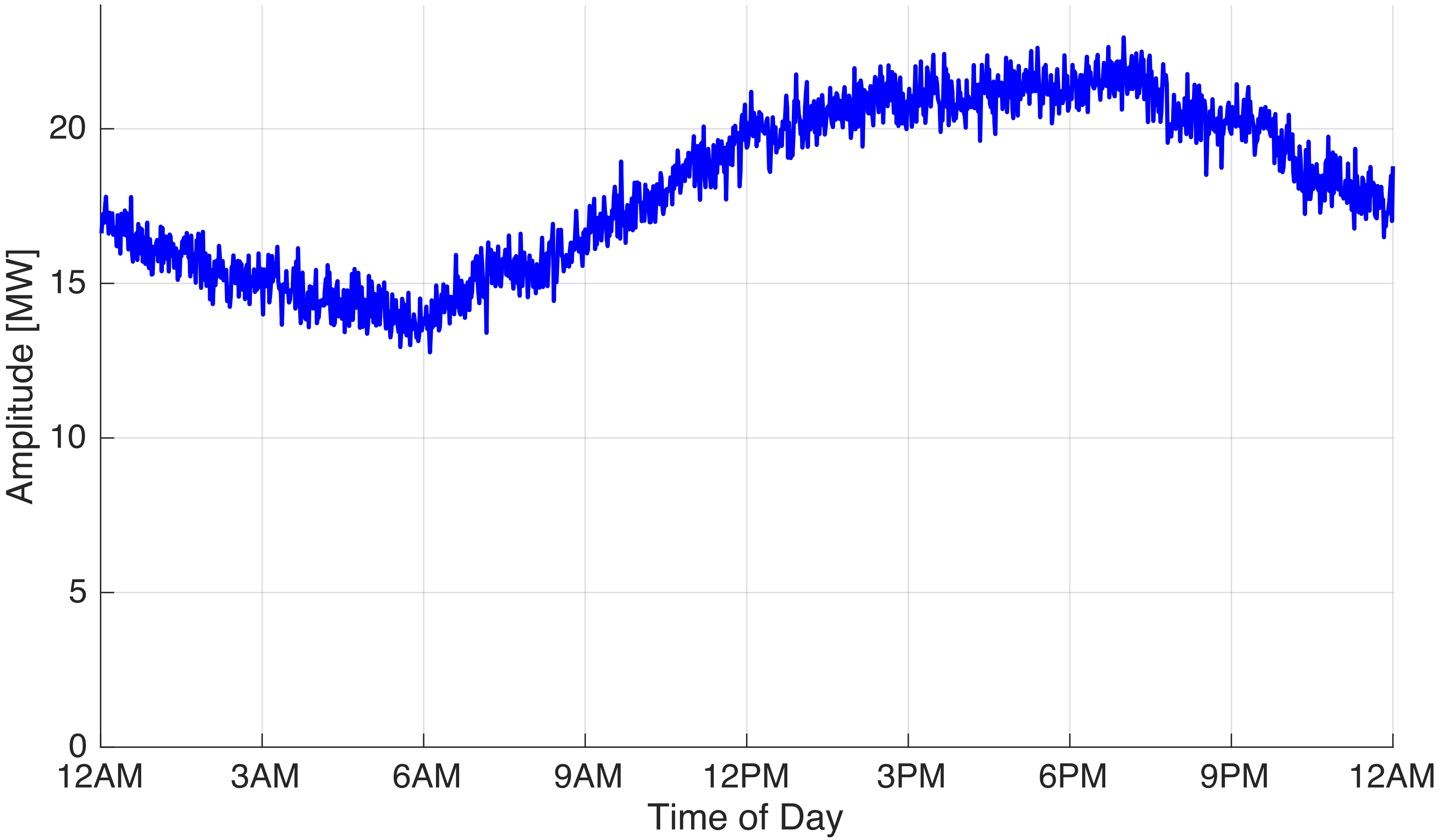
# Inferring the behavior of distributed flexible electric loads

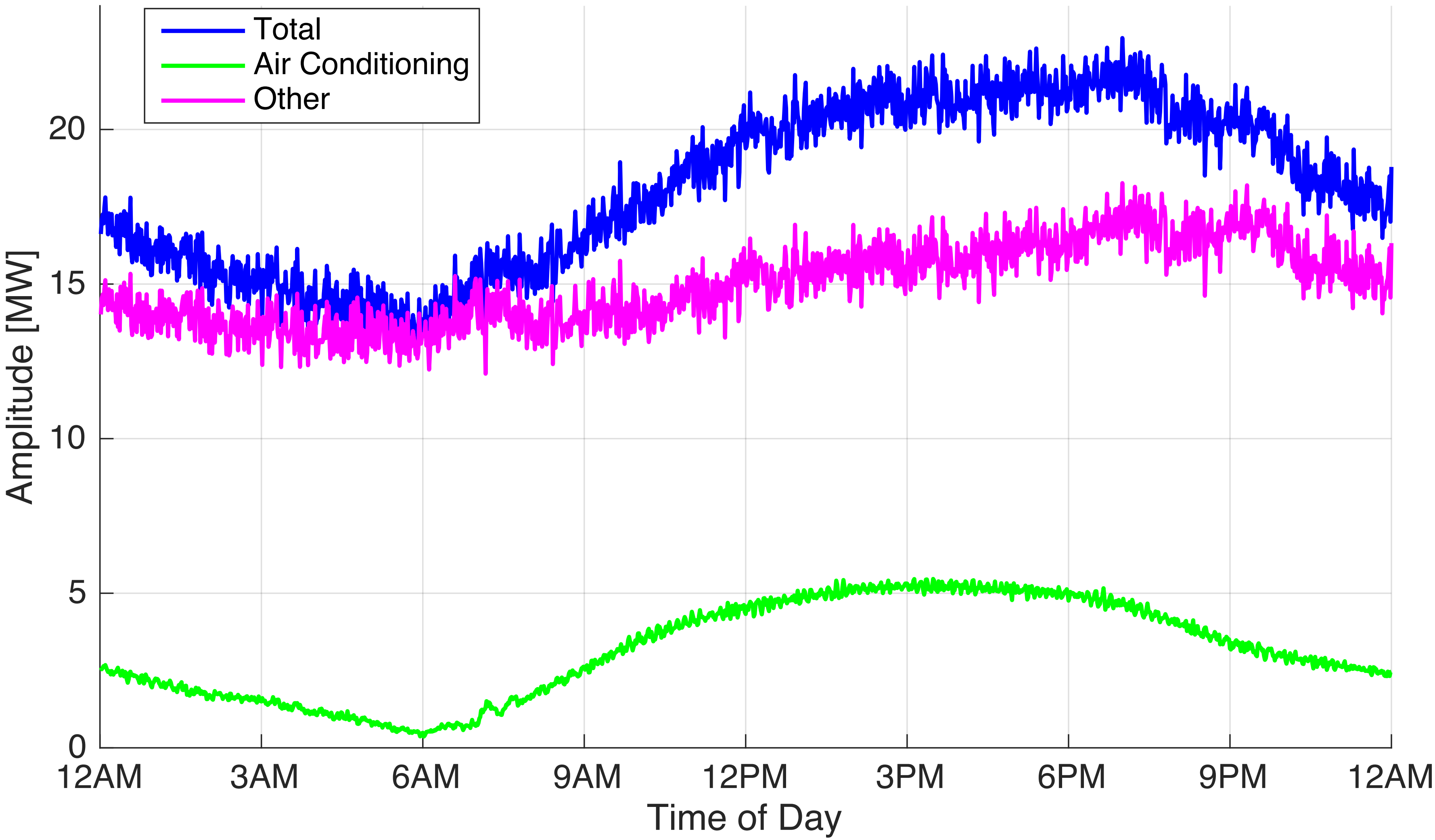
Greg Ledva, Laura Balzano, & Johanna Mathieu
Department of Electrical Engineering & Computer Science
University of Michigan

Funded by NSF (ECCS-1508943)

# Load Served by a Substation



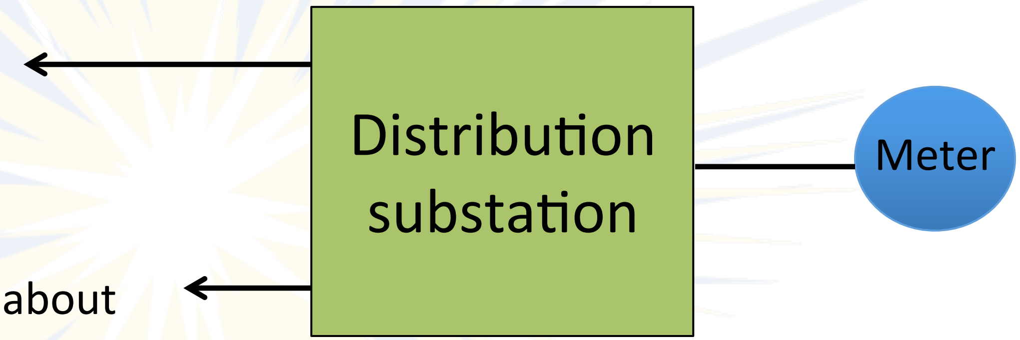
# Load Served by a Substation



# Disaggregating substation load data

Power consumption of all the loads/generators we care about (e.g., Air Conditioning)

Power consumption of all the loads/generators we DON'T care about ("Other loads")



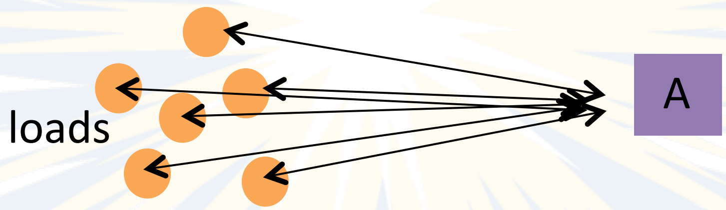

In this talk, we use measurements of real power only.

We could consider additional measurements (reactive power, voltage, etc.) from multiple meters at different points in the distribution network.

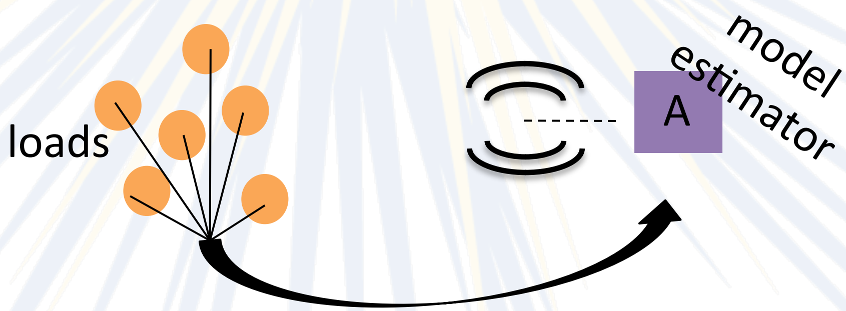
# Why disaggregate the substation load?

- Load coordination feedback

***Bi-directional communication***

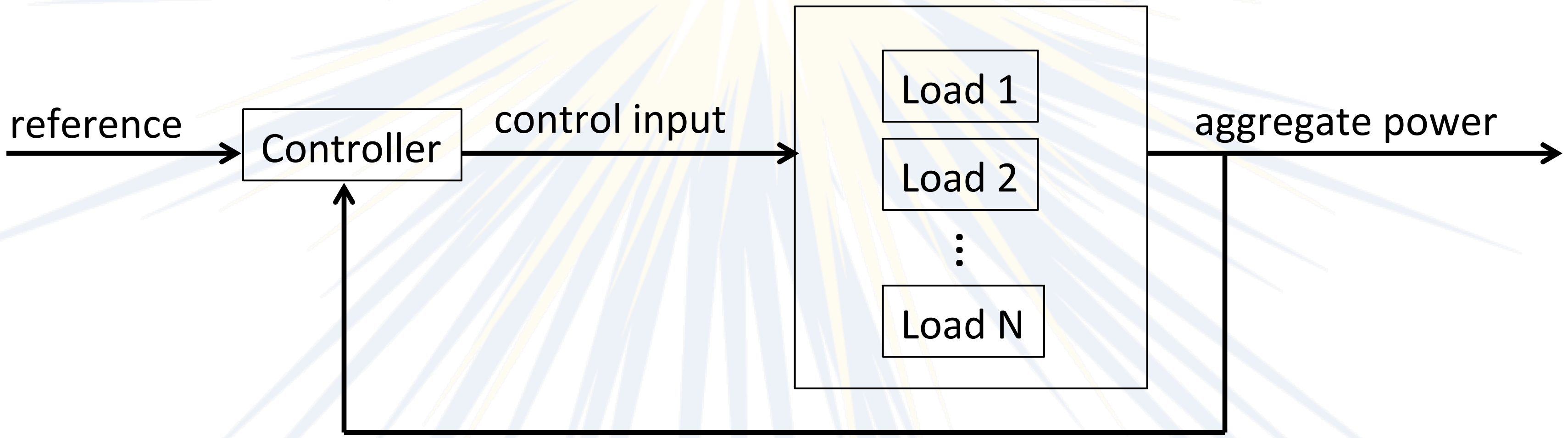



***Broadcast control, aggregate measurements***



# Why disaggregate the substation load?

- Load coordination feedback
  - (noisy) measurements of the aggregate power of coordinated loads are assumed in Mathieu et al. 2013; Can Kara et al. 2013; Bušić and Meyn 2016; Callaway 2009; …



# Why disaggregate the substation load?

Additional uses in demand response…
- Load aggregator bidding
- Demand response event signaling (when/how much)

Beyond demand response…
- Energy efficiency via conservation voltage reduction
  - Disaggregate by load type
- Contingency planning
  - Disaggregate motor loads
- Reserve planning
  - Disaggregate PV production

# Connections to other problems

- Non-intrusive load monitoring (NILM) [Hart 2010; Ziefman and Roth 2011; Berges et al. 2009; Zoha et al. 2012; Dong, Sastry, et al. 2014; …]
- Energy disaggregation [Wytock & Kolter 2013; Kolter and Jaakkola 2012; Dong, Satsry, et al. 2013; Kim et al. 2010 …]

Problem: Infer individual load behavior from a single power measurement (usually) sampled at high frequency (10kHz-1MHz) from the household main

Solution approaches: offline algorithms including change detection, supervised learning, unsupervised learning

# Key differences

- We assume measurements at the substation, not the household
- We infer aggregate load (e.g., all air conditioning load), not individual load behavior
- We solve the problem online, not offline
- We use lower frequency measurements (e.g., taken every second to minute)
- In some cases, we may get to be "intrusive," but not in this talk!

# Possible Methods

- State estimation
  - Linear techniques require linear system models
  - Nonlinear techniques can be computationally demanding
- Online learning
  - Data-driven, model-free
- Hybrid approach: Dynamic Mirror Descent
  [Hall & Willet 2015]
  - Admits dynamic models of arbitrary forms
  - Optimization-based method to choose a weighted combination of the estimates of a collection of models

# Outline

- Dynamic Mirror Descent
- Problem setting: Plant data/models
- Algorithm Models
- Results
- Next steps

# Dynamic Mirror Descent

- Mirror Descent: online algorithm to estimate a fixed state
- Dynamic Mirror Descent: online algorithm to estimate a dynamic state using a *collection of models* [Hall & Willet, "Online Convex Optimization in Dynamic Environments," IEEE Journal of Selected Topics in Signal Processing 2015]

1. Compute the error between the model predictions and the measured data (i.e., loss function $\ell_t(\widehat{\theta}_t^i, y_t)$)
2. Update the state in the direction of the negative gradient of the loss function

$$\widetilde{\theta}_t^i = \arg\min_{\theta \in \Theta} \eta_t \left\langle \nabla \ell_t(\widehat{\theta}_t^i, y_t),\ \theta \right\rangle + D\left(\theta \| \widehat{\theta}_t^i\right)$$

# Dynamic Mirror Descent

3. Use the estimated states to evaluate the models for the next time step

$$\widehat{\theta}^i_{t+1} = \Phi^i_t(\widetilde{\theta}^i_t)$$

4. Compute a weighted version of the estimates

$$\widehat{\theta}_{t+1} = \sum_{i=1}^{N^{\mathrm{mdl}}} w^i_{t+1} \widehat{\theta}^i_{t+1}.$$

5. Update the model weights

$$w^i_{t+1} = \frac{\lambda}{N^{\mathrm{mdl}}} + (1-\lambda) \frac{w^i_t \exp\left(-\eta^r \, \ell_t\left(\widehat{\theta}^i_t, y_t\right)\right)}{\sum_{j=1}^{N^{\mathrm{mdl}}} w^j_t \exp\left(-\eta^r \, \ell_t\left(\widehat{\theta}^j_t, y_t\right)\right)}$$

# Algorithmic guarantees

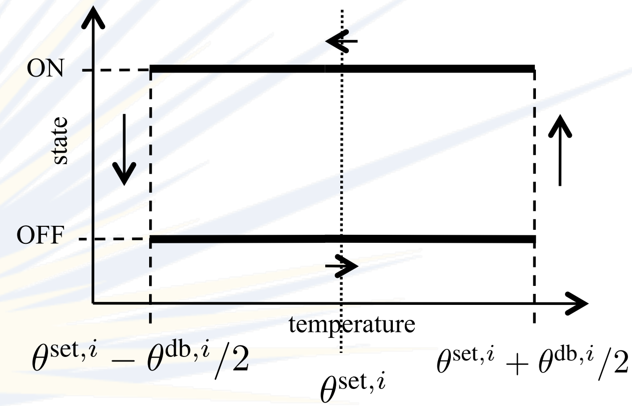
- Regret: performance with respect to a comparator $\boldsymbol{\theta}_T$

$$R_T(\boldsymbol{\theta}_T) \triangleq \sum_{t=1}^{T} \ell_t(\widehat{\theta}_t) - \sum_{t=1}^{T} \ell_t(\theta_t).$$

- Often the comparator is the performance of a batch algorithm
- Hall and Willet derive bounds on the regret and show that for many classes of comparators regret scales sublinearly in $T$

# Problem Setting: Plant Data/Models

- **Air conditioners:** 1000 equivalent thermal parameters (ETP) models, i.e., three-state hybrid models [Sonderegger 1978]




$$\theta^i_{t+1} = A^i\theta^i_t + B^i m^i_t + E^i d^i_t$$

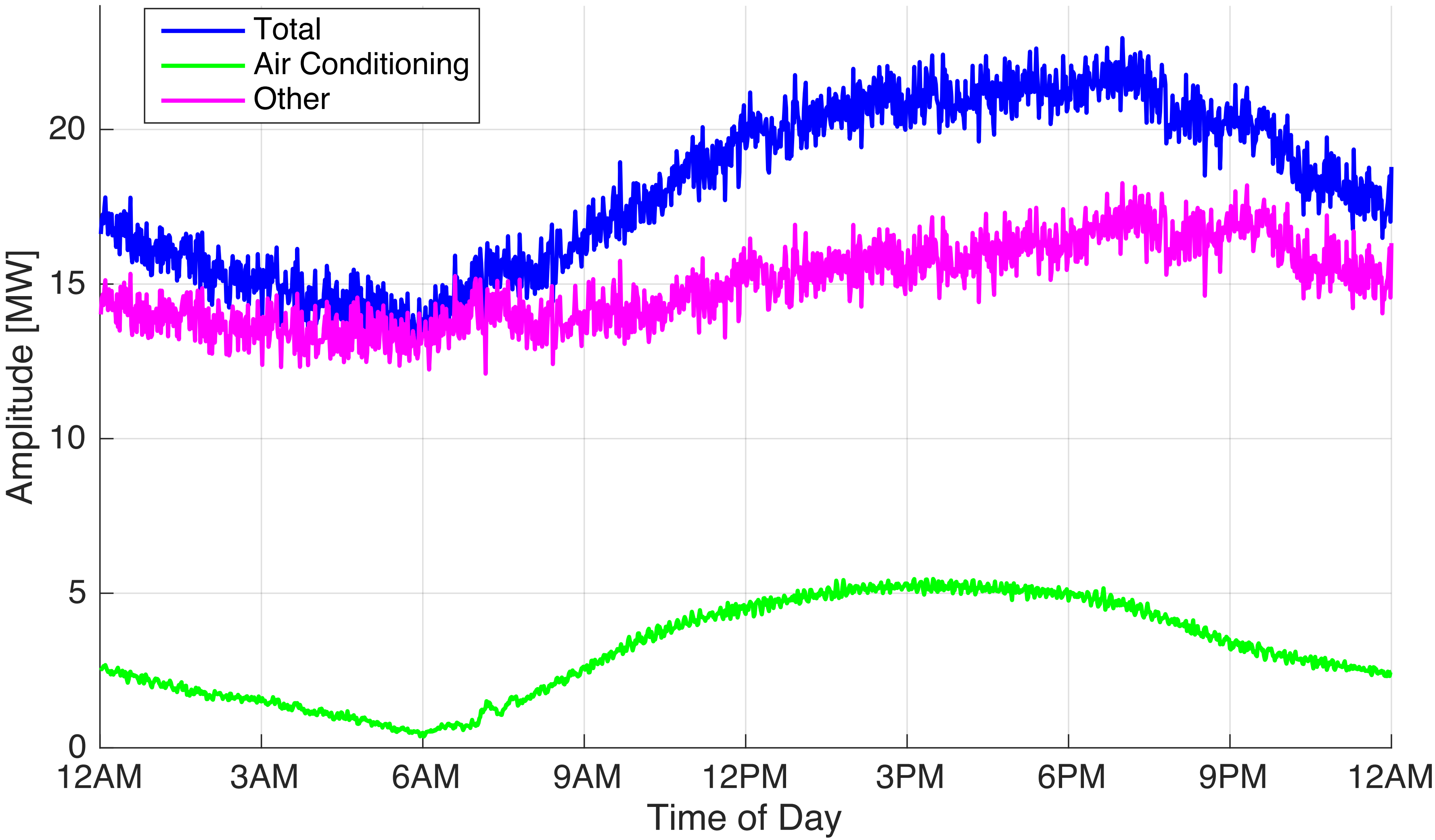
$$m^i_{t+1} = \begin{cases} 0 & \text{if } \theta^{a,i}_{t+1} < \theta^{\text{set},i} - \theta^{\text{db},i}/2 \\ 1 & \text{if } \theta^{a,i}_{t+1} > \theta^{\text{set},i} + \theta^{\text{db},i}/2 \\ m^i_t & \text{otherwise} \end{cases}$$

$$P^i_t = (|Q^{\text{h},i}| \ m^i_t)/\eta^i$$

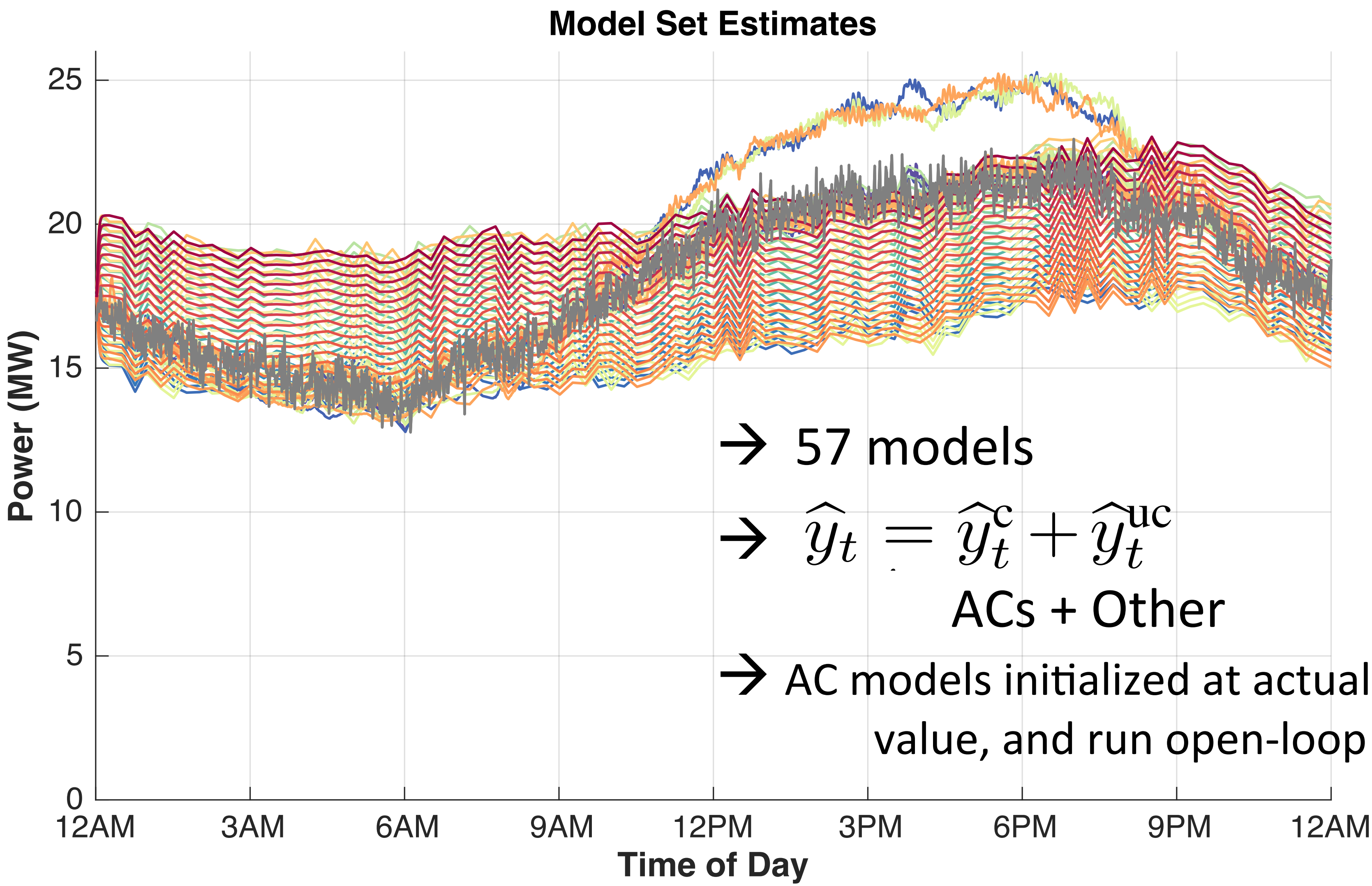
where $\theta^i_t = \left[\theta^{a,i}_t \quad \theta^{m,i}_t\right]^T$

- **Other loads:** data from Pecan Street Inc. Dataport

# Problem Setting: Plant Data/Models



# Algorithm Models

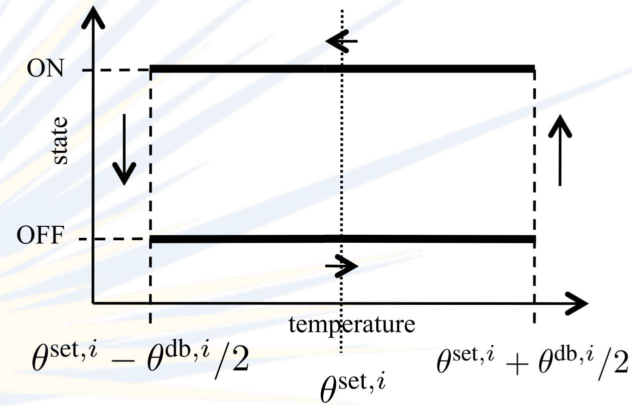



- 57 models
- $\widehat{y}_t = \widehat{y}_t^{\mathrm{c}} + \widehat{y}_t^{\mathrm{uc}}$
  ACs + Other
- AC models initialized at actual value, and run open-loop

# Algorithm Models: Air conditioners - 1

## 1000 two-state hybrid models

[Chong & Debs 1979; Ihara & Schweppe 1981]



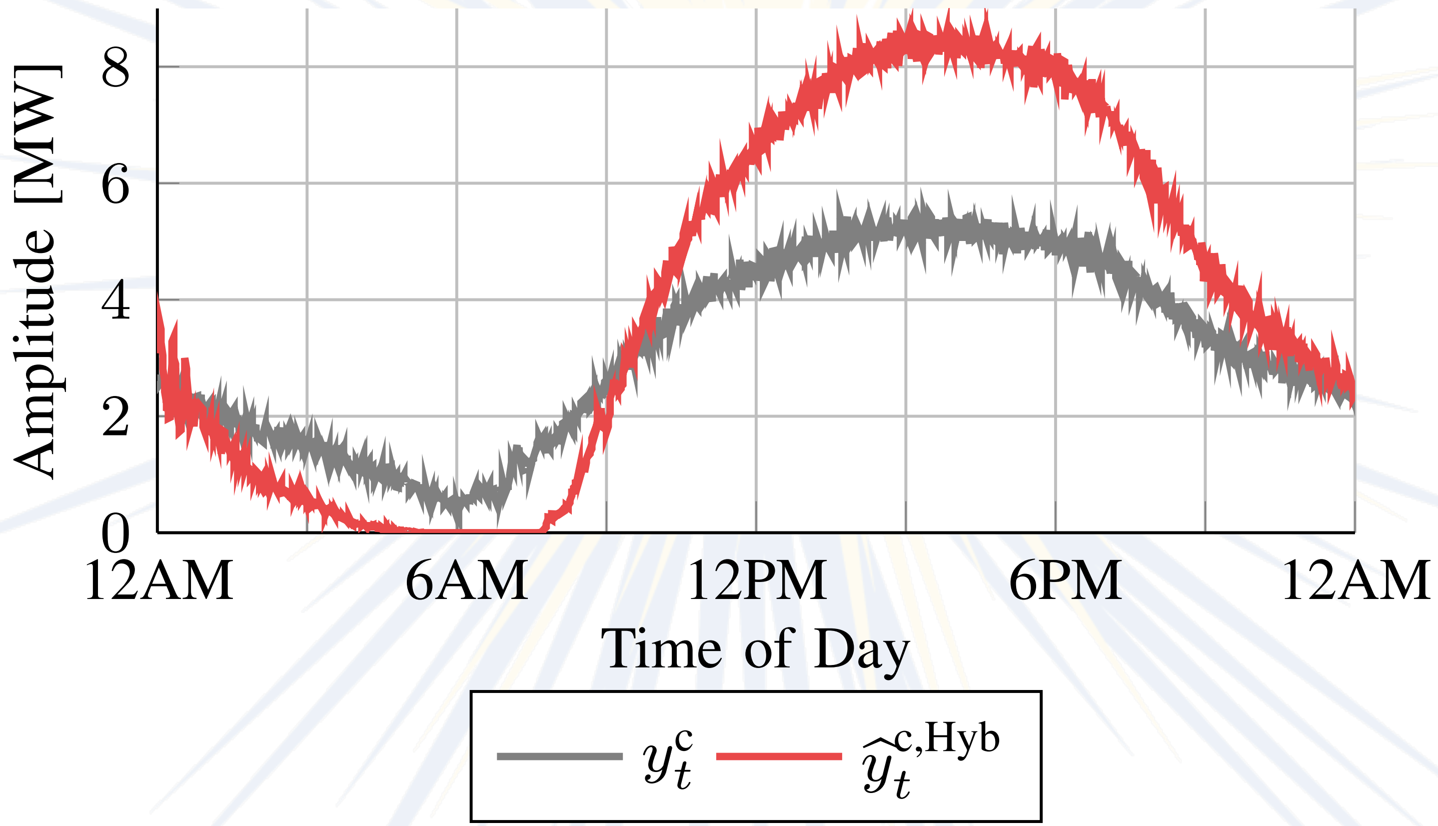

$$\theta^i_{t+1} = A^i\theta^i_t + B^i m^i_t + E^i d^i_t$$

$$m^i_{t+1} = \begin{cases} 0 & \text{if } \theta^{a,i}_{t+1} < \theta^{\text{set},i} - \theta^{\text{db},i}/2 \\ 1 & \text{if } \theta^{a,i}_{t+1} > \theta^{\text{set},i} + \theta^{\text{db},i}/2 \\ m^i_t & \text{otherwise} \end{cases}$$

$$P^i_t = (|Q^{\text{h},i}|\ m^i_t)/\eta^i$$

where $\boxed{\theta^i_t = \theta^{a,i}_t}$

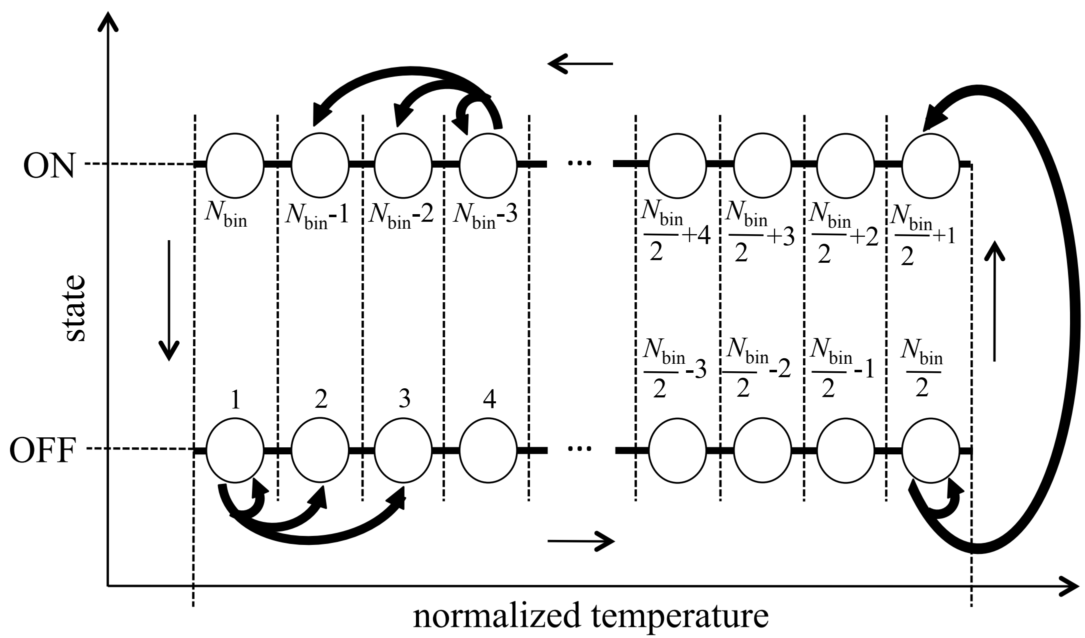
# Algorithm Models: Air conditioners - 1



# Algorithm Models: Air conditioners - 2

Linear Time Invariant (LTI) aggregate system models
[Mathieu et al. 2013]

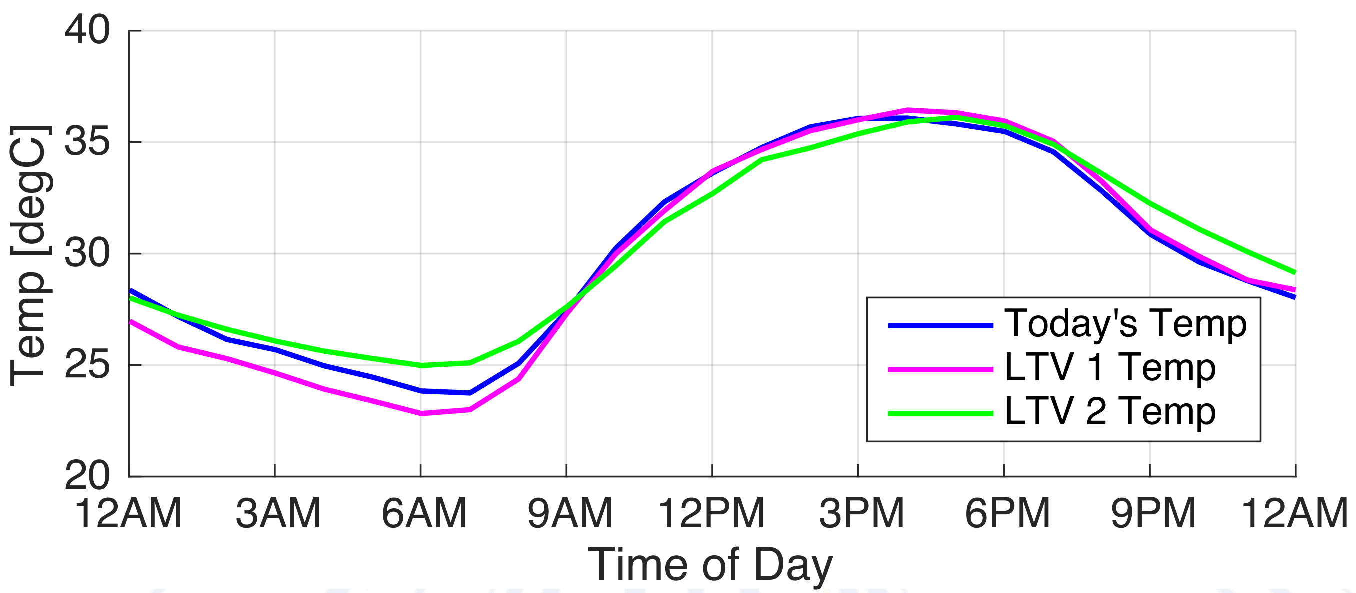
$$x_{t+1}^{i} = A^{i}\, x_{t}^{i} \qquad i \in \mathbb{N}^{\text{temps}}$$
$$\widehat{y}_{t}^{\text{c,LTI},i} = C^{i}\, x_{t}^{i} \qquad i \in \mathbb{N}^{\text{temps}}.$$



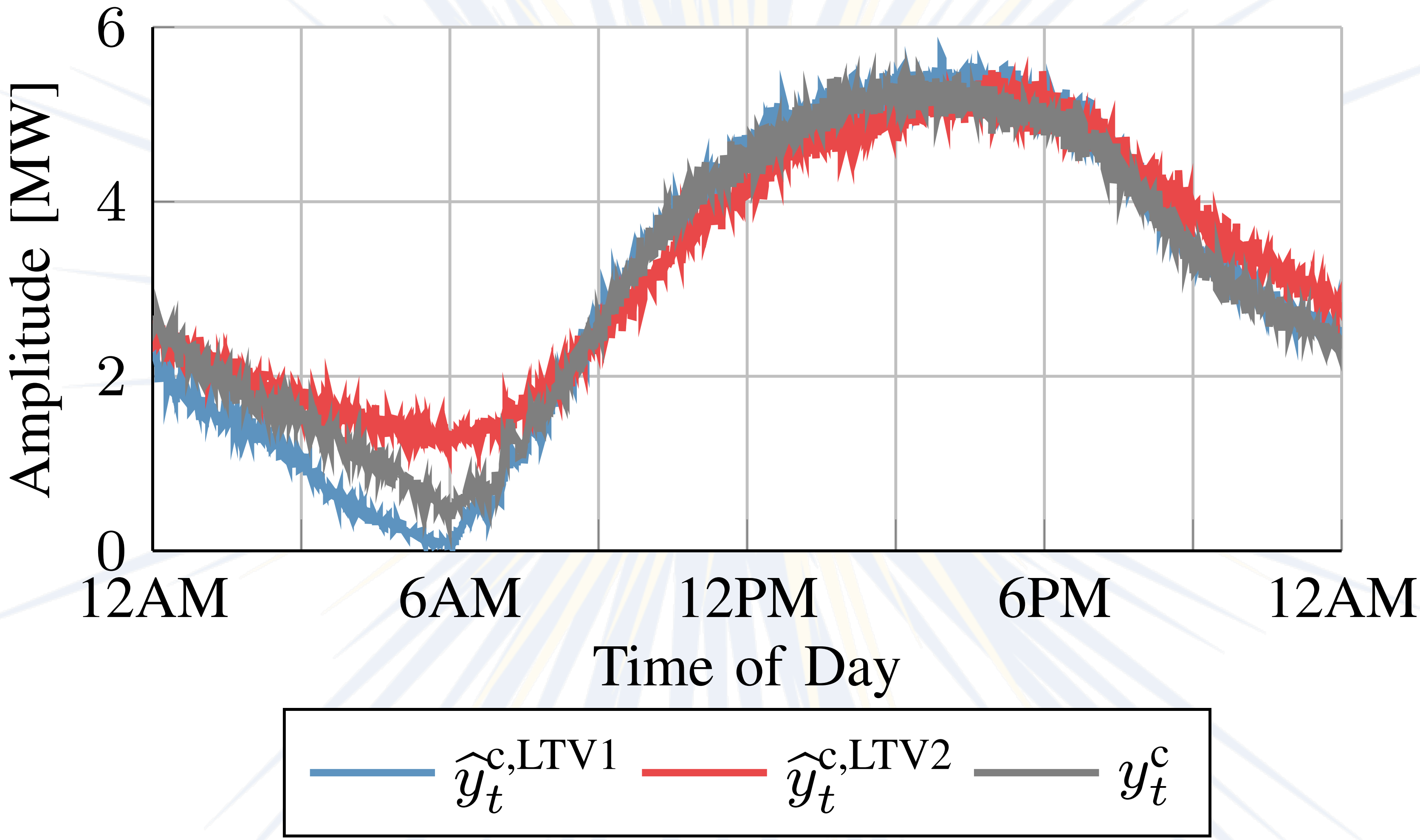
# Algorithm Models: Air conditioners - 3

Linear Time Varying (LTV) aggregate system models
[Mathieu et al. 2015]

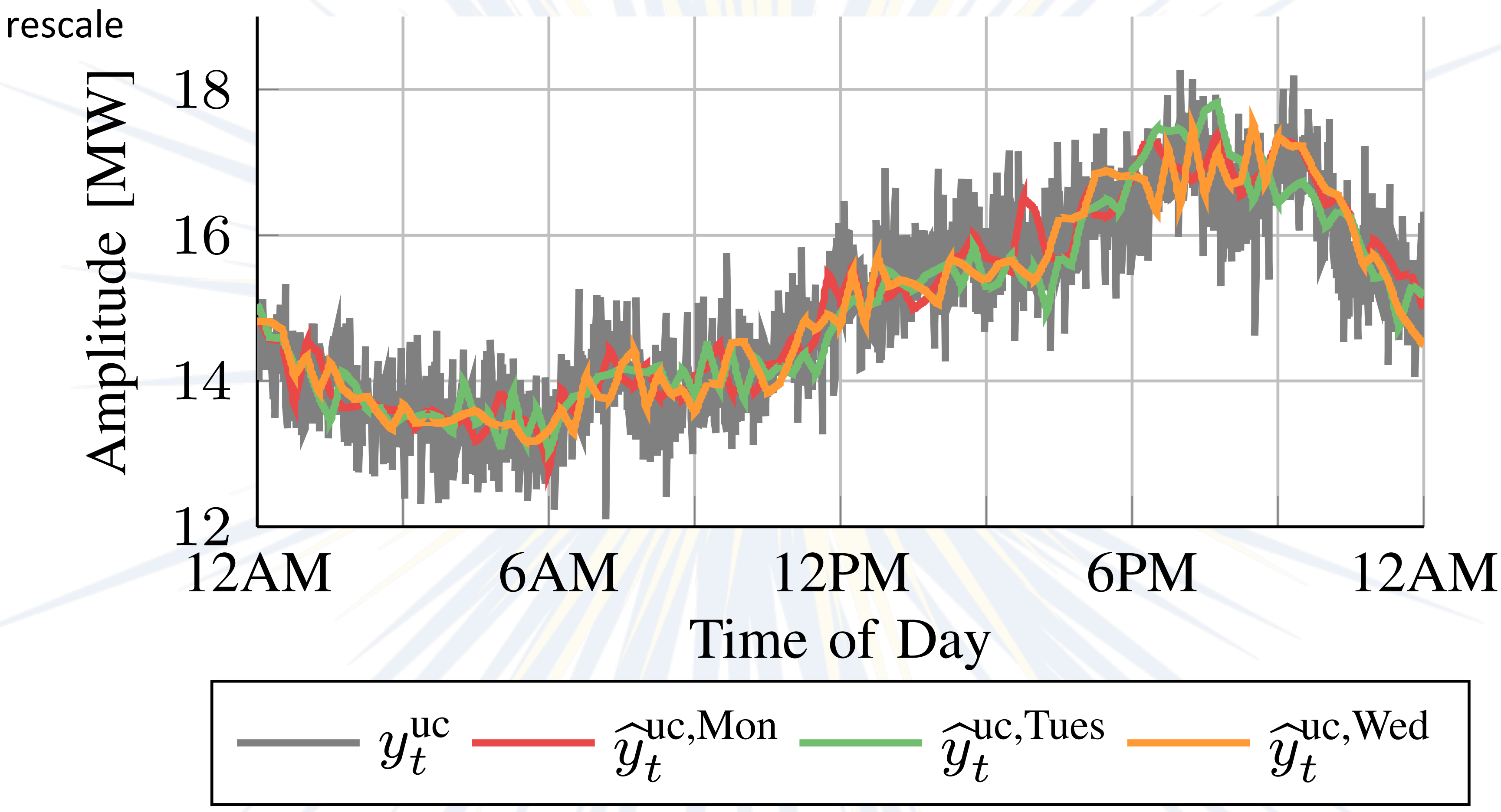
$$x_{t+1} = A_t\, x_t$$
$$\widehat{y}_t^{\,\mathrm{c,LTV}} = C_t\, x_t.$$



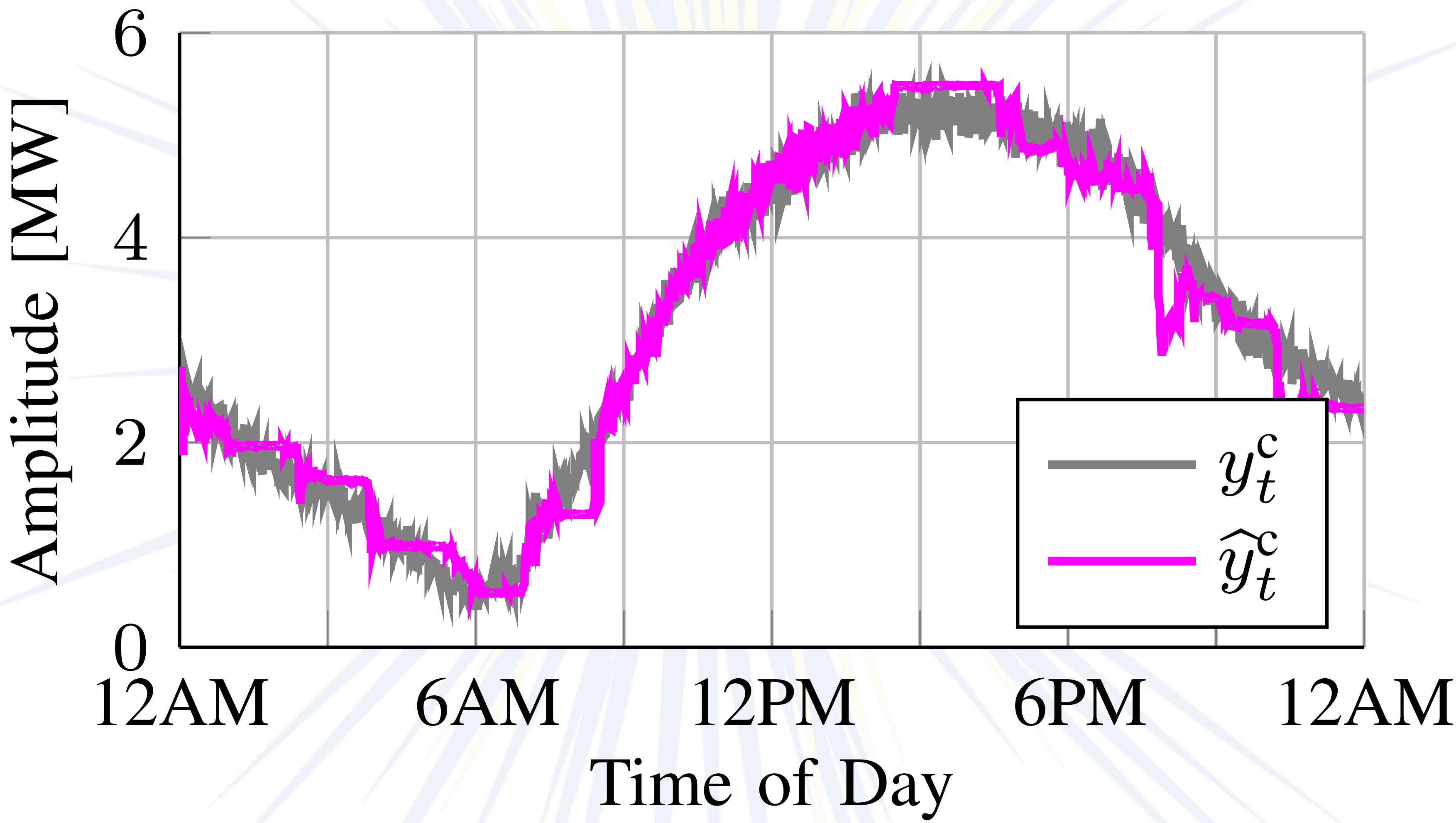
# Algorithm Models: Air conditioners - 3



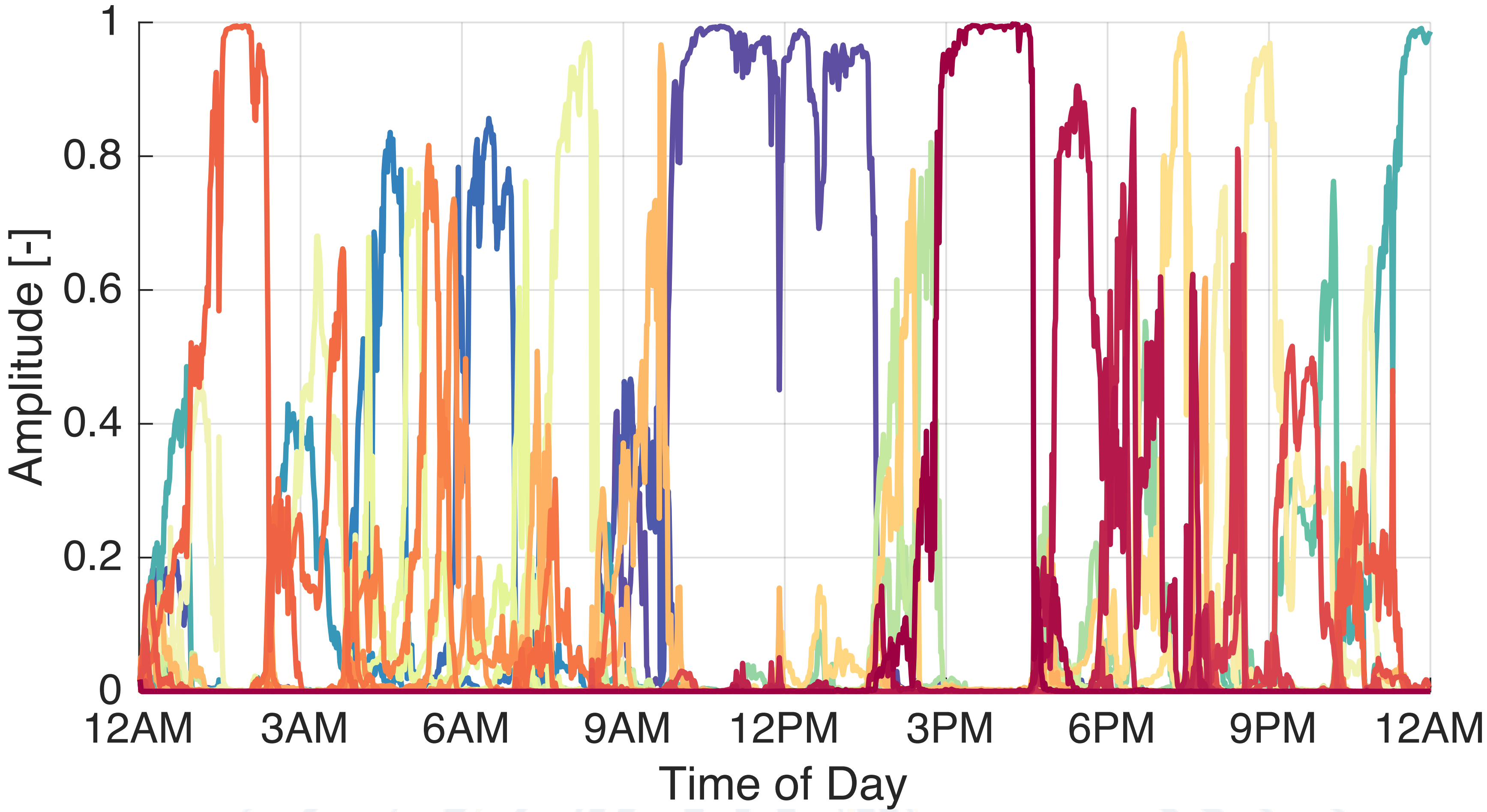
# Algorithm Models: Other loads



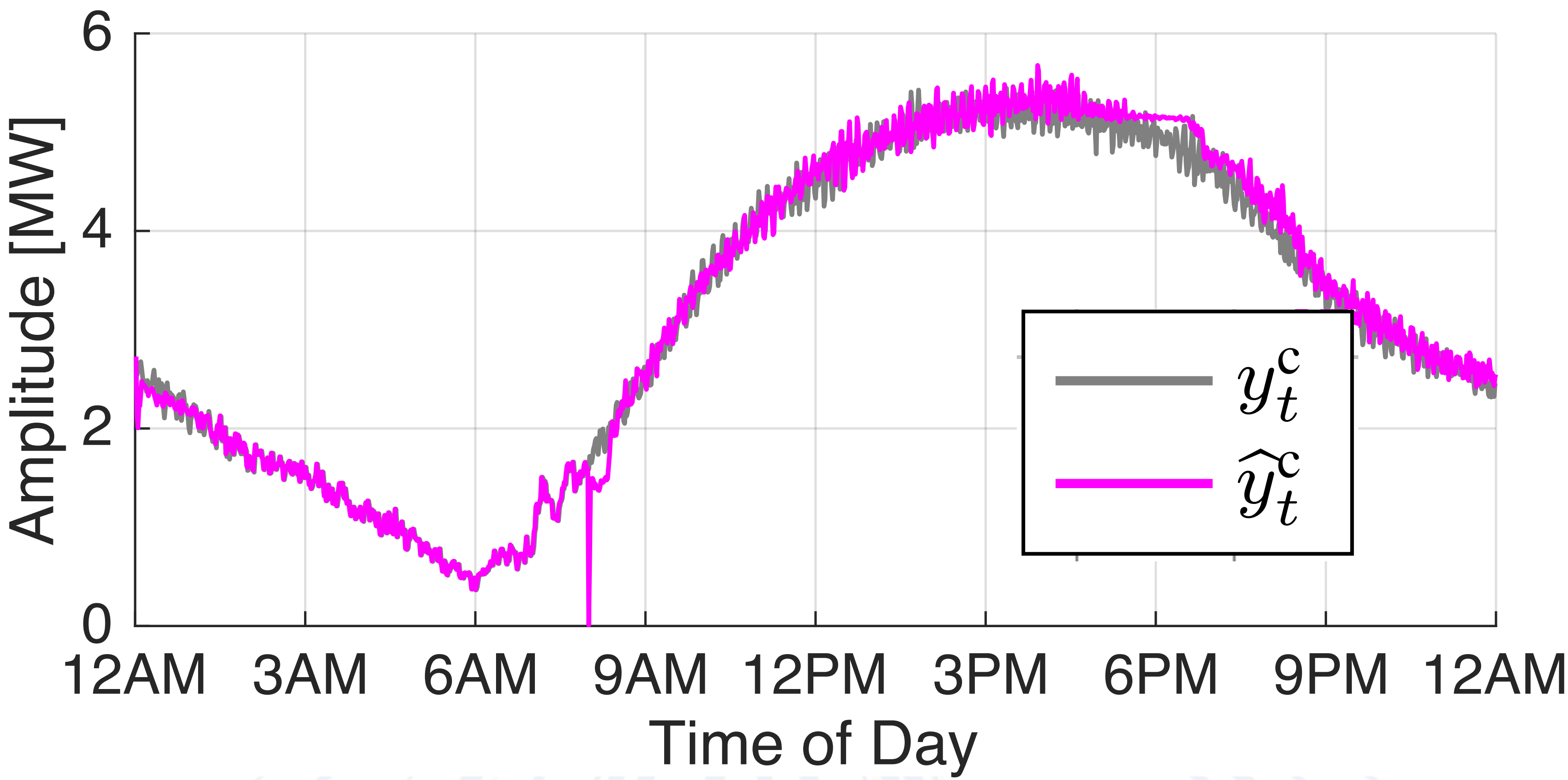
# Prediction Results



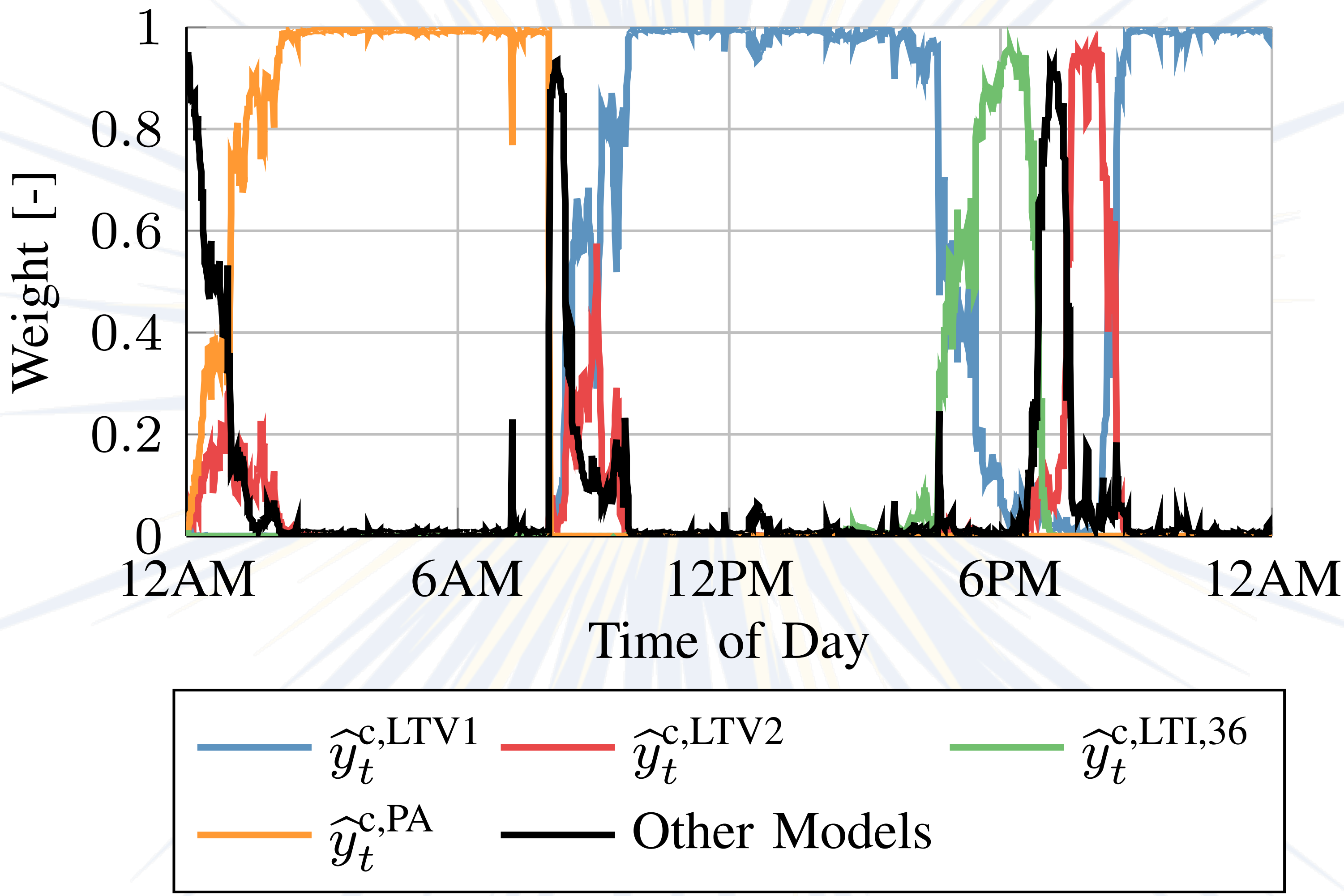
# Weightings:
## each color is a different model



# Prediction Results: Better Models

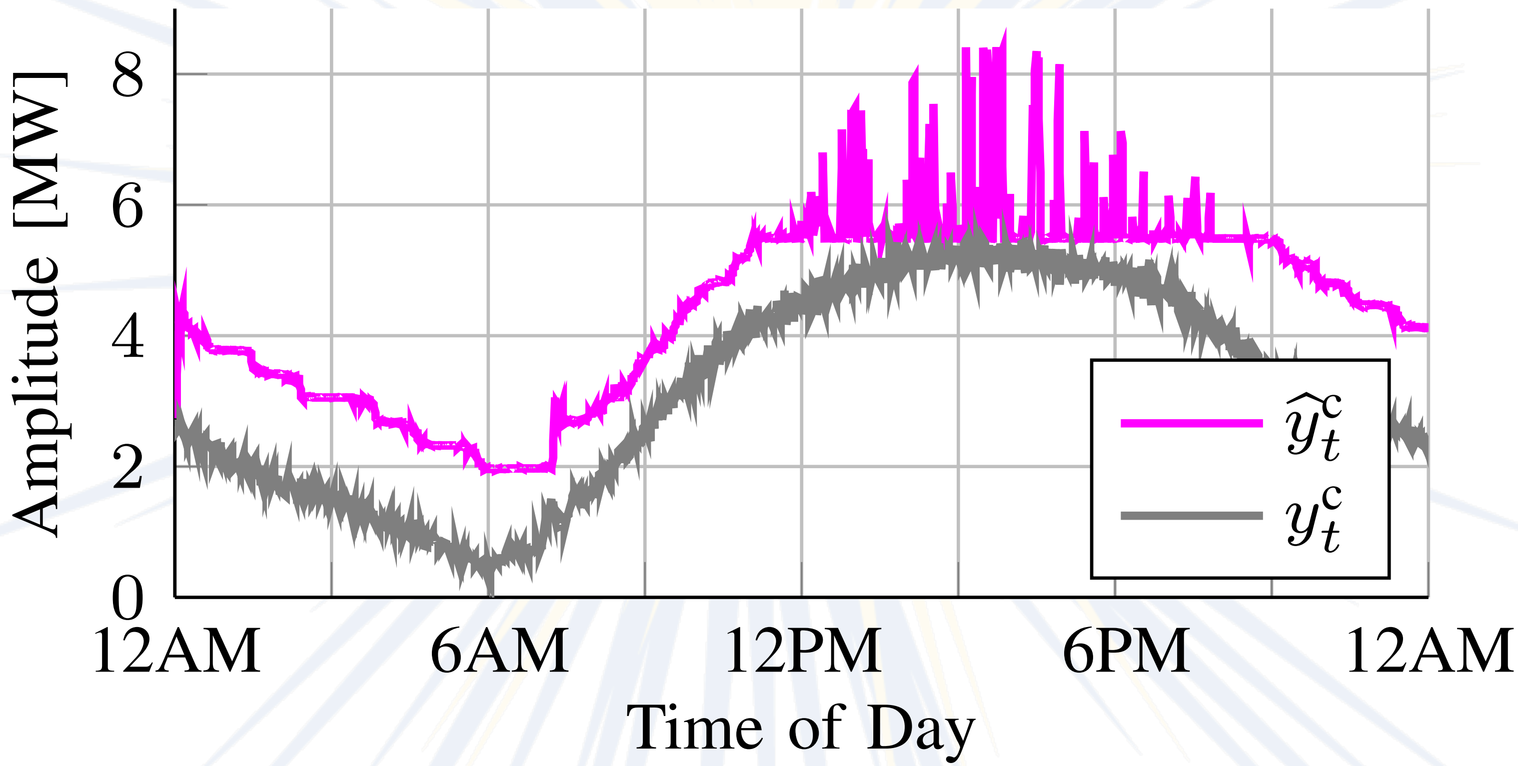


# Weightings: Better Models



# Prediction Results: Bad Models

- All "other load models" are too low.

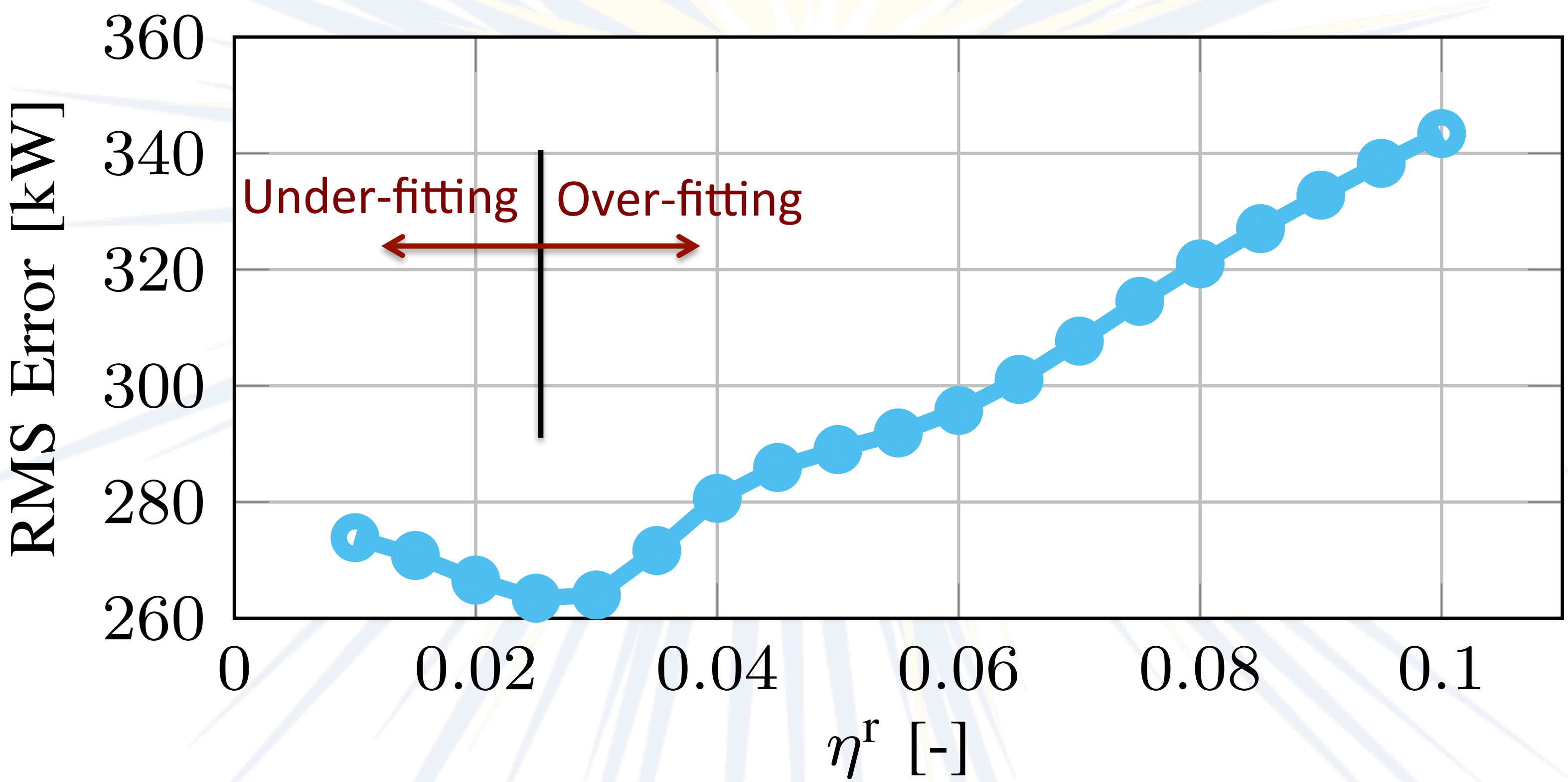


# Results: Summary

| Case | RMS Error (kW) |
|---|---|
| **Benchmark:** Use current outdoor temperature, LTI models, and interpolation to predict | 738 |
| **DMD Case 1**: Includes every combination of aggregate air conditioner model and “other load model” | 264 |
| **DMD Case 2:** Case 1 models plus a smoothed version of the actual “other loads” | 211 |
| **DMD Case 3:** Case 2 models plus more accurate models of the aggregate air conditioning load over time periods where the other models are less accurate | 175 |
| **DMD Case 4**: Includes “other load models” that underestimate the “other load” | 1392 |

# Results: Varying Algorithm Parameters

Recall: $$w_{t+1}^{i} = \frac{\lambda}{N^{\text{mdl}}} + (1-\lambda)\frac{w_t^i \exp\left(-\eta^r \, \ell_t\left(\widehat{\theta}_t^i, y_t\right)\right)}{\sum_{j=1}^{N^{\text{mdl}}} w_t^j \exp\left(-\eta^r \, \ell_t\left(\widehat{\theta}_t^j, y_t\right)\right)}$$



# Next steps

- Investigate more realistic settings
- Develop better load models
- Improve the algorithm, e.g., alternative weighting functions
- Investigate identifiability
- Incorporate additional measurements (reactive power, voltage) into the approach

# Conclusions

- Dynamic Mirror Descent (DMD) enables us to solve the substation disaggregation problem leveraging dynamical models of arbitrary form
- DMD can work well (on simple examples); however, it is easy to find instances where it does not work well

More details: Ledva, Balzano, and Mathieu, "Inferring the Behavior of Distributed Energy Resources with Online Learning," Allerton 2015.