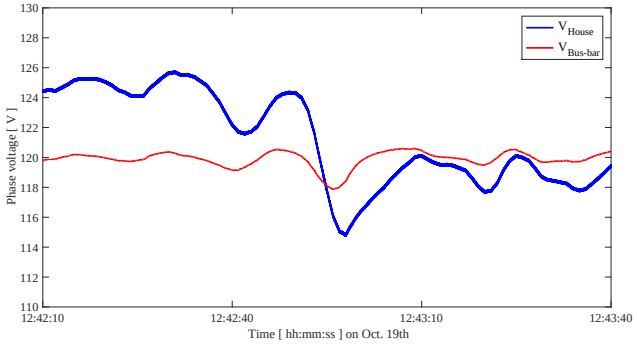


Figure 9. Simulation results for a regulated system using conventional sample based controllers.

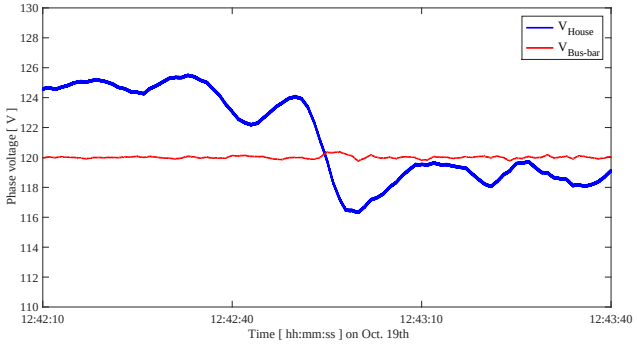


Figure 10. Simulation results for the proposed preemptive control scheme.

bandwidth of the voltage regulator can lead to sub-synchronous oscillations and instability. Our proposed preemptive controller can mitigate the above issues as we do not have to wait for a fluctuation to be observed first. By directly predicting the solar fluctuations, we preemptively calculate the required dispatch command and start to inject the compensatory power.

Such command is not causal and requires information of the future. This is the reason we have to rely on a very short-term prediction mechanism such as an AR model to predict the future fluctuations in the production level. Now that the prediction for the upcoming solar irradiance is available, we can inject a compensatory power using our battery storage system to cancel the induced solar production fluctuation at house 1. The goal is to have the two opposing power fluctuations cancel each other at the main bus bar to

minimize the effects of solar intermittencies on other loads. Figure 10 illustrates the results of our preemptive compensation. In this example we have used the model of the system to develop a deadbeat controller which utilizes the prediction information and cancels the expected error of each control period at that period.

It can be observed that the behavior of the system has improved significantly due to the non-causal information predicted by the AR model and actions that are taken preemptively. Hence, preemptive control can be a viable option to cancel the induced excursions in a system that has a high penetration level of intermittent resources.

5. CONCLUSIONS

This paper introduced a preemptive control scheme to deal with excursions in a power system with high degrees of renewable penetration. Due to the large geographical scale of a power system, various dynamics such as the dynamics of the power network will lead to large phase shifts between the arrival and the appearance of induced disturbances on the state variables of interest. Additionally, power systems often suffer from communication delays. To cope with such delays and phase shifts in a regulatory control process, we proposed the utilization of preemptive control to predict the upcoming disturbances from renewable energy resources and to act preemptively based on such information without waiting for and without observing an error signal. Simulation results were provided to show the effectiveness of such approach using real-world solar production data and system models.

REFERENCES

[1] P. Shamsi, H. Xie, A. Longe, and J. Y. Joo, "Economic dispatch for an agent-based community microgrid," IEEE Transactions on Smart Grid, vol. PP, no. 99, pp. 1-8, 2015.

[2] M. Mahmoodi, P. Shamsi, and B. Fahimi, "Economic dispatch of a hybrid microgrid with distributed energy storage," IEEE Transactions on Smart Grid, vol. 6, no. 6, pp. 2607-2614, Nov 2015.

[3] L. Wang, D. H. Liang, A. F. Crossland, P. C. Taylor, D. Jones, and N. S. Wade, "Coordination of multiple energy storage units in a low-voltage distribution network," Smart Grid, IEEE Transactions on, vol. 6, no. 6, pp. 2906-2918, 2015.

[4] V. Disfani, P. Ubiratan, and J. Kleissl, "Model predictive on-load tap changer control for high penetrations of pv using sky imager solar forecast," California Solar Initiative RD&D Program, 2015.

[5] M. Falahi, S. Lotfifard, M. Ehsani, and K. Butler-Purry, "Dynamic model predictive-based energy management of dg integrated distribution systems," Power Delivery, IEEE Transactions on, vol. 28, no. 4, pp. 2217-2227, 2013.

[6] P. Bacher, H. Madsen, and H. A. Nielsen, "Online short-term solar power forecasting," Solar Energy, vol. 83, no. 10, pp. 1772-1783, 2009.

[7] J. Zeng and W. Qiao, "Short-term solar power prediction using a support vector machine," Renewable Energy, vol. 52, pp. 118-127, 2013.

[8] M. Bouzerdoum, A. Mellit, and A. M. Pavan, "A hybrid model (sarimaâĂŞsvm) for short-term power forecasting of a small-scale grid-connected photovoltaic plant," Solar Energy, vol. 98, pp. 226-235, 2013.

[9] P. Shamsi, M. Marsousi, H. Xie, W. Fries, and C. Shaffer, "Dictionary learning for short-term prediction of solar pv production," in Power & Energy Society General Meeting, 2015 IEEE. IEEE, 2015, pp. 1-5.

[10] X. Li and P. Shamsi, "Model predictive current control of switched reluctance motors with inductance auto-calibration," IEEE Transactions on Industrial Electronics, vol. 63, no. 6, pp. 3934-3941, June 2016.

[11] —, "Inductance surface learning for model predictive current control of switched reluctance motors," IEEE Transactions on Transportation Electrification, vol. 1, no. 3, pp. 287-297, Oct 2015.

SECTION

2. SUMMARY AND CONCLUSIONS

This dissertation introduced economic dispatch and pre-emptive control in a community microgrid by using a clustering-based Markov switch approach in predicting solar power. First, a market for economic dispatch in a community microgrid was introduced, which was based on a standard auction market with passive buyers where sellers provide bids by announcing their available capacity and its linear cost model. Market was cleared by intersecting the demand and the ascending list of offers. Second, a new prediction scheme based on dictionary learning was introduced. This algorithm is based on sparse coding technique used in image processing. By assuming structural regularities in solar production, up-coming generation levels were predicted. Third, a new switched auto-regressive model was presented for very-short time prediction of solar production data. The algorithm trains multiple reference regression feature vectors and will chose one based on the last set of observed data. Also, a novel Markov jump chain was added to this prediction method to improve its accuracy and reduce the impact of noise on the sub-system selection process. Lastly, a preemptive control scheme to deal with excursions in a power system with high degrees of renewable penetration was introduced to predict the upcoming disturbances from renewable energy resources and to act preemptively based on such information without waiting for and without observing an error signal.

VITA

Huaiqi Xie was born in Wuhan, Hubei Province, China. He received his BSEE degree in 2011 from Huazhong University of Science and Technology, Wuhan, China and MSEE degree in 2013 from Missouri University of Science and Technology, Rolla, MO, USA. In August 2013, he began to pursue Ph.D degree in the department of Electrical and Computer Engineering at Missouri University of Science and Technology and he received his Ph.D in Electrical Engineering from Missouri S&T in May 2018. His research interests included energy management, smart grids, dc-dc and dc-ac converters.