Residential Electricity Load Smoothing Study

Submitted to Environmental Defense Fund

April 2016

Executive Summary

Distributed PV systems have the potential to provide significant value to utilities by contributing to Renewable Portfolio Standards and clean energy goals, reducing peak grid demand, and reducing load variability. Under current conditions, increasingly dense penetrations of distributed PV often result in the utility distribution system absorbing excess amounts of energy production during the daytime with a steep ramp-up of grid demand in the late afternoon and evening.

In collaboration with Environmental Defense Fund (EDF), Pecan Street has completed a study on three methods for residential energy load leveling and PV intermittency management that have the potential to smooth out grid impacts from homes with distributed energy resources (DERs). Smoothing of the energy demand is defined as reduced ramping rates and reduced peak demand. For the purposes of this evaluation, the smoothing methodologies implemented were evaluated for their ability to (a) reduce variability resulting from distributed PV by increasing the amount of PV used locally at the home and at the transformer level and (b) reduce the quantity of power demanded from the grid during the home's peak events and the peak demand aggregated to the transformer level. Methodologies to leverage DERs to reduce ramping rates and to serve as flexible demand resources that utilities can call upon to meet steep ramping demand warrants further study.

PV intermittency management and peak demand reduction will alleviate the grid management challenges currently posed by dense networks of distributed energy resources while enabling homes with PV to provide valuable ancillary services to utilities.

The first method analyzed by the team sought to reduce peak demand through in-home load management with the objective of enabling the home to throttle back its power consumption to stay within the locally-produced PV power capacity. The second method analyzed sought to reduce the amount of power flowing onto the grid from DERs and reduce the home's peak grid demand through integration of a residential battery system. The third method sought to reduce the grid management challenges presented by PV intermittency and reduce peak grid demand at the transformer level through real-time coordination electric use in one home with surplus PV-generated power in a neighboring home.

In summary, the evaluated methods were:

Method 1	Real-time load control management to cap the total power consumption of a single home at any one point in time through coordinated use of appliances
Method 2	Introduction of distributed energy storage systems to store energy produced on-site and to discharge the energy during periods of peak demand within the home
Method 3	Balancing demand and supply at the transformer level through real-time load control management that coordinates the use of energy-dense devices, such as electric vehicles, with PV generation from homes on a shared transformer

Methods one and two were analyzed through development of algorithms for real-time load management and implementation of these algorithms within scenario models developed using data from Pecan Street's robust dataset of home energy consumption and PV system generation profiles on 200 homes in Austin, Texas. The impact of the load management algorithms were calculated for the top 1% of peak energy values for a group of 32 homes participating in Pecan Street's research that all have PV systems of at least 4kWh and electric vehicles (EVs) with Level 2 charging systems. The top 1% of peaks was selected as the metric for evaluation because it covers the 350 highest 15-minute interval electricity use periods over the course of the analyzed year, equalling the top 87 hours of peak usage annually.

Analysis of transformer-level load balancing, Method 3, was undertaken through a real-world experiment involving a home with an EV and a second home on the same transformer with a PV system. A software and hardware device were developed and installed on an EV charging station that collected real-time data on PV production at the neighboring home and responded to that data by charging the EV when power production at the home with PV exceeded a programmed amount.

The result of this experiment was that the aggregated demand profile presented by these two homes nearly eliminated negative power flow and steep demand peaks. Additional funding would enable EDF and Pecan Street to undertake an expanded study into this methodology to validate opportunities for residential demand and generation smoothing across home types and climates.

Overall, the three residential smoothing methods enable increased local consumption of the homeowner's PV power and reduced the home's peak grid demand, thereby enabling increased levels of distributed energy resources (DER) penetration without adversely affecting the utility grid.

Residential PV Intermittency and Peak Demand Management Analysis

Background

Reduction of peak demand on the distribution grid is often a desired objective for utilities that wish to avoid purchasing expensive peak energy, and also wish to avoid the capital expense of expanding the capacity of the distribution system to handle increased power levels. Balancing distributed generation and demand would enable customer-sited PV systems to provide valuable peak demand reduction services to the utility and reduce the challenges utilities face from PV intermittency as DER penetration levels increase.

Additionally, residential PV smoothing results in reduced load variability from distributed PV systems. Load variability from DERs is problematic because it causes changes in the temperature of distribution hardware that can lead to higher rates of grid system equipment failure. These temperature changes create thermal expansion variations in the components of the distribution hardware. As these components expand and contract, cyclical mechanical stresses are introduced into the system that, over time, create fatigue failures in the system hardware.

A comparison of power and temperatures on two transformers in Austin in the summer of 2014, one on which 5 of the 8 homes served have PV arrays without smart inverters averaging 5.5 kW in size per home (transformer 7002) and the other on which 1 of 8 homes had PV (transformer 7007), reveals that the presence of a dense network of PV systems results in signifiant daily fluctuations in power and temperature, as shown in Figure 1. The analyzed transformers are the same age, model, and are located in the same neighborhood.

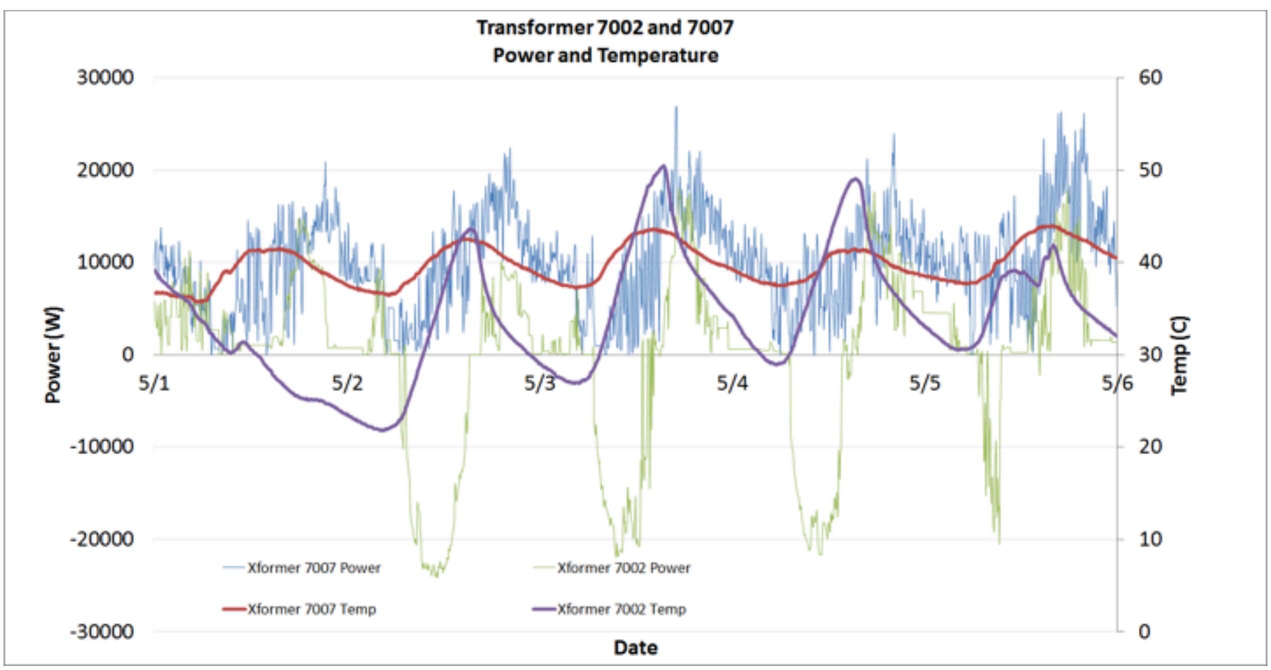

Figure 1. Comparison of power and temperature fluctuations on transformer with and without PV loading.

Through analysis of data from Pecan Street's database on residential energy use in Texas, Pecan Street and EDF identified three strategies to smooth out the residential impact on the grid by managing demand variability and PV intermittency on the customer-side of the meter. Pecan Street's evaluation documented the technical feasibility and potential benefits at a per-home level from deploying these strategies in homes with PV systems and/or electric vehicles:

1. Leveling load through coordinated charging of EV with HVAC system use so that the EV charging system ceased pulling electricity from the grid whenever the HVAC was in use to cap total home power demand.

2. Introducing energy storage at the residence to store energy produced locally by the PV system and discharge it to meet the home's power demands during the home's peak consumption periods.

3. Balancing distributed power generation and supply at the transformer level by shifting EV charging at one home to align with PV energy production at another home on the same transformer. A future study could evaluate the impact of this strategy with any electric load that has a designated circuit within the home, such as HVACs and pool pumps.

These intermittency management techniques can be implemented to consume more of the homeowner's PV power on-site and/or to balance loads between homes with and without PV systems at the aggregated transformer level, thereby enabling increased levels of distributed energy resources (DER) penetration without adversely affecting the utility grid. The three tested methodologies also showed potential for mitigating steep ramping periods at an aggregated level and providing flexible generation resources that can be called upon by utilities to meet periods of steep ramping. Additional funding would enable further evaluation of the impacts on ramping rates.

The economic benefits to the utility can be measured in terms of reduced distribution upgrade and maintenance costs, reduced transmission costs, new business models and transaction opportunities for homes with energy storage, avoided distribution outages, avoided generation capacity investments and reduced electricity generation costs (by avoiding costly generators during peak or ramping periods). The actual impact on the distribution grid would depend on the density of penetration and the locations of the homes with PV relative to the location of distribution transformers and feeders. In general, voltage and frequency control become more problematic for homes that are further from distribution substations.

Determining the optimal approach to evaluating the cost and benefits to the utility from deployment of distributed energy resources and implementation of various intermittency and smoothing strategies is a topic of great interest to the energy industry. EDF's on-going participation in developing standards for this type of cost/benefit analysis is of critical importance, and Pecan Street looks forward to supporting this investigation with additional technical feasibility analysis and ground-truth data.

Method 1: Peak Demand Reduction through Home Load Control

To understand the opportunity for PV intermittency reduction presented by load control management within the home, the research team developed and modeled an avoidance algorithm in which an EV charger was disabled whenever the home's air conditioning compressor was running. The results of this model are shown graphically in Figure 2, in which the actual grid load (blue) is compared to the modeled grid load (green). The SciLab code for this model is shown for reference in Appendix A.

To quantify the impact of implementing this type of load control, the team created a model using data on home energy use from a randomly-selected residence in Pecan Street's database that met the criteria of having an HVAC system and an EV with at least one year of whole home and circuit-level data captured by Pecan Street. The research team applied the avoidance algorithm to

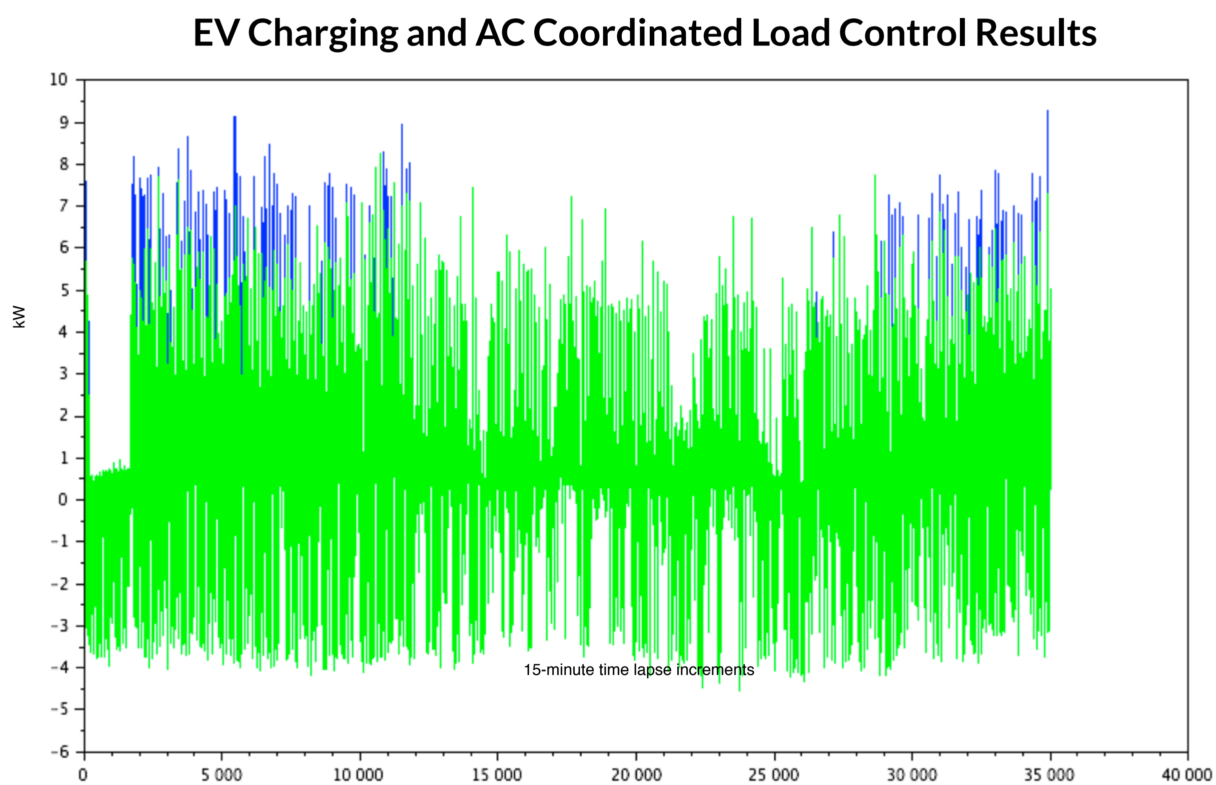

Figure 2. Load profile of home with EV charging and AC coordinated load control compared to one year actual grid load

this model, then sorted the peak values of the actual grid load signal and compared those to the sorted peak values of the modeled load level signal, as shown in Figure 3.

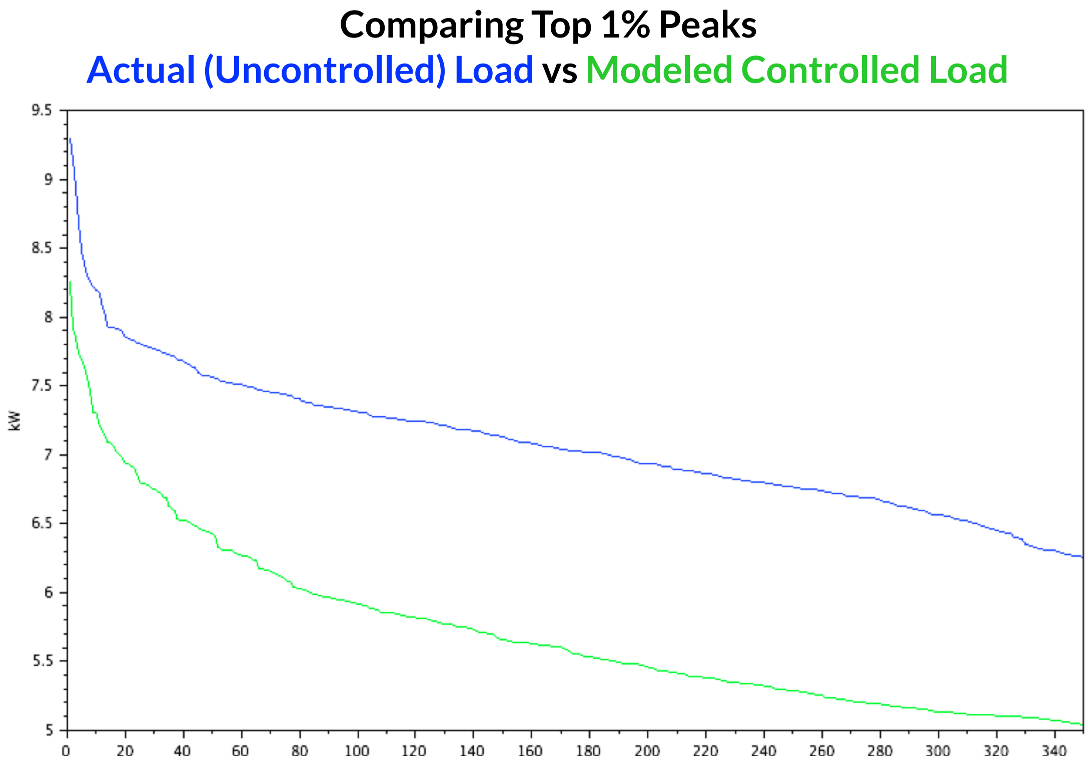

Figure 3 . Grid peak values sorted to reveal top 1% load controlled peaks compared to actual peaks

To achieve a meaningful understanding of the potential economic impact on the utility from implementation of this type of home load control technique, the team looked at the impact on the highest 1% of the actual peak load values since the highest peak values are the ones most likely to impact the distribution grid. For the home shown in Figure 3, the average peak value was reduced by 19.16%.

The model was then run on five homes for which Pecan Street had at least one year of circuit-level electricity use, whole home electricity use, and PV generation data to test that the modeling algorithms performed as intended. In addition to calculating the percent reduction in top peak values, the average values for EV charging and the primary AC compressor were recorded as EV_On and Air1_On, respectively, to serve as indicators of problems with the algorithm. These values are consistent across homes and with anticipated load levels, and therefore support the conclusion that the algorithm performed as intended to reduce peak consumption levels.

The load control algorithm resulted in reductions on the top 1% of peak values for the five homes that ranged from 0.8% to 19.2%, as shown in Table 1, with an average reduction of 9.3%. The variability in impacts could be driven by a range of factors that warrant additional investigation in a future study.

Home Data ID #	% Reduction in average value of top 1% of peaks	EV_On (Avg kW)	Air1_On (Avg kW)
26	19.2	3.4	2.5
545	8.4	3.3	2.8
1629	0.8	3.3	2.8
1642	13.3	3.4	2.5
7719	4.8	3.3	3.3

Table 1. Results for Methodology 1 implementation on sample of 5 homes

The model was then expanded to a group of 32 homes, all of which have a PV system and an electric vehicle, for which Pecan Street has detailed electricity use and generation data. The results revealed that there would be an average 3.4% reduction in the top 1% of peaks from the group of 32 homes if the home leveling controls to coordinate EV charging and HVAC use were implemented in each home.

Method 2: Intermittency Management Through On-Site Energy Storage

Residential energy storage is an effective tool to reduce load variability on the grid. Though the price and often murky permitting process for small-scale energy storage systems, defined as systems of 10 kW or less, are currently prohibitive to mass adoption, the Department of Energy and private entities are engaging in R&D efforts to overcome these barriers in the coming decade. Therefore, it is important to analyze the potential for customer-sited batteries to solve a wide range of grid challenges, including intermittency management and peak grid demand reduction.

Inclusion of battery systems in homes with PV enable the home to offer valuable ancillary services to the grid such as power factor correction, frequency regulation, demand response and rapid discharge of flexible resources during times of steep ramping on the grid. Integration of batteries also enables residential PV systems to continue producing and providing power to the home when the grid goes down, increasing community resiliency to power outages because homes with storage can serve as a valuable community resource when power would otherwise be unavailable to neighbors.

This analysis focuses on the impact of introducing a 7 kWh energy storage system into residences with PV systems to reduce the intermittency challenges these homes present to the grid. Modeling and scenario simulation were conducted using high-resolution residential data collected in Pecan Street's residential energy testbed.

Figure 4 shows an example of the energy consumption for all of the loads in a particular home (Dataport Home ID 26) for a period of 7 days starting July 1, 2014. During the week depicted, the home constantly consumed a minimum amount of energy around 1kW and frequently consumed between 1 and 4 kW, with maximum consumption reaching approximately 7.5 kW. This load profiles reflects typical consumption patterns found in homes participating in Pecan Street's research testbed in Central Texas, with HVAC use consuming a significant amount of power in the warm summer months.

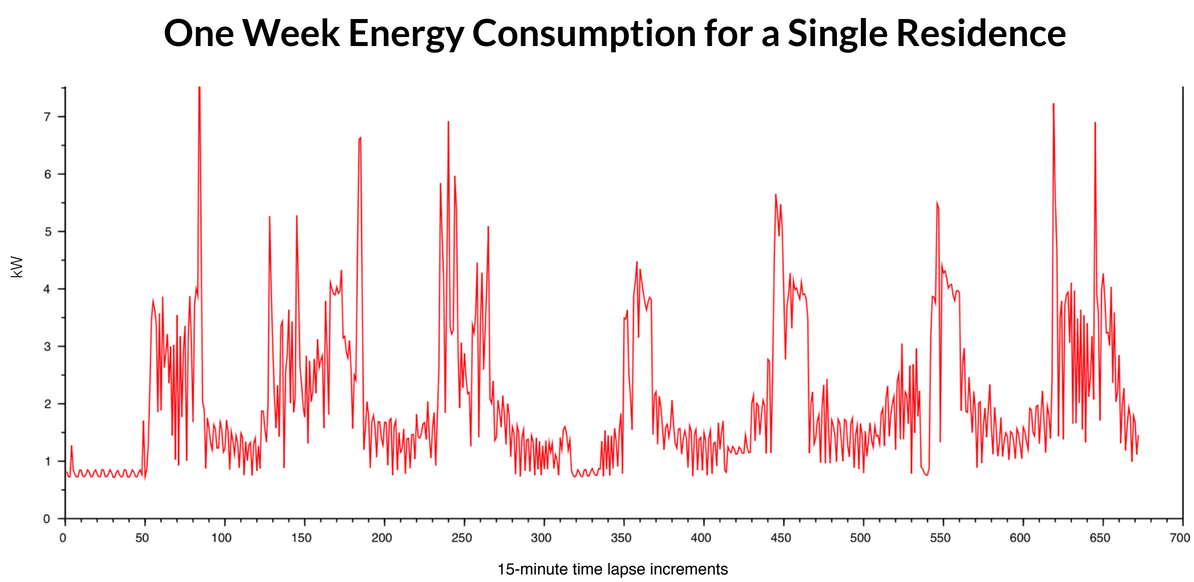

Figure 4. One week of total energy consumption at 15-minute intervals for a single home.

Figure 5 depicts one year of 15-minute interval energy consumption data for the same home. This one-year snapshot covers the time period from July 1, 2014 through June 30, 2015.

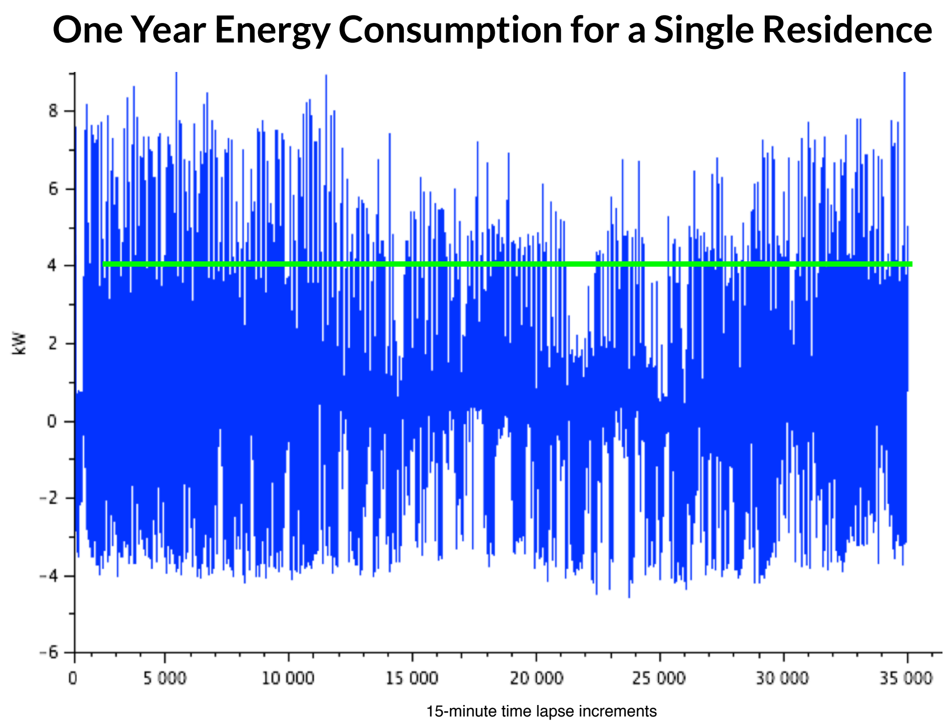

Figure 5. One year of total energy consumption at 15-minute intervals for a single home

Drawing a line through the 4kW mark, as depicted in Figure 5, shows the home frequently "peaking" above 4 kW - indicated by the green line - with the highest observed use hitting 8.5 kW.

This home has PV generation as evidenced by the negative power consumption shown at intervals throughout the year. The negative values represent times when the home's PV power is flowing back onto the grid. It can be inferred that the array is around 4kW since that appears to be the approximate peak value of power flow onto the grid.

Among the emerging applications for energy storage is the leveling of residential energy loads by locally storing PV-produced energy and then discharging it to meet the home's peak energy needs. Leveling of loads can reduce the variability of energy demand to the power distribution grid, reduce the magnitude of peak demand for the home and, when aggregated, reduce peak demand for the entire grid if the home loads are coincident with system peak.

For the initial modeling, it was assumed that the residential battery had a capacity of 7 kWh with a maximum charging rate of 6.4 kW and a maximum discharge rate of 6.0 kW. The model was programmed to simulate battery discharge whenever the residential grid load exceeded 4 kW and to charge whenever the net residential load dropped below zero (when PV generation exceeded total residential energy use).

The model evaluated the impact of these parameters on the reduction of the average value of the top 1 percent of peak values over the course of a year for the same home in the previous example (home project ID number 26). The results of the model depicting the load profile with the introduction of energy storage compared to the actual observed grid load are shown in Figure 6. In this model, use of the battery-stored energy is prioritized to keep the home load below 4 kW at all times. The power to grid is offset between the actual grid load and modeled grid load because some of the PV power is used to charge the batteries. A larger battery system would reduce more of the power supplied to grid because it would hold more energy on-site. The results are that peak reduction is dramatic with virtually all peak values above 4 kW being eliminated. The SciLab code for this mode is shown for reference in Appendix B.

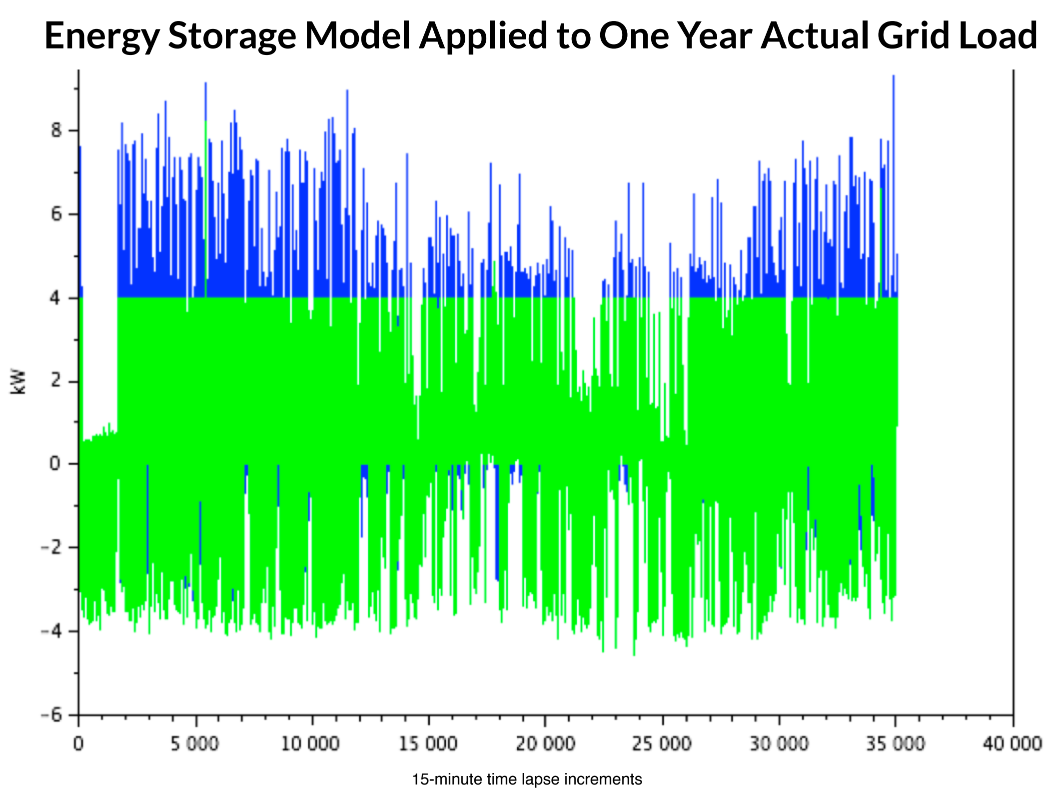

Figure 6. Modeled grid load from inclusion of energy storage compared to actual observed grid load over one year.

Home Data ID #	% reduction in average value of top 1% of peaks
26	43.1
545	5.8
1629	28.8
1642	33.3
7719	23.1

Table 2. Results for Methodology 2 Implementation on sample of 5 homes.

The project team ran the energy storage model for the same group of five homes that were used in the load control algorithm modeling. The model yielded reductions in the top 1% of peaks ranging from 5.8% to 43.1%, as show in Table 2, with an average reduction of 26.8%. The variability in impacts could be driven by a range of factors that warrant additional investigation in a future study.

The peak reduction in the top 1% of peak values for these homes seem significant, but additional analysis and field testing is necessary to validate the results and to verify that these reductions achieve the anticipated environmental and grid management benefits.

Estimated Grid Impact

Preliminary modeling of the impact of PV and energy storage integration on eight homes all on one transformer was completed to gain insight into the impact at the transformer level resulting from introduction of residential energy storage systems. Figure 7 shows the total grid load for the eight homes over one year, January 1 - December 31, 2014, comparing no storage to modeled storage.

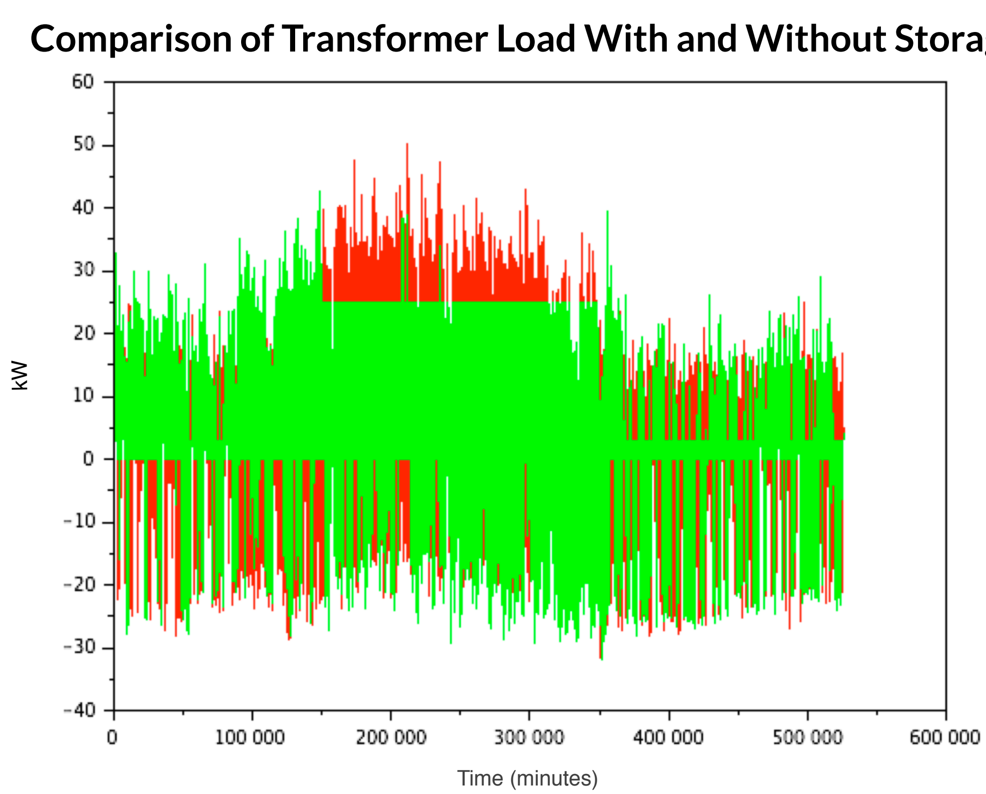

Figure 7. Total actual grid load of 8 homes at one transformer compared to storage-improved load over 1 year at 1-minute intervals

The modeling results show that integration of storage can reduce peak draw onto the transformer and attenuate excess power supply flowing onto the grid. The SciLab code for this model is shown for reference in Appendix C.

One important discovery made at this early stage of modeling transformer loads was that the effectiveness of the algorithm used to make storage charge/discharge decisions depends on the time of year and region as energy generation and consumption patterns significantly fluctuate across seasons and regions.

All of the data used in this initial model was pulled from homes in the Austin area, for which summer energy use is driven by HVAC loads, as shown in Figure 8 with the HVAC use shown in orange compared to all other electric loads for that day. This regional and seasonal variation makes it necessary to develop separate winter and summer algorithms for each region observed.

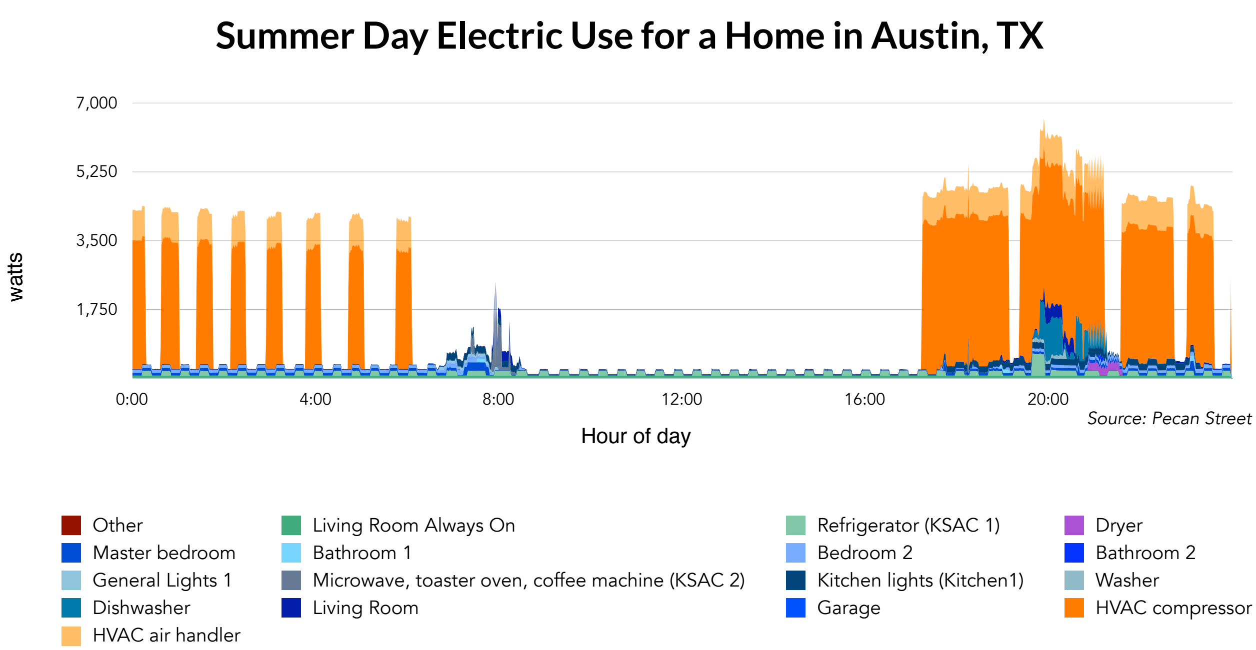

Figure 8. Disaggregated electric use observed by Pecan Street Inc. for a home in Austin, Texas over one day in summer 2013

To understand the impact of various storage capacities on the 8 home model, the number of service hours of improved load over one year at the transformer was evaluated, where improved load means reduced negative power or reduced peak power.

The results are shown in Figure 9, with the number of improved service hours on the y-axis out of the total number of annual service hours, 8,760. The fitted-curve of the model results showing the number of hours of improved load at the transformer appears to approach a maximum number of service hours per year that can be improved by introducing energy storage. Pecan Street will continue to expand the transformer model and gain a better understanding of the interactions among residential use, generation, and storage to minimize load variability. These insights will help guide figure of merit discussions that correspond to environmental and grid management benefits.

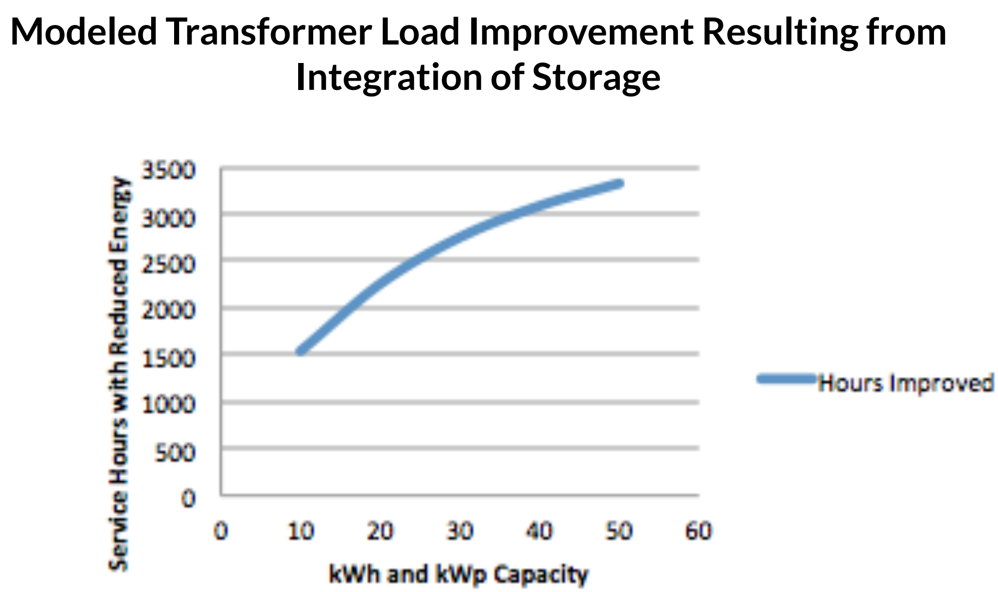

Figure 9. Annual service hours improved by energy storage capacity

The streamlined SciLab code used to model the PV/battery leveling and EV/Air1 leveling is recorded in Appendix D. This version of the code performs both of the analyses in a single run. The sample size was expanded to include 32 homes that have PV, EV charging, and Air Conditioning loads. The percent reduction in the top 1% of peaks is shown in the recorded output in Appendix E.

Comparison of Intra-Home Residential Smoothing Approaches

Comparing the impacts of implementing home load controls (Method 1) and introducing distributed energy storage (Method 2) on the intermittency management challenges presented to utilities by distributed PV systems reveals that energy storage systems are more effective at managing renewables intermittency at the source of generation. However, significant gains can be made through the less expensive approach of home load management. Figure 10 depicts the percent reduction resulting from each of these two methods as a function of reduced peak demand events at each home compared to the home's total annual kWh use.

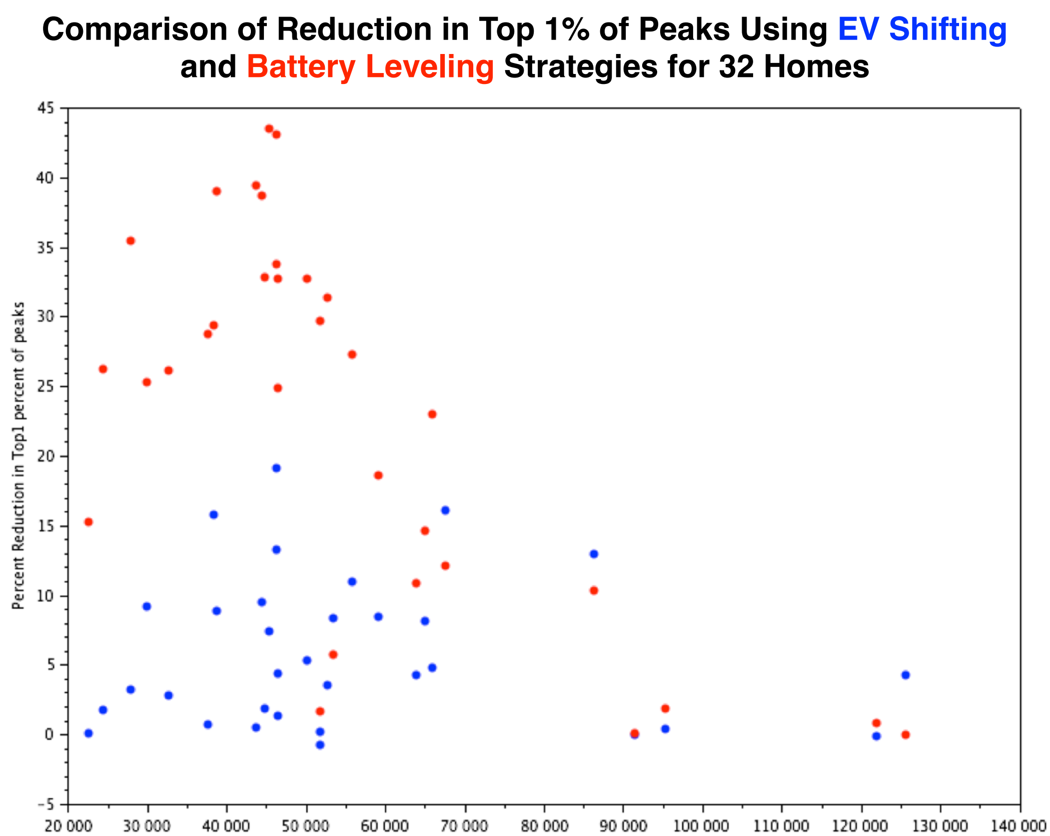

Figure 10. Annual energy use versus % reduction in top 1% of peaks for load leveling from battery storage compared to EV charge control

The homes analyzed have various levels of energy consumption, therefore each home's impact on the utility distribution load was weighted by the magnitude of the load at each particular point in time. Simple weighted averaging of each home by its respective annual consumption does not yield an accurate grid impact assessment because consumption over the year varies significantly. The Scilab analysis instead aggregates the relevant loads in a time synchronous fashion, yielding a more accurate model of the impact on the distribution grid.

Using this approach, the analysis suggests that if this group of homes were all on the same distribution system, their aggregate contribution to the distribution load would be impacted as follows:

1. There would be a 9.5% reduction in the top 1% of peaks from these homes if the PV/Battery leveling controls were implemented in each home.
2. There would be a 3.4% reduction in the top 1% of peaks from these homes if the EV/Air1 leveling controls were implemented in each home.

The results of these models suggest that an expanded study to validate the results on a larger and more diverse population of homes would be beneficial for utilities and other organizations seeking to understand the most effective methods for managing PV intermittency and utilizing DER to meet steep ramping demand.

In examining the sample of 32 homes more closely it becomes apparent that homes with higher ratios of PV-supplied electricity usage to total electricity consumption provide more benefit to the grid. Figure 11 shows the peak reduction potential versus the PV Ratio (average PV Generation/ Average usage) for 20 homes out of the 32 that had a high PV ratio. The peak reduction values of these high impact homes and their coincident aggregated average reductions on the top 1 percent peaks are provided in Appendix F.

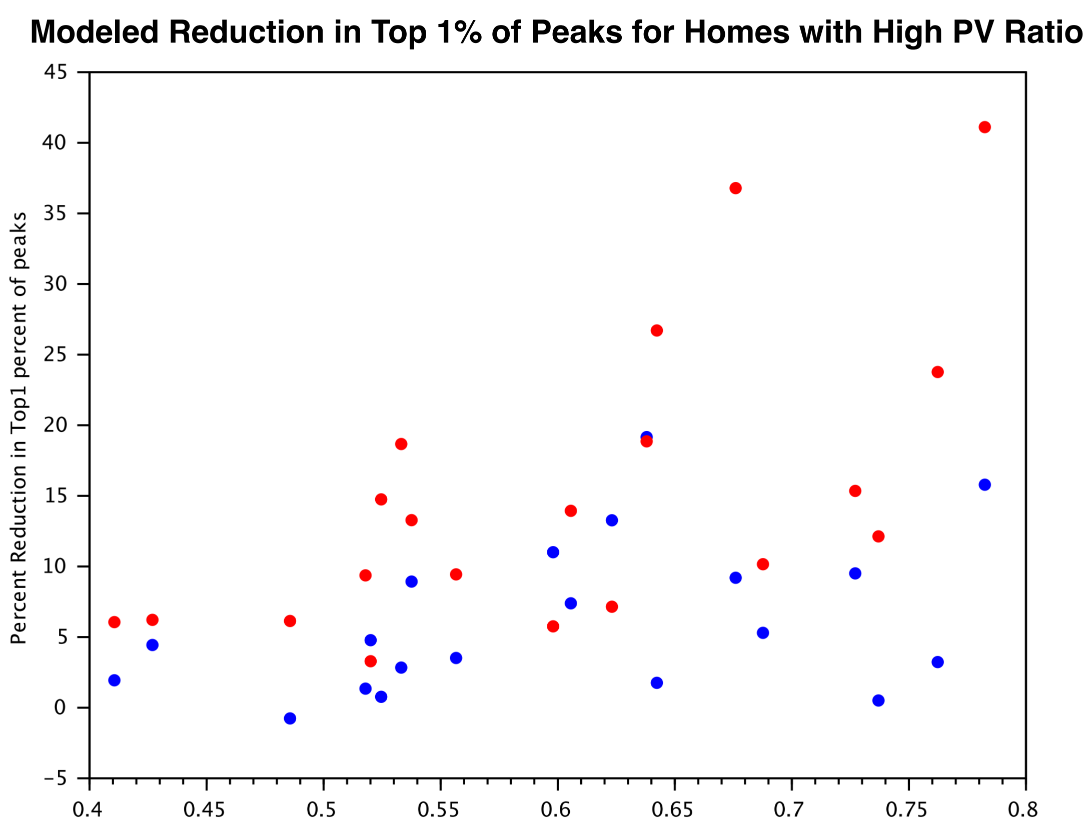

Figure 11. **Use to grid consumption ratio for 20 homes with PV systems averaging 5.5 kWh showing impact of load leveling from battery storage compared to EV charge control**

Method 3: Transformer-Level Load Balancing

To analyze how EV and PV systems spread across homes on the same distribution network can be synchronized to level grid demand, the project team developed an algorithm and custom EV charging system that modulated electric vehicle charging in response to energy generated by a PV system on a neighboring home tied into the same transformer. The EV charging system used the data on PV generation to make decisions that governed when and at what level of power draw the EV charged. If the electric vehicle was available for charging, the controller allowed it to charge at a rate of 1.4 kW. The charge rate was increased if the solar production of the neighboring house exceeded 1.4 kW. Using this method, the vehicle was always allowed to charge at a lower level and increased its charging when the neighboring structure had excess energy available.

Figure 12 shows the electrical demand for the residential structure with PV using a 15-minute running average. Averaging the curve allows general trends to be seen. On this particular day the home produced excess power in long stretches between 10 AM and 2 PM.

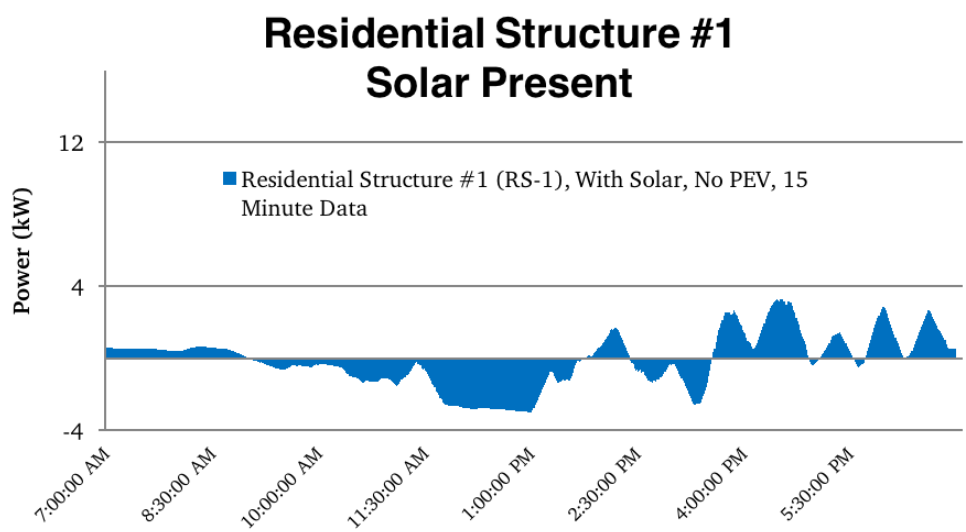

Figure 12. One-day residential structure demand/generation Curve, with solar, at 15-minute increments.

Figure 13 shows the combined electrical load of the two residential structures in the study. In this case, note that even the addition of a second structure without PV does not guarantee periods free from excess production.

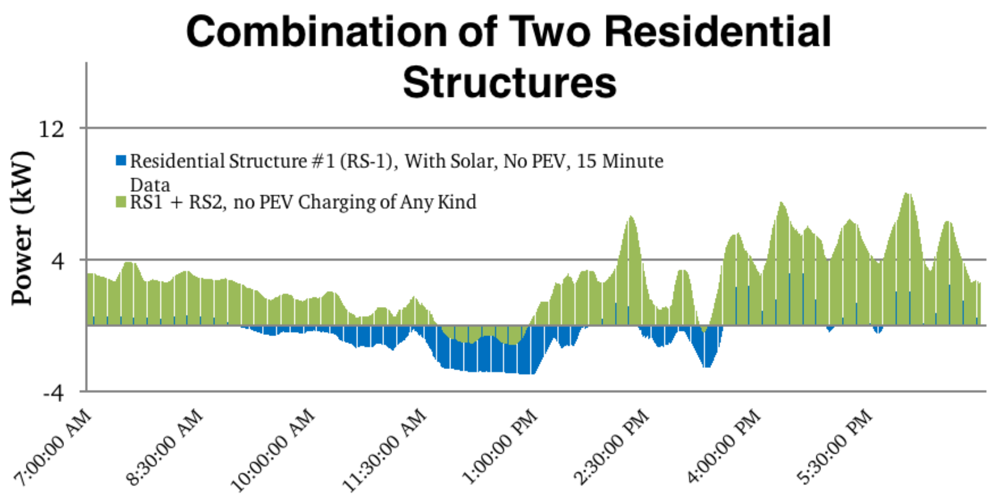

Figure 13. Demand curve of both residential structures showing the impact of PV and EV charging on the grid.

To model a worst case charging profile, the electric vehicle's charging was added to the load profile as an uncontrolled event not coordinated with the home producing PV power starting at 4 PM. Figure 14 shows the resulting load curve of the two residential structures, with the non-solar home charging an electric vehicle from 4 to 7 PM. The resulting load profile has an even higher demand curve in the afternoon and evening, with the utility needing to supply an average of 10-12 kW between 4:30 PM and 7:00 PM.

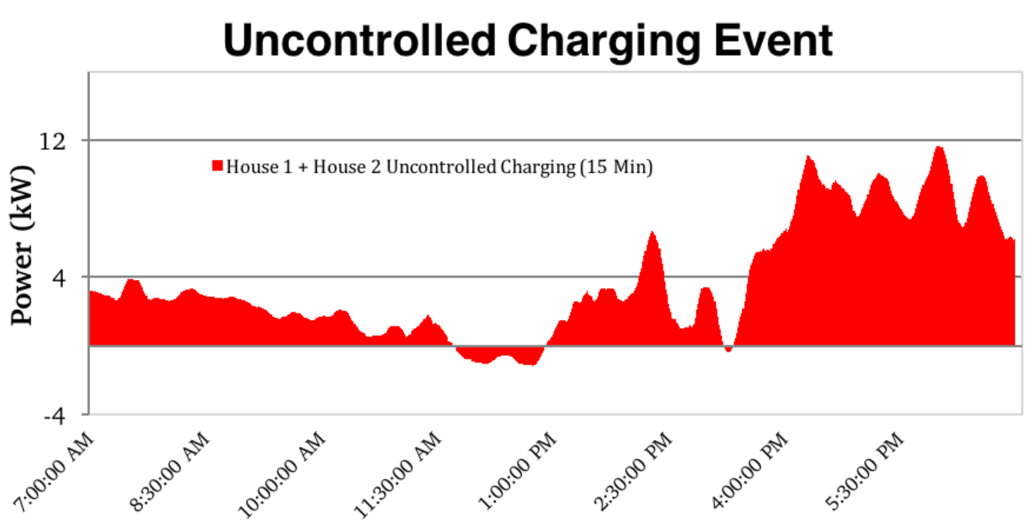

Figure 14. Modeled scenario of worst case charging event.

When the charging control device was employed to minimize the impact of excess solar production, the combined residential loads of the two homes were smoothed out with all PV power being consumed prior to the energy hitting the transformer, significantly reducing peak demand.

Figure 15 shows the charge profile of the electric vehicle compared to the load profile of the residential structure with PV. The shape of the charge profile is a mirror image of the excess PV generation profile during the hours of 10:30 AM to 3:00 PM. While this approach results in a slower EV charge time, the benefit to the utilities may be significant enough that an incentive, such as lower energy prices during off-peak periods, could be offered to compensate the owner for any potential inconvenience.

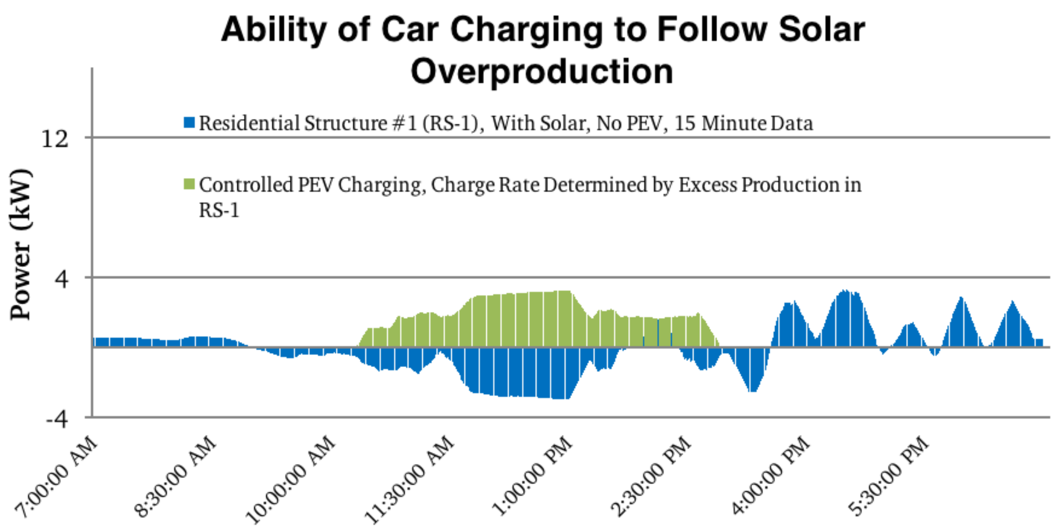

Figure 15. Controlled car charging load profile

Figure 16 shows the actual measured combined load profile of the two residential structures with PV and controlled EV charging in place.

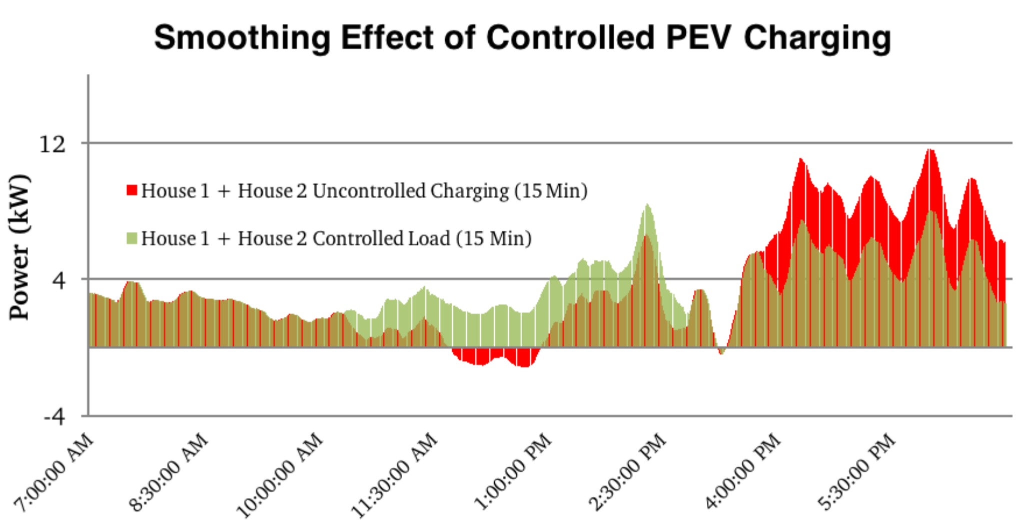

Figure 16. Inter-home coordination of distributed generation and demand

Coordination of PV generation and large load consumption between homes on the same transformer resulted in excess energy being pushed back onto the grid for only a small period of time in the afternoon, significantly reducing the potential grid management challenges these two homes presented individually to the utility when not operated in a coordinated manner.

The demonstration provides an example of the system-wide benefits that can be achieved if grid-edge, real-time data collection is used in the residential load control decision-making process. Pecan Street has collected 1-minute interval, circuit-level energy consumption and generation data for over 800 homes across the country and can provide guidance on best practices on data collection, storage, and management.

Residential Intermittency Data & Statistics

Data Overview

Pecan Street's residential energy database has granular data representations of over 1,500 home years of residential energy usage patterns, at least 300 home years of which include on-site PV production. Embedded in this data are insights into how residential loads and energy storage may be coordinated to better align residential PV production with home energy consumption on an individual home and aggregated basis.

Data Source

The data used in this study was provided by Pecan Street's subscriber base, where participants may opt-in to have their residence's electricity and power usage data monitored at a high resolution. Pecan Street uses the eGauge Systems energy monitor, which is installed inside the main electrical service panel by a certified, master electrician. This device can log up to 12 current transducers (CTs) that record current flow, voltage, and real and apparent power on individually labeled circuit breakers at 1-minute intervals. Loads are verified and appropriately labeled by the master electrician during installation. Electric data can be viewed on any web-enabled device through the built-in web server. Data is pushed to an external server where it is maintained, managed, and owned by Pecan Street Inc. Through the support of Pecan Street's industry partners, this data is made available at no cost to university researchers through Pecan Street's online, interactive data portal, Dataport. The data is available at affordable rates to industry partners.

To establish a baseline of load variability for homes without PV systems, the energy use profiles of 32 homes were analyzed for this study. All the analyzed homes contained a single HVAC unit, no pool, and an electric vehicle. The homes were analyzed for basic parameters such as mean electrical usage, maximum electrical demand, and standard deviation of electrical usage. Table 3 shows the overall statistics for the electrical usage of the 32 examined residences. The average maximum demand over all homes was 14.2 kW. The average for the homes over the year was only 1.3 kW, meaning that the Crest Factor or Maximum/Average is a figure of greater than 10:1. The standard deviation was 1.6 kW, therefore the average demand is 8 standard deviations lower than the maximum demand, indicating a high variability in home demand power.

Electrical Demand Statistics

Maximum 12 Month Demand	14.2 kW
Standard Deviation of Demand	1.6 kW
Average 12 Month Demand	1.3 kW
Maximum/Average (Crest Factor)	11.1 kW

Table 3. Electrical demand statistics for examined set of 32 homes

In addition, the homes were examined to see how electrical appliances that are perceived to be heavy users of electricity are used simultaneously. Three appliances were selected: Air Conditioning (HVAC), plug-in electric vehicles (EV), and home refrigerators. Three combinations were examined: HVAC and EV, which would yield the greatest electrical demand, HVAC and fridge, and refrigerator and EV.

Table 4 shows the concurrence of the three combinations of major residential loads. For this study, concurrence is defined as the percent of the year that the devices were on in the same one-minute period. In this case, the two largest power demands, HVAC and EV, were on at the same time 1.4% of the year, or roughly 122 hours.

Electrical Load Concurrence

HVAC and Refrigerator	9.3%
HVAC and Electric Vehicle	1.4%
Refrigerator and Electric Vehicle	3.7%

Table 4. Electrical load concurrence for examined examined set of 32 homes

Interestingly, the number of times that the individual loads were occurring simultaneously was low. This indicates that the variability of power demand for a residential structure may be due to a wide variety of electrical loads and not necessarily the perceived "heavy hitters" during much of the year. Further analysis is required to determine what percent of the top energy peaks during the year involve the alignment of the loads making the greatest demands to the grid.

Appendix A
SciLab Code for Modeling EV/Air1 Avoidance Load Leveling

```
//Clear Scilab and initialize MacBook memory for maximum data handling capacity
 clear
 stacksize('max');

 //DEFINE FUNCTIONS////////////

 //Get_Power_Data/////////////////////////////////////////////
 function [Date, Grid, Use, Gen, EV, Air1, Air2, Misc_Loads, Increments, Interval,
Date_Range]=Get_Power_Data(Target_File);
 housefile = fullfile(Target_File);
 housedata2 = csvRead(housefile,",",[]);

 Date=getdate(housedata2(:,1));
 Use=housedata2(:,2);
 Gen=housedata2(:,3);
 EV=housedata2(:,4);
 Air1=housedata2(:,5);
 Air2=housedata2(:,6);
 Misc_Loads=Use-EV-Air1-Air2;

 [Increments,y]=size(Date);//Count size of datafile
 Interval=etime(Date(3,1:10),Date(2,1:10))/3600;//Interval in hours
 Date_Range=string(Date(1,2))+"/"+string(Date(1,6))+"/"+string(Date(1,1))+" through "+string(Date(Increments,2))
+"/"+string(Date(Increments,6))+"/"+string(Date(Increments,1));
 Grid=Use-Gen;
 endfunction
 ////////////////////////////////////////////////////////
 //Time_Check    Count intervals over Grid_Max for Grid(or under if Over_Under = -1)   and calculates total time.
 function [Over_Max]=Time_Check(Grid, Grid_Max, Interval, Over_Under)

 Over_Max=0;//Inititalizes counter for original grid signal conditional events
 [Increments,y]=size(Grid);

 for n=1:Increments
   if Grid(n)>Grid_Max*Over_Under  then Over_Max=Over_Max+1;
   else end
 end
 Over_Max=Over_Max*Interval
 endfunction

 ////////////////////////////////////////////////////////
 //Find_On_Value
 function [On_Value, On_StDev]=Find_On_Value(Vector)

 Target=Vector;
 counter=0;
 Total=zeros(Target);
 Test=mean(Target);
 disp(Test);
 [Increments,y]=size(Target);
```

```
for n=1:Increments
  if Target(n)>=Test then counter=counter+1;Total(counter)=Target(n);
  else end
end
On_Value=sum(Total)/counter;

if On_Value>(1.1*Test) then On_Value=Find_On_Value (Total(1:counter)); else
end
On_StDev=stdev(Total(1:counter));

endfunction

////////////////////////////////////////////////////////////

//Energy_Check    Count energy intervals over (or under if Over_Under = -1) Grid_Max for Grid  and calculates total energy.
function [Over_Max]=Energy_Check(Grid, Grid_Max, Interval, Over_Under)

Over_Max=0;//Inititalizes counter for original grid signal conditional events
[Increments,y]=size(Grid);

for n=1:Increments
  if Grid(n)>Grid_Max*Over_Under  then Over_Max=Over_Max+Grid(n);
  else end
end
Over_Max=Over_Max/Interval
endfunction

////////////////////////////////////////////////////////////
function [Lev]=Level_Max_Min(V, Max, Min)

[n2,y]=size(V);
Lev=zeros(V);

for n=1:n2
  if V(n)>Max  then Lev(n)=V(n)-Max;
     elseif V(n)<Min then Lev(n)=V(n)-Min;
  else end
end
endfunction
////////////////////////////////////////////////////////////
//time_shift     timeshifts vector V to avoid vector A. values larger than V_On to avoid A_On vector values. Also returns delay
vector. delay units are intervals. convert to hour by multiplying by intervals per time unit (hour).

function [V_Shift, V_Buff]=time_shift(V, A);
[Increments,y]=size(V);
Bank=0;
[V_On,V_On_StDev]=Find_On_Value(V);

V_On=V_On-V_On_StDev;

[A_On,A_On_StDev]=Find_On_Value(A);

A_On=A_On-A_On_StDev;
```

```
clf(2)
scf(2)
G1=gsort(Grid);
G2=gsort(Grid-EV+EV_Shift);
plot(G1(1:Increments*Top),"b");
plot(G2(1:Increments*Top),"g");

xtitle(Target_File(run),'Top '+string(Top*100)+' percent of peaks sorted by decreasing magnitude. Blue=Grid. Green=Grid Levelled. Top '+string(Top*100)+' percent of peaks reduced by an average of '+string(reduction)+'  percent','kW');

clf(3)
scf(3)
G1=Grid;
G2=Grid-EV+EV_Shift;
plot(G1,"b");
//plot(G2,"g");

xtitle(Target_File(run),'15 minute increments. Blue=Grid. Green=Grid Levelled. Top '+string(Top*100)+' percent of peaks reduced by an average of '+string(reduction)+'  percent','kW');

n1=Find_On_Value(EV);
n2=Find_On_Value(Air1);

end
////////////////////////////////////////////////////////////
```

Appendix B
SciLab Code for Modeling Battery Load Leveling

```
//Clear Scilab and initialize MacBook memory for maximum data handling capacity
 clear
 stacksize('max');

 //DEFINE FUNCTIONS////////////

 //Get_Power_Data//////////////////////////////////////////////
 function [Date, Grid, Use, Gen, EV, Air1, Air2, Misc_Loads, Increments, Interval,
Date_Range]=Get_Power_Data(Target_File);
 housefile = fullfile(Target_File);
 housedata2 = csvRead(housefile,",",[]);

 Date=getdate(housedata2(:,1));
 Use=housedata2(:,2);
 Gen=housedata2(:,3);
 EV=housedata2(:,4);
 Air1=housedata2(:,5);
 Air2=housedata2(:,6);
 Misc_Loads=Use-EV-Air1-Air2;

 [Increments,y]=size(Date);//Count size of datafile
 Interval=etime(Date(3,1:10),Date(2,1:10))/3600;//Interval in hours
 Date_Range=string(Date(1,2))+"/"+string(Date(1,6))+"/"+string(Date(1,1))+" through "+string(Date(Increments,2))
+"/"+string(Date(Increments,6))+"/"+string(Date(Increments,1));
 Grid=Use-Gen;
 endfunction
 /////////////////////////////////////////////////////////

 //Time_Check    Count intervals over Grid_Max for Grid(or under if Over_Under = -1)   and calculates total time.
 function [Over_Max]=Time_Check(Grid, Grid_Max, Interval, Over_Under)

 Over_Max=0;//Inititalizes counter for original grid signal conditional events
 [Increments,y]=size(Grid);

 for n=1:Increments
   if Grid(n)>Grid_Max*Over_Under  then Over_Max=Over_Max+1;
   else end
 end
 Over_Max=Over_Max*Interval
 endfunction

 /////////////////////////////////////////////////////////
 //Find_On_Value
 function [On_Value, On_StDev]=Find_On_Value(Vector)

 Target=Vector;
 counter=0;
 Total=zeros(Target);
```

```
Test=mean(Target);
disp(Test);

[Increments,y]=size(Target);

for n=1:Increments
  if Target(n)>=Test then counter=counter+1;Total(counter)=Target(n);
  else end
end
On_Value=sum(Total)/counter;

if On_Value>(1.1*Test) then On_Value=Find_On_Value (Total(1:counter)); else
end
On_StDev=stdev(Total(1:counter));

endfunction

//////////////////////////////////////////////////////////////

//Energy_Check    Count energy intervals over (or under if Over_Under = -1) Grid_Max for Grid  and calculates total
energy.
function [Over_Max]=Energy_Check(Grid, Grid_Max, Interval, Over_Under)

Over_Max=0;//Inititalizes counter for original grid signal conditional events
[Increments,y]=size(Grid);

for n=1:Increments
  if Grid(n)>Grid_Max*Over_Under  then Over_Max=Over_Max+Grid(n);
  else end
end
Over_Max=Over_Max/Interval
endfunction

//////////////////////////////////////////////////////////////
function [Lev]=Level_Max_Min(V, Max, Min)

[n2,y]=size(V);
Lev=zeros(V);

for n=1:n2
  if V(n)>Max  then Lev(n)=V(n)-Max;
     elseif V(n)<Min then Lev(n)=V(n)-Min;
  else end
end
endfunction
/////////////////////////////////////////////////////////////
//time_shift     timeshifts vector V to avoid vector A. values larger than V_On to avoid A_On vector values. Also returns
delay vector. delay units are intervals. convert to hour by multiplying by intervals per time unit (hour).

function [V_Shift, V_Buff]=time_shift(V, A);
[Increments,y]=size(V);
Bank=0;
```

```
[V_On,V_On_StDev]=Find_On_Value(V);

V_On=V_On-V_On_StDev;

[A_On,A_On_StDev]=Find_On_Value(A);

A_On=A_On-A_On_StDev;

for n=1:Increments
  if V(n)>= V_On & (A(n)>=A_On) then Bank=Bank+V(n);V(n)=0;
  elseif V(n)<V_On & (A(n)<A_On & Bank>V_On) then V(n)=V(n)+V_On;Bank=Bank-V_On;if V(n)<V_On &
A(n)<A_On then V(n)=V(n)+Bank;Bank=0; else end;
    else end;
    V_Buff(n)=Bank/V_On;

end
V_Shift=V;
endfunction
////////////////////////////////////////

//Average_On finds average value of a load vector (Negative) excluding values smaller than Load_Percent times the
moving average.
function [On_Value]=Average_On(Target_Vector, Noise_Percent);
Total=0;
count=1;
for n=1:Increments
   if Target_Vector(n)>((Total/count)*Noise_Percent) then
  Total=Total+Target_Vector(n);
  count=count+1;
  else end;

end
  On_Value=Total/count;

endfunction

////////////////////////////////////////
function [Lev, Lev1]=Battery_Limits(Lev, Capacity, Chg_Rate, Dis_Rate)

Lev1=cumsum(Lev);
[n2,y]=size(Lev);

for n=1:n2
  if Lev1(n)>Capacity then Lev(n)=0; Lev1=cumsum(Lev);
  elseif Lev1(n)<0 then Lev(n)=0; Lev1=cumsum(Lev);
  elseif Lev(n)<Chg_Rate then Lev(n)=Chg_Rate;Lev1=cumsum(Lev);
   elseif Lev(n)>Dis_Rate then Lev(n)=Dis_Rate;Lev1=cumsum(Lev);
    else end;
end

Lev1=cumsum(Lev);
endfunction
/////////////////////////////////
```

```
Target File(1)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_26_15minute_group1_expanded_1year_20140701_to_20150630.csv";
Target_File(2)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_370_15minute_group1_expanded_1year_20140701_to_20150630.csv";
Target_File(3)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_545_15minute_group1_expanded_1year_20140701_to_20150630.csv";
Target_File(4)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_1629_15minute_group1_expanded_1year_20140701_to_20150630.csv";
Target_File(5)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_1642_15minute_group1_expanded_1year_20140701_to_20150630.csv";
Target_File(6)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_3192_15minute_group1_expanded_1year_20140701_to_20150630.csv";
Target_File(7)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_3967_15minute_group1_expanded_1year_20140701_to_20150630.csv";
Target_File(8)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_4767_15minute_group1_expanded_1year_20140701_to_20150630.csv";
Target_File(9)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_6691_15minute_1year_July1st_2014_to_Jun30_2015.csv";
Target_File(10)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_7719_15minute_group1_expanded_1year_20140701_to_20150630.csv";
Target_File(11)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_9729_15minute_group1_expanded_1year_20140701_to_20150630.csv";
Target_File(12)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_9776_1minute_1year_July1st_2014_to_Jun30_2015.csv";
Target_File(13)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_9932_15minute_group1_expanded_1year_20140701_to_20150630.csv";
////////////////////////////////////////////////////////////
for run=12:12
[Date,Grid,Use,Gen,EV,Air1,Air2,Misc_Loads,Increments,Interval,Date_Range]=Get_Power_Data(Target_File(run));
Max=4;Min=0;Capacity=7/Interval;Chg_Rate=-6.4;Dis_Rate=6;
Top=.01;
[Lev]=Level_Max_Min(Grid,Max,Min);
[Lev,Lev1]=Battery_Limits(Lev,Capacity,Chg_Rate,Dis_Rate);
G1=Grid;
G2=Grid-Lev;
R1=gsort(G1);M1=mean(R1(1:Top*Increments));
R2=gsort(G2);M2=mean(R2(1:Top*Increments));
reduction=M2/M1;reduction=int(reduction*10000)/100;
clf(1)
scf(1)
plot(R1,"b");
plot(R2,"g");
xtitle(Target_File(run),'15 minute increments. Blue=Grid. Green=Grid Levelled. Top '+string(Top*100)+' percent of
peaks reduced by an average of '+string(reduction)+'  percent','kW');
clf(3)
scf(3)
plot(G1,"b");
plot(G2,"g");
xtitle(Target_File(run),'15 minute increments. Blue=Grid. Green=Grid Levelled. Top '+string(Top*100)+' percent of
peaks reduced by an average of '+string(reduction)+'  percent','kW');
end
////////////////////////////////////////////////////////////
```

Appendix C
SciLab Code for Modeling Transformer Load Leveling with Energy Storage

```
//************************WINTER
Winter_Grid_Limit=3;
Summer_Grid_Limit=25;
Storage_Cap = 50;
Storage_Current = 10;
Storage_Output = Storage_Cap;
Grid_Limit = Winter_Grid_Limit;
Grid_Min = 0;

for x = 1:150000
   Grid = total_grid_sort(x,1);
   capped_grid(x,1) = Grid;

if ((Grid >Grid_Limit) & (Storage_Current>0))
   Required = Grid-Grid_Limit;
   if (Required<Storage_Output)
      capped_grid(x,1) = Grid - Required;
      Storage_Current = Storage_Current-Required/60;
   end
   if (Required>=Storage_Output)
      capped_grid(x,1) = Grid - Storage_Output;
      Storage_Current = Storage_Current - Storage_Output/60;
   end
end

if((Grid<Grid_Min)& (Storage_Current<=Storage_Cap))
  Required = abs(Grid-Grid_Min);
  if (Required<=Storage_Output)
   Storage_Current = Storage_Current + .9*Required/60;
   capped_grid(x,1) = Grid + Required;
  end
  if (Required>Storage_Output)
   Storage_Current = Storage_Current + .9*Storage_Output/60;
   capped_grid(x,1) = Grid + Storage_Output;
  end
end
end

   Storage_Current = Storage_Current + .9*Required/60;
   capped_grid(x,1) = Grid + Required;
  end
  if (Required>Storage_Output)
   Storage_Current = Storage_Current + .9*Storage_Output/60;
   capped_grid(x,1) = Grid + Storage_Output;
  end
end
end
```

```
//************************Summer

Grid_Limit = Summer_Grid_Limit;
for x = 150001:350000
   Grid = total_grid_sort(x,1);
   capped_grid(x,1) = Grid;
if ((Grid >Grid_Limit) & (Storage_Current>0))
   Required = Grid-Grid_Limit;
   if (Required<Storage_Output)
      capped_grid(x,1) = Grid - Required;
      Storage_Current = Storage_Current-Required/60;
   end
   if (Required>=Storage_Output)
      capped_grid(x,1) = Grid - Storage_Output;
      Storage_Current = Storage_Current - Storage_Output/60;
   end
end

//******************Winter Rules

Grid_Limit = Winter_Grid_Limit;

for x = 350001:array_length
   Grid = total_grid_sort(x,1);
   capped_grid(x,1) = Grid;
if ((Grid >Grid_Limit) & (Storage_Current>0))
   Required = Grid-Grid_Limit;
   if (Required<Storage_Output)
      capped_grid(x,1) = Grid - Required;
      Storage_Current = Storage_Current-Required/60;
   end
   if (Required>=Storage_Output)
      capped_grid(x,1) = Grid - Storage_Output;
      Storage_Current = Storage_Current - Storage_Output/60;
   end
end

if((Grid<Grid_Min)& (Storage_Current<=Storage_Cap))
  Required = abs(Grid-Grid_Min);
  if (Required<=Storage_Output)
plot(total_grid_sort,"r");
plot(capped_grid,"g");

negative = 0;
positive = 0;

for x = 1:array_length
   if(((total_grid_sort(x,1)) < 0) & (capped_grid(x,1) >= 0))
      negative = negative+1;
   end
end
for x = 1:array_length
   if((total_grid_sort(x,1)> 0) & (capped_grid(x,1)<total_grid_sort(x,1)))
      positive= positive+1;
   end
end
hours = positive/60 + negative/60
```

Appendix D
Scilab Code Used to Model the PV/Battery leveling and EV/Air1 leveling

```
//Clear Scilab and initialize MacBook memory for maximum data handling capacity
  clear
  stacksize('max');

  //DEFINE FUNCTIONS////////////

  //Get_Power_Data//////////////////////////////////////////////
  function [Date, Grid, Use, Gen, EV, Air1, Air2, Misc_Loads, Increments, Interval,
Date_Range]=Get_Power_Data(Target_File);
  housefile = fullfile(Target_File);
  housedata2 = csvRead(housefile,",",[]);

  Date=getdate(housedata2(:,1));
  Use=housedata2(:,2);
  Gen=housedata2(:,3);
  EV=housedata2(:,4);
  Air1=housedata2(:,5);
  Air2=housedata2(:,6);
  Misc_Loads=Use-EV-Air1-Air2;

  [Increments,y]=size(Date);//Count size of datafile
  Interval=etime(Date(3,1:10),Date(2,1:10))/3600;//Interval in hours
  Date_Range=string(Date(1,2))+"/"+string(Date(1,6))+"/"+string(Date(1,1))+" through "+string(Date(Increments,2))
+"/"+string(Date(Increments,6))+"/"+string(Date(Increments,1));
  Grid=Use-Gen;
  endfunction
  ///////////////////////////////////////////////////////////

  //Time_Check    Count intervals over Grid_Max for Grid(or under if Over_Under = -1)   and calculates total time.
  function [Over_Max]=Time_Check(Grid, Grid_Max, Interval, Over_Under)

  Over_Max=0;//Inititalizes counter for original grid signal conditional events
  [Increments,y]=size(Grid);
   for n=1:Increments
    if Grid(n)>Grid_Max*Over_Under  then Over_Max=Over_Max+1;
    else end
  end
  Over_Max=Over_Max*Interval
  endfunction

  ///////////////////////////////////////////////////////////
  //Find_On_Value
  function [On_Value, On_StDev]=Find_On_Value(Vector)
  Target=Vector;
  counter=0;
  Total=zeros(Target);
  Test=mean(Target);
end

if((Grid<Grid_Min)& (Storage_Current<=Storage_Cap))
```

```
if((Grid<Grid_Min)& (Storage_Current<=Storage_Cap))
   Required = abs(Grid-Grid_Min);
   if (Required<=Storage_Output)
     Storage_Current = Storage_Current + .9*Required/60;
     capped_grid(x,1) = Grid + Required;
   end
   if (Required>Storage_Output)
     Storage_Current = Storage_Current + .9*Storage_Output/60;
     capped_grid(x,1) = Grid + Storage_Output;
   end
end
end
//*******************Winter Rules

Grid_Limit = Winter_Grid_Limit;
for x = 350001:array_length
    Grid = total_grid_sort(x,1);
    capped_grid(x,1) = Grid;
if ((Grid >Grid_Limit) & (Storage_Current>0))
    Required = Grid-Grid_Limit;
    if (Required<Storage_Output)
        capped_grid(x,1) = Grid - Required;
        Storage_Current = Storage_Current-Required/60;
    end
    if (Required>=Storage_Output)
        capped_grid(x,1) = Grid - Storage_Output;
        Storage_Current = Storage_Current - Storage_Output/60;
    end
end
if((Grid<Grid_Min)& (Storage_Current<=Storage_Cap))
   Required = abs(Grid-Grid_Min);
   if (Required<=Storage_Output)
     Storage_Current = Storage_Current + .9*Required/60;
     capped_grid(x,1) = Grid + Required;
   end
   if (Required>Storage_Output)
     Storage_Current = Storage_Current + .9*Storage_Output/60;
     capped_grid(x,1) = Grid + Storage_Output;
   end
end
end
plot(total_grid_sort,"r");
plot(capped_grid,"g");
negative = 0;
positive = 0;
for x = 1:array_length
    if(((total_grid_sort(x,1)) < 0) & (capped_grid(x,1) >= 0))
        negative = negative+1;
    end
end

for x = 1:array_length
    if((total_grid_sort(x,1)> 0) & (capped_grid(x,1)<total_grid_sort(x,1)))
        positive= positive+1;
    end
end
hours = positive/60 + negative/60
```

```
[Increments,y]=size(Target);

for n=1:Increments
  if Target(n)>=Test then counter=counter+1;Total(counter)=Target(n);
  else end
end
On_Value=sum(Total)/counter;

if On_Value>(1.1*Test) then On_Value=Find_On_Value (Total(1:counter)); else
end
On_StDev=stdev(Total(1:counter));

endfunction

//////////////////////////////////////////////////////////////

//Energy_Check    Count energy intervals over (or under if Over_Under = -1) Grid_Max for Grid  and calculates total energy.
function [Over_Max]=Energy_Check(Grid, Grid_Max, Interval, Over_Under)

Over_Max=0;//Inititalizes counter for original grid signal conditional events
[Increments,y]=size(Grid);

for n=1:Increments
  if Grid(n)>Grid_Max*Over_Under  then Over_Max=Over_Max+Grid(n);
  else end
end
Over_Max=Over_Max/Interval
endfunction

//////////////////////////////////////////////////////////////
function [Lev]=Level_Max_Min(V, Max, Min)

[n2,y]=size(V);
Lev=zeros(V);

for n=1:n2
  if V(n)>Max  then Lev(n)=V(n)-Max;
     elseif V(n)<Min then Lev(n)=V(n)-Min;
  else end
end
endfunction
//////////////////////////////////////////////////////////////
//time_shift     timeshifts vector V to avoid vector A. values larger than V_On to avoid A_On vector values. Also returns delay
vector. delay units are intervals. convert to hour by multiplying by intervals per time unit (hour).

function [V_Shift, V_Buff]=time_shift(V, A);
[Increments,y]=size(V);
Bank=0;
[V_On,V_On_StDev]=Find_On_Value(V);

V_On=V_On-V_On_StDev;

[A_On,A_On_StDev]=Find_On_Value(A);
```

```
A_On=A_On-A_On_StDev;

for n=1:Increments
   if V(n)>= V_On & (A(n)>=A_On) then Bank=Bank+V(n);V(n)=0;
   elseif V(n)<V_On & (A(n)<A_On & Bank>V_On) then V(n)=V(n)+V_On;Bank=Bank-V_On;if V(n)<V_On & A(n)<A_On
then V(n)=V(n)+Bank;Bank=0; else end;
       else end;
       V_Buff(n)=Bank/V_On;

end
V_Shift=V;
endfunction
//////////////////////////////////////////////

//Average_On finds average value of a load vector (Negative) excluding values smaller than Load_Percent times the moving
average.
function [On_Value]=Average_On(Target_Vector, Noise_Percent);
Total=0;
count=1;
for n=1:Increments
    if Target_Vector(n)>((Total/count)*Noise_Percent) then
   Total=Total+Target_Vector(n);
   count=count+1;
   else end;

end
   On_Value=Total/count;

endfunction

/////////////////////////////////////////////
function [Lev, Lev1]=Battery_Limits(Lev, Capacity, Chg_Rate, Dis_Rate)

Lev1=cumsum(Lev);
[n2,y]=size(Lev);

for n=1:n2
   if Lev1(n)>Capacity then Lev(n)=0; Lev1=cumsum(Lev);
   elseif Lev1(n)<0 then Lev(n)=0; Lev1=cumsum(Lev);
   elseif Lev(n)<Chg_Rate then Lev(n)=Chg_Rate;Lev1=cumsum(Lev);
     elseif Lev(n)>Dis_Rate then Lev(n)=Dis_Rate;Lev1=cumsum(Lev);
       else end;
end

Lev1=cumsum(Lev);

endfunction

//////////////////////////////////////

Target_File(1)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_26_15minute_group1_expanded_1year_20140701_to_20150630.csv";
Target_File(2)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
```

```
dataid_545_15minute_group1_expanded_1year_20140701_to_20150630.csv";
 Target_File(3)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_1629_15minute_group1_expanded_1year_20140701_to_20150630.csv";
 Target_File(4)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_1642_15minute_group1_expanded_1year_20140701_to_20150630.csv";
 Target_File(5)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_6691_15minute_1year_July1st_2014_to_Jun30_2015.csv";
 Target_File(6)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_7719_15minute_group1_expanded_1year_20140701_to_20150630.csv";
 Target_File(7)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_114_15minute_group2_expanded_1year_20140701_to_20150630.csv";
 Target_File(8)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_364_15minute_group2_expanded_1year_20140701_to_20150630.csv";
 Target_File(9)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_624_15minute_group2_expanded_1year_20140701_to_20150630.csv";
 Target_File(10)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_661_15minute_group2_expanded_1year_20140701_to_20150630.csv";
 Target_File(11)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_1169_15minute_group2_expanded_1year_20140701_to_20150630.csv";
 Target_File(12)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_1714_15minute_group2_expanded_1year_20140701_to_20150630.csv";
 Target_File(13)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_2470_15minute_group2_expanded_1year_20140701_to_20150630.csv";
 Target_File(14)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_2638_15minute_group2_expanded_1year_20140701_to_20150630.csv";
 Target_File(15)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_2814_15minute_group2_expanded_1year_20140701_to_20150630.csv";
 Target_File(16)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_3036_15minute_group2_expanded_1year_20140701_to_20150630.csv";
 Target_File(17)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_3367_15minute_group2_expanded_1year_20140701_to_20150630.csv";
 Target_File(18)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_3723_15minute_group2_expanded_1year_20140701_to_20150630.csv";
  Target_File(19)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_4336_15minute_group2_expanded_1year_20140701_to_20150630.csv";
   Target_File(20)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_4352_15minute_group2_expanded_1year_20140701_to_20150630.csv";
    Target_File(21)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_6990_15minute_group2_expanded_1year_20140701_to_20150630.csv";
     Target_File(22)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_6836_15minute_group2_expanded_1year_20140701_to_20150630.csv";
     Target_File(23)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_7850_15minute_group2_expanded_1year_20140701_to_20150630.csv";
     Target_File(24)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_7863_15minute_group2_expanded_1year_20140701_to_20150630.csv";
     Target_File(25)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_7940_15minute_group2_expanded_1year_20140701_to_20150630.csv";
     Target_File(26)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_8156_15minute_group2_expanded_1year_20140701_to_20150630.csv";
     Target_File(27)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_2018_15minute_group3_expanded_1year_20140701_to_20150630.csv";
     Target_File(28)="/Users/Berthaskell/Desktop/Sci Lab Modelling/Input Data/
dataid_4957_15minute_group3_expanded_1year_20140701_to_20150630.csv";
```

```
 reduction_Lev=M2/M1;
 Reduction_Lev(run)=100-int(reduction_Lev*10000)/100;
reduction_Shift=M3/M1;
Reduction_Shift(run)=100-int(reduction_Shift*10000)/100;

Total_Use=sum(Use);
Total_EV=sum(EV);

disp([run Reduction_Lev(run) Reduction_Shift(run) Total_Use]);

 plot(Total_Use,Reduction_Shift(run),".b");
 plot(Total_Use,Reduction_Lev(run),".r");

 //plot(R2,"g");

 //clf(10+run)
 //scf(10+run)

 //plot(G1,"b");
 //plot(G2,"g");

 //xtitle(Target_File(run),'15 minute increments. Blue=Grid. Green=Grid Levelled. Top '+string(Top*100)+' percent of peaks
reduced by an average of '+string(reduction)+'  percent','kW');

end

//////////

 R1=gsort(A_G1);M1=mean(R1(1:Top*Increments));

 R2=gsort(A_G2);M2=mean(R2(1:Top*Increments));

 R3=gsort(A_G3);M3=mean(R3(1:Top*Increments));

 reduction_Lev=M2/M1;
 reduction_Shift=M3/M1;

 A_Reduction_Lev=100-int(reduction_Lev*10000)/100;
 A_Reduction_Shift=100-int(reduction_Shift*10000)/100;

disp("Aggregated reduction of top 1% of peaks from battery levelling / EV Shifting:");
disp ([A_Reduction_Lev A_Reduction_Shift])

   xtitle('Number of houses='+ string(run),'TOTAL kWh used  (Blue=Battery Levelling. Red=EV Shifted.)','Percent Reduction in
Top'+string(Top*100)+' percent of peaks');
```

Appendix E
PV/Battery Leveling and EV/Air1 Leveling Results Top 1% of Peaks

-->exec('/Users/Berthaskell/Desktop/Sci Lab Modelling/EDF final LEV and SHIFT .sce', -1)

Run	Reduction_Lev	Reduction Shift	Total Use
1.	43.11	19.16	46197.907
2. 5.8	8.38	53394.54	
3. 28.77	0.77	37654.621	
4. 33.83	13.27	46184.903	
5. 0.1	0.	91392.517	
6. 23.06	4.78	65806.812	
7. 32.91	1.94	44749.266	
8. 38.78	9.51	44376.81	
9. 35.48	3.23	27878.982	
10. 32.81	5.3	49971.018	
11. 25.37	9.2	29795.311	
12. 12.19	16.11	67460.256	
13. 26.18	2.84	32675.84	
14. 29.78	- 0.76	51674.097	
15. 31.42	3.52	52563.982	
16. 39.05	8.93	38581.886	
17. 14.71	8.15	64862.913	
18. 29.44	15.79	38376.685	
19. 1.66	0.25	51709.846	
20. 32.75	1.35	46363.19	
21. 39.42	0.51	43669.29	
22. 26.31	1.76	24254.249	
23. 43.56	7.39	45192.3	
24. 18.62	8.51	59005.988	
25. 24.94	4.44	46346.641	
26. 27.3	11.01	55765.059	
27. 10.39	13.02	86247.108	
28. 1.93	0.39	95307.675	
29. 10.88	4.31	63861.97	
30. 0.82	- 0.09	121884.83	
31. 0.06	4.34	125617.11	
32. 15.32	0.11	22411.412	

Aggregated reduction of top 1% of peaks from battery leveling / EV Shifting:

9.53 / 3.39 (percent)

Appendix F
PV/Battery Leveling and EV/Air1 Leveling Results Top 1% of Peaks Using Homes with Higher Use/Grid Ratio

-->exec('/Users/Berthaskell/Desktop/Sci Lab Modelling/Adaptive threshold .sce', -1)

Run	Reduct_Lev	Reduct_Shift	Total Use	Ave Use	Ave Grid	Use/Grid
1.	18.86	19.16	46197.907	1.3187345	0.8413279	0.6379812
2.	14.75	0.77	37654.621	1.0748636	0.5638569	0.5245846
3.	7.15	13.27	46184.903	1.3183633	0.8215144	0.6231321
4.	3.29	4.78	65806.812	1.8784772	0.9769406	0.5200705
5.	6.06	1.94	44749.266	1.2773826	0.5245135	0.4106158
6.	15.35	9.51	44376.81	1.2667507	0.9210278	0.7270790
7.	23.77	3.23	27878.982	0.7958147	0.6066519	0.7623029
8.	10.16	5.3	49971.018	1.4264392	0.9809115	0.6876644
9.	36.79	9.2	29795.311	0.8505170	0.5749464	0.6759964
10.	18.67	2.84	32675.84	0.9327426	0.4972865	0.5331444
11.	6.14	- 0.76	51674.097	1.4750541	0.7163613	0.4856509
12.	9.44	3.52	52563.982	1.5004562	0.8351112	0.5565716
13.	13.28	8.93	38581.886	1.1013327	0.5920254	0.5375537
14.	41.11	15.79	38376.685	1.0954751	0.8571267	0.7824246
15.	9.37	1.35	46363.19	1.3234526	0.6853915	0.5178814
16.	12.13	0.51	43669.29	1.2465543	0.918709	0.7369988
17.	26.71	1.76	24254.249	0.6923455	0.4447118	0.6423264
18.	13.94	7.39	45192.3	1.2900291	0.7812431	0.6056012
19.	6.22	4.44	46346.641	1.3229801	0.5647354	0.4268661
20.	5.76	11.01	55765.059	1.591832	0.9519434	0.5980175

Aggregated reduction of top 1% of peaks from battery leveling / EV Shifting:

19.63   3.01