Congestion Control of Smart Distribution Grids using State Estimation

Abishek S, *Non-Member, IEEE,* and Balakrishnan Narayanaswamy *Member, IEEE*

***Abstract*—Power utilities worldwide face two major challenges - peak demand and power (supply - demand) imbalance. In the midst of these difficulties faced by utilities, growing fuel costs, environmental awareness and government directives have increased the push to deploy Electric Vehicles (EVs). One single EV being charged at its peak rate imposes an instantaneous load equivalent to that of 10 average households on the grid, making it essential to schedule the EV charging in order to prevent grid failures. Our approach to this problem is motivated by parallels to the development of the internet and in particular internet protocols such as TCP, where agents respond to signals from the central authority to curtail load when the grid is congested. We show that using high resolution measurements from smart meters and distribution feeders and without measurements at any intermediate nodes, we can use recently developed semi-definite programming based state estimation techniques to accurately infer the state of the gird. We then show how to convert this line level congestion information into signal loads to users to curtail usage. In combination with smart home agents that automatically control consumption, we show how this state estimation and signaling protocol leads to reduced congestion and losses while minimizing user inconvenience.**

I. Introduction

POWER utilities worldwide face two major challenges - peak demand and power (supply - demand) imbalance. Peak demand is a period in which the demand for electrical power is at a significantly higher than average supply level. In order to satisfy large peak demand, generation and distribution companies have to make large capital expenditures in new generation stations and larger capacity lines and transformers. In addition, in free market situations this forces companies to purchase electricity on the expensive 'spot market' [1]. Satisfying peak demand requires generation companies to install and use expensive peaking power plants (that are seldom run), which in turn increases the spot prices substantially. For example, it is estimated that a 5% lowering of demand would have resulted in a 50% price reduction during the peak hours of the California electricity crisis in 2000/2001 [2]. Because of the quadratic dependence between current and electricity losses, peaks can also lead to substantial energy losses. A related problem faced by utilities is that of supply-demand imbalance. In current electricity markets "demand exhibits virtually no price responsiveness and supply faces strict production constraints and very costly storage. Extreme volatility in prices and profits will be the outcome."[3].

Abishek S is a student at the Department of Electrical Engineering, Indian Institute of Technology Madras. e-mail: ee08b055@ee.iitm.ac.com.

Balakrishnan Narayanswamy is with IBM Research Labs, India, Bangalore. e-mail: Murali.Balakrishnan@in.ibm.com.

In parallel, economic and environmental concerns have led to an increased interest in incorporating greater amounts of electricity from renewable sources into the grid [4]. New policy decisions, such as the Kyoto Protocol, have helped facilitate giant strides in this direction. Distributed generation [5], especially solar and wind power scattered across different locations, is gaining considerable importance and is being increasingly perceived as vital towards achieving carbon reduction [6]. Extracting the maximum value from a time varying and intermittent renewable resource requires intelligent optimization of generation [7], storage [8], and loads [9].

In the midst of these difficulties faced by utilities, growing fuel costs, environmental awareness and government directives have increased the push to deploy Electric Vehicles (EVs). In addition playing an important role in pushing the world towards a more sustainable form of energy consumption, Electric Vehicles (EVs), are also a form of storage. EVs can be recharged by connecting directly to the power grid. However, as pointed out by [10], one single EV being charged at its peak rate imposes an instantaneous load equivalent to that of 10 average households on the grid. Thus, it is very essential to 'schedule' the EV charging in order prevent colossal grid failures due to all EVs charging simultaneously. In addition to incorporating renewable sources, demand side storage management is crucial for enhancing efficiency [11], reducing costs and risks to the market participants [2], and increasing the stability [12] of the next generation smart grid. While the electricity generation and distribution companies reap these benefits, rational users who participate in demand side management (DSM) programs would naturally optimize their usage to minimize their own cost and thereby maximize their welfare [13]. Since consumers have to cope with volatile renewable availability and real time prices, the need for online optimization algorithms for demand management that provide utility to the user under arbitrary fluctuations in supply, load, and prices cannot be overemphasized.

Fortunately, the introduction of new communication [14] and control [15] infrastructure into the grid is expected to allow increased prosumer participation in the smart grid. This participation will be very much necessary to reduce costs. However, the in-feasibility of continuous human intervention and consumption control has led to the model of autonomous software agents, representing the consumers, that intelligently optimize and schedule energy usage [16], [17]. Our focus in this paper is on the *design of communication protocols* between distribution utilities and smart consumption agents that exploit this advanced communication and control infrastructure to allow coordination of consumption and help maintain the stability of the grid. Smart utilization of resources can greatly reduce losses in the distribution of electric power. In this regard, congestion control is an useful technique, as it can bring down losses in the grid without having to compromise on user experience. Congestion control techniques leverage the fact that electric loads can be 'scheduled' appropriately such that the loads consume power when the overall demand for power from the grid is lower. Therefore, for efficient congestion control, every load must be indifferent to the exact times in which they consume power except that a certain required amount of power gets consumed within a certain time-interval. These types of models are very common in the practice of EV charging or home backup UPS charging.

As pointed out in [10], congestion control in distribution grids share many common features to the congestion control techniques employed in computer networks, and there is much that can be learnt from successes in the area of network management. However, some of the distinguishing differences between that need to be accounted for are

- The penalty paid for a congested grid is much higher than in the internet. In the internet, congestion results in packets being dropped which can be easily retransmitted. However, in the case of electric grids, congestion may engage the circuit breaker or trip relays which are expensive to remedy and can severely disrupt power transmission.
- It is much harder in electric grids, to implicitly infer the congestion as opposed to the internet where round trip delays can be used as a metric to infer congestion.

In this paper, we show how it is possible to *infer grid congestion* with minimal expensive instrumentation and show how *TCP like algorithms can 'schedule' loads* to minimize congestion experienced by the grid.

II. Previous Work and Our Contributions

In the recent times, there have been a lot of attempt in reducing congestion and losses in the electric grid by clever scheduling of the loads. There have been works that analyze consumer behavior when the electric utilities charge tariff dynamically based on the demand. These methods to an extent have been shown to affect a rational consumer's behaviour to behave in a way that is healthy for the grid [18].

Works involving load scheduling based techniques are proposed [10] which assumes the presence of a large number of smart sensors distributed across the grid. These sensors provide useful data to the utilities and customers to aid in dynamic pricing and online optimization. Gathering credible data from the sensors however requires an ubiquitous communication network on top of the grid which may not be always available. Also in [10], the scheduling strategies requires the transmission of control packets every few milli-seconds and therefore making it very bandwidth expensive.

In this paper, we propose schemes that addresses the above concerns. We use the technique of State Estimation to infer congestion in the grid. Using state estimation helps in doing away with huge number of sensors. Thus fewer readings need to be transmitted to a control centre. Also, the control strategies proposed do not transmit control signals very frequently thus eliminating the necessity of a sophisticated communication system. The issue of fairness of the scheduler is embedded in the system model as detailed below.

III. Model Description

Traditionally, most electricity distribution grids are radial in nature [19] and hence can be represented in the form of a tree with the nodes representing the buses and the undirected edges representing the electrical line connections. This tree can further be modelled as a rooted tree with the root node representing the generator sub-station and the leaf nodes representing the loads (the customers buying electricity). This rooted tree structure is used as a model of the distribution grid in this paper.

The loads are parametrized by a *power requirement* which is the total amount of power they want to consume across some time duration called a *consumption period*. In this model, the total power requirements of the loads are assumed to be static and do not change with time i.e across different consumption periods, the total requirement of a load remains unchanged. The consumption period is further divided into equal smaller durations called *slots*. With each load, a *strategy vector* is associated which is a vector of powers consumed by the load across all slots in a consumption period. Each load is free to consume any amount of power in the individual slots as long as the total power consumed across all slots in a consumption period equals the specified requirement.

User preference is also captured by further allowing the loads to be active (consume power) only in some subset of slots during any consumption period. That is, for every load, we can further associate a binary 'indicator' vector indicating which subset of slots in any consumption period the load can consume power. An example where such a model can be applied is in the case of EV (Electric Vehicle) charging by individual customers at their homes. Some people may have to use their cars for work and can charge only during off-work hours. However, such users will be in-different as to 'how' the car gets charged during the off-work hours as long as it is charged by beginning of the work hour next day. In this example, the consumption period could be one day and the slots' duration can refer to one hour. The power requirement of the EV is specified by its storage capacity. Hence any scheduler has complete freedom in choosing an appropriate strategy vector so long as the the total power requirement is met and the user's preference (indicator) is respected.

Notation

There are assumed to be N loads in the system. Load i's total power consumption requirement is denoted by the number B_i. Consumption periods are indexed by $t = 1, 2..$, i.e if for instance the duration of a consumption period is one hour, $t = 1$ represents the first hour, $t = 2$ represents the second hour and so on. Each of these consumption periods are further divided into s slots. For example $s = 24$, if the consumption period refers to a day and if each slots represent one hour.

Load i's strategy vector during the consumption period t is denoted by $\mathbf{y_t^i}$. This is a $s \times 1$ vector. The preference indicator vector for load i is denoted by $\mathbf{I_i}$ which is a $s \times 1$ binary vector. The constraints that the vectors satisfy is,

$$\mathbf{I_i^T y_t^i} = B_i \quad (1)$$

$$y_t^i(j) \geq 0 \quad , 1 \leq j \leq s. \quad (2)$$

where $y_t^i(j)$ represents the j^{th} component of the vector $\mathbf{y_t^i}$. The constraint 2 states that the power consumed by a load is strictly non-negative, i.e the loads are assumed to have no capability to source power to the grid.

Objective

Given the setting of the distribution grid as specified above, an effective congestion control mechanism must in a distributed fashion choose the strategy vectors for the loads so as to minimize the sum total of losses on all links in the grid. Loss in this context refers to the energy loss as heat due to flow of current across an impedance. The total loss in the grid in a given slot is a function of the powers consumed by all loads in that particular slot.

Under, the special situation in which all the line impedances are the same and all loads have the same preference indicator vector, then the optimal strategy would be for each load to equally distribute its consumptions across the valid slots in a consumption period. However, the distribution grid is almost always never uniform and user preference even within a small community can widely vary. Under these circumstances, equally distributing the consumption across the available slots may not necessarily be the optimal choice.

IV. Algorithm

To achieve the objective specified above, each of the loads must adapt their strategy vectors based on explicit congestion signal given from the grid. Therefore there are two main components to achieving congestion control

- Strategy vector adaptation
- Generation of feedback signal representing grid congestion.

The second point is what is unique in this approach, as we use algorithms to cleverly use measurements from limited sensors to infer congestion thereby distinguishing ourselves from previous works which assumes the presence of a large number of sensors and a ubiquitous communication network capable of communicating all the measurements.

Feedback and Distributed Control

This section outlines the strategy used by the loads to adapt to congestion feedback signal. This is an instance of distributed control as each load adapts its strategy vector independently of the other loads.

The distributed control strategy that is used is similar to TCP control algorithms used in the internet. But unlike in the internet, there is no way for the loads to implicitly infer the congestion in the grid by themselves. Therefore, for the loads to adapt their strategies, the grid must indicate the state of congestion in the grid through a feedback signal. The feedback signal and the loads' strategy vector adaptation must be designed in a way so as to minimize the congestion in the grid. In the algorithm considered, the feedback signal sent to the loads is the sum of all link losses from the load to the root (substation). That is, the feedback signal sent at the end of every consumption period to a particular load is a $s \times 1$ vector denoting the sum of losses (heat losses) on the path from that particular load to the root node in all slots. It is also clear that the feedback signal given to each load is different. However, all loads identically follow the same strategy for adaptation which is described by the update equation below.

$$\mathbf{p_i^{t+1}} = \mathbf{y_i^t} - \epsilon_t * \mathbf{f_i^t} \quad (3)$$

where $\mathbf{f_i^t}$ is the feedback vector for load i in the t^{th} iteration and ϵ_t is a value to scale the feedback signal which is proportional to $\frac{1}{\sqrt{t}}$.

However, $\mathbf{p_i^{t+1}}$ may not satisfy constraint (1) and (2) . Hence, to $\mathbf{p_i^{t+1}}$, an error vector $\mathbf{e}$ must be added to get the strategy vector for the next set of slots. That is, the next strategy vector, $\mathbf{y_i^{t+1}}$ can be computed as follows.

$$\mathbf{y_i^{t+1}} = \min_{||\mathbf{e}||} \left(\mathbf{p_i^{t+1}} + \mathbf{e} \right) s.t,$$

$$\mathbf{I_i^{'}} * \mathbf{y_{t+1}^i} = B_i. \quad (4)$$

Equation 4 is choosing among all the strategy vectors that satisfy (1) and (2), the vector chosen as the strategy vector has the least norm-2 deviation from $\mathbf{p_i^{t+1}}$.

State Estimation

The following section outlines the procedure to compute the feedback signals to be sent to different loads based on very limited number of sensors using a method called 'State Estimation'. The only sensors used in the process are the power sensors measuring the total power consumed by each of the loads. This measurement data is in fact sufficient to estimate the congestion in the grid. In the present day context, the sensors for measuring power consumption by the end users are already (or soon will be) in place and hence no additional sensors need to be installed to estimate congestion.

State estimation in the context of distribution grids refers to the task of estimating all the node voltages and line currents. One way to to do this would be to simply put in sensors and meters at all buses and lines to measure the state of the system. As it is very hard to measure and transmit the voltages and currents at all points in the grid, typically only a few of the node voltages and line currents are measured. The rest of the quantities are then estimated using the fact that the circuit topology and the line impedances are known.

The estimation problem can be relaxed to a Semi Definite Programming problem as specified in [20]. In our model, the only measurements made are the power consumed by each of the loads in every slot. Using this, the state of the system can be estimated and hence the feedback vector for each of the loads can be computed.

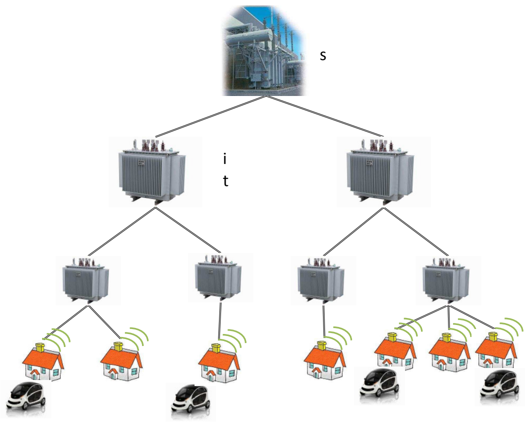

Figure 1. A schematic distribution grid with smart meters and responsive loads.

The available measurements for state estimation are the real and reactive powers consumed by every load in the system. It is enough if all the complex node voltages are estimated. The line currents can then be subsequently found out using Kirchoff's laws. Let $\mathbf{v} := [v_1, v_2, ..., v_N] \in \mathbb{C}^N$ denote the voltages at the N buses in the system. Let $\mathbf{z} = [\{P_n\}_{n\in N_d}, \{Q_n\}_{n\in N_d}]$ represent the vector of possibly noisy measurements where P_n is the real power consumed by node n and Q_n is the reactive power consumed by node n. In this setting, these correspond to the real and reactive powers consumed by the load at the leaves of the tree and the power consumed at all intermediate nodes which is zero (the nodes that are not connected to either a load or a generator consumes zero net power). That is $N_d = N \backslash \{root\}$ is the set of nodes for which we have mesurements about power consumption. Given $\mathbf{z}$, the goal of the state estimation is to estimate $\mathbf{v}$.

For every power system, a symmetric admittance matrix can be defined. The admittance matrix $\mathbf{Y}$ is an $N \times N$ matrix, where N is the number of buses in the system. The entries of this matrix is given by,

$$Y_{mn} := \begin{cases} -y_{mn} & \text{if (m,n)} \in \mathbb{E} \\ y_{nn} + \sum_{v\in N_n} y_{nv} & \text{if m=n} \\ 0, & otherwise. \end{cases} \quad (5)$$

where y_{mn} denotes the line admittance between bus m and n, and y_{nn} bus $n's$ admittance to ground. $\mathbb{E}$ denotes all pairs (i, j) such that buses i and j are connected by a transmission line.

If we define $\mathbf{i}$ as the vector of all injection currents into a bus, then using Kirchoff's law the following relation holds

$$\mathbf{i} = \mathbf{Y}\mathbf{v}. \quad (6)$$

The real and reactive powers measured at the n^{th} bus, P_n and Q_n respectively are related to the injection current and bus voltage as follows,

$$P_n + jQ_n = V_n I_n^*. \quad (7)$$

where I_n is the injection current at the n^{th} bus.

In general, z_l (the l^{th} component of vector $\mathbf{z}$) is related to the vector $\mathbf{v}$ as

$$z_l = h_l(\mathbf{v}) + e. \quad (8)$$

where h_l is the non-linear relationship specified by equation (7) and e is the measurement noise. Hence a ML estimator for $\mathbf{v}$ can be found by solving

$$\begin{aligned} \hat{\mathbf{v}} \quad &= \underset{\mathbf{v}}{\text{argmin}} \textstyle\sum_{l=1}^{L} w_l [z_l - h_l(\mathbf{v})]^2 \quad \text{s.t} \\ \mathbf{i} &= \mathbf{Y}\mathbf{v}. \end{aligned} \quad (9)$$

where w_l represents the inverse of the measurement noise variance in the sensor measuring z_l. Equation (9) can be solved using iterative Gauss-Newton methods which are widely used to solve non-linear system of equations. However, the Gauss-Newton methods are very sensitive to the initial guess and also show some convergence issues as pointed out in [20]. Hence to mitigate these effects, we adopt a convex optimization framework to solve this estimation problem.

SDP Formulation

Define the vector $\mathbf{x} := [Re^T(\mathbf{v}), Im^T(\mathbf{v})]$. If this vector is known, then the entire state of the grid is known. Thus, the state estimation procedure needs to solve for $\mathbf{x}$. Define the matrix $\mathbf{X}$ which is the outer product of the vector $\mathbf{x}$, i.e $\mathbf{X} := \mathbf{x}\mathbf{x}^\mathbf{T}$.

As defined in [20], define the matrices $\mathbf{Y}_n$ for $n = 1, 2..N$ as,

$$\mathbf{Y}_n = \mathbf{e}_n \mathbf{e}_n^T \mathbf{Y}. \quad (10)$$

where $\mathbf{e_n}$ defines the n^{th} canonical basis for $\mathbb{R}^{2N}$. Also define real matrices $\mathbf{H_{P,n}}$ as,

$$\mathbf{H}_{P,n} := \frac{1}{2} \begin{bmatrix} Re(\mathbf{Y}_n + \mathbf{Y}_n^T) & Im(\mathbf{Y}_n^T - \mathbf{Y}_n) \\ Im(\mathbf{Y}_n - \mathbf{Y}_n^T) & Re(\mathbf{Y}_n + \mathbf{Y}_n^T) \end{bmatrix} \quad (11)$$

and real matrices $\mathbf{H_{Q,n}}$ as,

$$\mathbf{H}_{Q,n} := \frac{-1}{2} \begin{bmatrix} Im(\mathbf{Y}_n + \mathbf{Y}_n^T) & Re(\mathbf{Y}_n - \mathbf{Y}_n^T) \\ Re(\mathbf{Y}_n^T - \mathbf{Y}_n) & Im(\mathbf{Y}_n + \mathbf{Y}_n^T) \end{bmatrix}. \quad (12)$$

From lemma 1 in [20],

$$P_n = Tr(\mathbf{H}_{P,n}\mathbf{X}) \quad (13)$$

$$Q_n = Tr(\mathbf{H}_{Q,n}\mathbf{X}) \quad (14)$$

Thus, equation (8) can now be written as

$$z_l = Tr(\mathbf{H}_l \mathbf{X}) + \epsilon \quad (15)$$

where $\mathbf{H}_l$ is specified according to equations (13) and (14).

To formulate the SDP, we use equations (13) and (14) in (8).

$$\begin{aligned} \hat{\mathbf{X}}_1 \quad &:= \underset{\mathbf{X}}{\text{arg-min}} \textstyle\sum_{l=1}^{L} w_l [z_l - Tr(\mathbf{H_l X})]^2, \quad s.t \\ &\mathbf{X} \succeq 0, \\ &rank(\mathbf{X}) = 1. \end{aligned} \quad (16)$$

where w_l are the reciprocal of the variance of measurement noise. Furthermore, only the relative values of w_l matter and not the absolute values. However, equation (16) is still not a convex optimization problem because of the rank constraint and the cost function has a degree 4.

The above problem can be converted into a convex optimization problem by dropping the rank constraint and by using the Schur's lemma as pointed out by [20], by introducing an auxiliary vector $\alpha \in \mathbb{R}^L$. The formulation of the state estimation problem is now in the standard SDP form,

$$\begin{aligned}\{\hat{\mathbf{X}}, \hat{\boldsymbol{\alpha}}\} \quad &:= \underset{\mathbf{X}}{\operatorname{argmin}}\{\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{\alpha}\} \quad s.t \\ &\mathbf{X} \succeq 0, \\ &\begin{bmatrix} -\alpha_l & \mathbf{z}_l - Tr(\mathbf{H}_l\mathbf{X}) \\ \mathbf{z}_l - Tr(\mathbf{H}_l\mathbf{X}) & -1 \end{bmatrix} \preceq 0, \forall l. \end{aligned} \tag{17}$$

The above can be solved as it is in the standard SDP formulation. If the solution to the above problem yields a rank-1 matrix for $\mathbf{X}$, then that solution is exact, i.e it is also a solution to the non-convex problem with the rank constraint. The vector $\mathbf{x}$ from the matrix $\mathbf{X}$, can be obtained by an eigenvalue eigenvector decomposition. The eigen-decomposition yields $\hat{\mathbf{X}} = \sum_{i=1}^{r} \lambda_i \mathbf{u}_i \mathbf{u}_i^T$ where $r = rank(\hat{\mathbf{X}})$, $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_r > 0$ are the eigenvalues, and $\{\mathbf{u}_i \in \mathbb{R}^{2N}\}_{i=1}^{r}$ are the corresponding eigenvectors. The rank-1 approximation of $\hat{\mathbf{X}}$ is $\lambda_1 \mathbf{u}_1 \mathbf{u}_1^T$. Hence the required vector $\hat{\mathbf{x}} = \sqrt{\lambda_1}\mathbf{u}_1$.

Our choice of SDP based state estimation was based on preliminary experiments with both Bayesian and conventional state estimation techniques where SDP based techniques were clearly more accurate. A more detailed evaluation of the SDP algorithm is an important direction of future work. We also note that in all our experiments the SDP algorithm found a rank 1 solution indicating that the solution is exact.

V. Simulations

This section outlines some simulations performed to show utility and convergence of the algorithm. All the simulations were performed on a 32 bus rooted tree distribution system with 16 users. The line impedances were chosen in a way such that it closely approximated the true values generally encountered. The number of slots per consumption period s was initialized to 10. Figure 2 shows that the algorithm converges rapidly, requiring only about 8 iterations to reach a near optimal solution. The strategy vectors of the loads in this case was initialized such that all of the loads chose one of the slots out of the s slots to consume its power. We see that the losses in the grid are reduced as the load strategy became more spread out over the slots, highlighting the importance of grid congestion control. The iterative algorithm has two main modules as illustrated - the control module and the state estimation module. The control module performs the update equation as specified in (3). The state estimation module is used to generate the feedback vector as required by the update (3). To perform this estimation, (17) is to be solved. The primal equation (17) is setup using the MATLAB tool SeDuMi [21]. This is converted into its dual representation using YALMIP [22] and is solved to find the optimal $\hat{\mathbf{X}}$.

In practice measurement sensors may be noisy, or may fail to report readings due to communication failures. As a result, it is important to evaluate the performance of the state estimation and signaling framework *when measurements are noisy*. Figure 3 shows the effect of noise on the performance of the algorithm. It can be seen that the algorithm converges even in the presence of substantial amounts of noise in the sensor readings. The X-axis plots the noise powers in the sensors (all sensors are assumed to have identical noise variances), and the Y-axis shows the total grid loss after the iterations have converged. The plot shows that the presence of noise increases the grid losses after convergence as compared to having ideal error free sensors. However, even in the presence of noises, it can be seen that grid congestion control can still be useful.

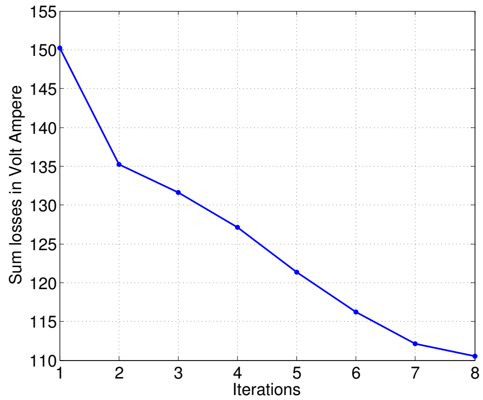

Figure 2. Reduction of grid losses and fast convergence

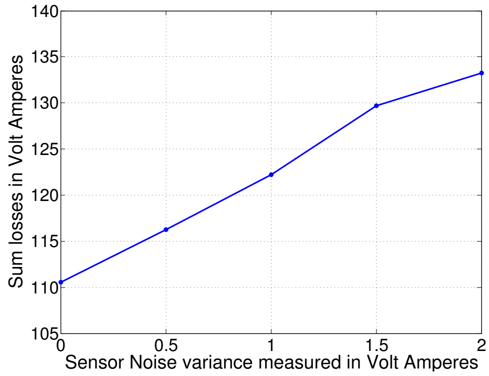

Figure 3. Congestion control is effective even with measurement noise

Figure 4 shows the effect of different choices of the learning parameter values ϵ on the performance of the algorithm. As the update (3) is a form of gradient descent, high values of ϵ tend to make the algorithm divergent. This behaviour is captured in Figure 4 as the losses in the grid blow up as the iterations proceed for large values of the learning parameter ϵ. Figure 5 also shows the divergent nature of the algorithm for large values of ϵ. In Figure 5, the X-axis plots the values of ϵ and the Y-axis plots the losses in the grid after 20 iterations. As expected, the losses are higher when ϵ value is high. This highlights that the value of the learning parameter should be chosen small enough for the algorithm to succeed.

We note that the unit of time is left undefined and that this depends on the physical parameters and goals of the system. If we are interested in maintaining stability of the system, we may need very high frequency sampling from the smart meter (order of a few kilohertz) which may be infeasible in todays

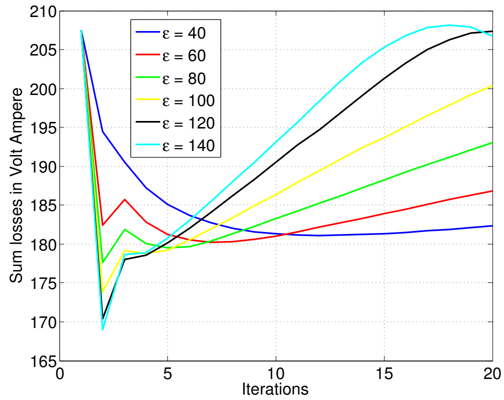

Figure 4. The effect of the value of the learning parameter ϵ on convergence. For high value of ϵ, the algorithm is seen to diverge

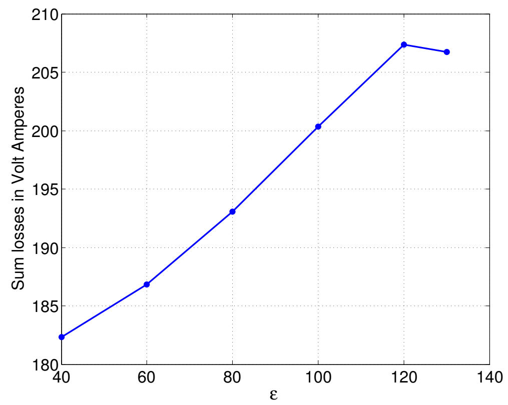

Figure 5. Grid losses after 20 iterations with different learning parameter ϵ

settings. However, for loss minimization, which is the focus of this paper one sample every few minutes is sufficient.

VI. Conclusions

In this paper we showed that using semi-definite programming based approaches to state estimation along with TCP like signalling approaches allows us to reduce grid congestion and losses using *only smart meter measurements*. While theoretically it is expected that SDP algorithms will continue to perform well when leaf nodes can supply power, a detailed study of the case in which users become (as it is expected in smart grids) "prosumers", i.e. consumers and producers at the same time of power, is left for future work. In order for the state estimation and consumption control to be effective in practice, they have to be conducted at the scale of grid phenomenon which occur at time scales of seconds or milliseconds. This requires the smart meters to send measurements at much finer temporal resolutions (which many currently deployed meters are capable of). This requires a much more efficient communication and processing infrastructure to run our algorithms which is an important direction of future work. In addition, grid congestion management is possible using currently deployed metering provided smart chargers and appliance are willing and able to respond (quickly) to the signals from the grid, which are yet to be developed.

References

[1] C. Harris, *Electricity Markets: Pricing, Structures and Economics*. Sussex, England: Wiley, 2006.

[2] International Energy Agency, "The power to choose - enhancing demand response in liberalised electricity markets findings of IEA demand response project," 2003.

[3] S. Borenstein, "The trouble with electricity markets: Understanding California's restructuring disaster," *The Journal of Economic Perspectives*, vol. 16, no. 1, pp. 191–211, 2002. [Online]. Available: http://www.ingentaconnect.com/content/aea/jep/2002/00000016/00000001/art00009

[4] R. Wiser and G. Barbose, "Renewables portfolio standards in the united states; a status report with data through 2007," Lawrence Berkeley National Laboratory, Tech. Rep. LBNL-154E, April 2008.

[5] T. Ackermann, G. Andersson, and L. Soder, "Distributed generation: a definition," *Electric Power Systems Research*, vol. 57, no. 3, pp. 195 – 204, 2001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0378779601001018

[6] G. Pepermans, J. Driesen, D. Haeseldonckx, R. Belmans, and W. D'haeseleer, "Distributed generation: definition, benefits and issues," *Energy Policy*, vol. 33, no. 6, pp. 787 – 798, 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0301421503003069

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

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

[9] A. Kowli and S. Meyn, "Supporting wind generation deployment with demand response," in *IEEE Power and Energy Society General Meeting*, july 2011, pp. 1 –8.

[10] O. Ardakanian, C. Rosenberg, and S. Keshav, "Real-time distributed congestion control for electrical vehicle charging," *Greenmetrics*, 2012.

[11] K. Spees and L. Lave, "Demand response and electricity market efficiency," *The Electricity Journal*, vol. 20, no. 3, pp. 69–85, April 2007. [Online]. Available: http://ideas.repec.org/a/eee/jelect/v20y2007i3p69-85.html

[12] F. Rahimi and A. Ipakchi, "Demand response as a market resource under the smart grid paradigm," *IEEE Transactions on Smart Grid*, vol. 1, no. 1, pp. 82 –88, june 2010.

[13] A.-H. Mohsenian-Rad and A. Leon-Garcia, "Optimal residential load control with price prediction in real-time electricity pricing environments," *IEEE Transactions on Smart Grid*, vol. 1, no. 2, pp. 120 –133, sept. 2010.

[14] V. Sood, D. Fischer, J. Eklund, and T. Brown, "Developing a communication infrastructure for the smart grid," in *Electrical Power & Energy Conference (EPEC), 2009 IEEE*. Ieee, 2009, pp. 1–7.

[15] H. Farhangi, "The path of the smart grid," *IEEE Power and Energy Magazine*, vol. 8, no. 1, pp. 18–28, 2010.

[16] P. Vytelingum, T. Voice, S. Ramchurn, A. Rogers, and N. Jennings, "Agent-based micro-storage management for the smart grid," in *AAMAS*, 2010, pp. 39–46.

[17] A. Mohsenian-Rad, V. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia, "Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid," *IEEE Transactions on Smart Grid*, vol. 1, no. 3, 2010.

[18] L. Huang, J. Walrand, and K. Ramchandran, "Optimal smart grid tariffs," in *Information Theory and Applications Workshop (ITA)*, 2012, pp. 212 –220.

[19] P. Barker and R. De Mello, "Determining the impact of distributed generation on power systems. i. radial distribution systems," in *IEEE Power Engineering Society Summer Meeting*, vol. 3, 2000, pp. 1645 –1656 vol. 3.

[20] H. Zhu and G. Giannakis, "Estimating the state of ac power systems using semidefinite programming," in *North American Power Symposium (NAPS), 2011*, aug. 2011, pp. 1 –7.

[21] J. F. Sturm, "Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones," 1998.

[22] J. Löfberg, "Yalmip : A toolbox for modeling and optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004. [Online]. Available: http://users.isy.liu.se/johanl/yalmip