Online Optimal Power Flow

Yujie Tang
Steven Low

Caltech

Krishnamurthy Dvijotham

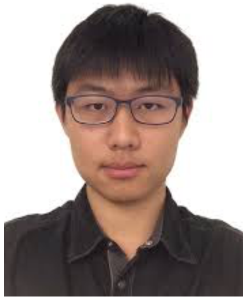
Pacific Northwest
NATIONAL LABORATORY

January 2017

Online OPF

Gan (FB)

Dvijotham (PNNL)



Tang (Caltech)

OPF relaxation


Bose (UIUC)


Chandy


Farivar


Gan (FB)


Lavaei (UCB)

Online OPF


Gan (FB)


Dvijotham (PNNL)


Tang

Dynamics


Bialek (Skoltech)


Li (Harvard)


Mallada (JHU)


Topcu (Austin)

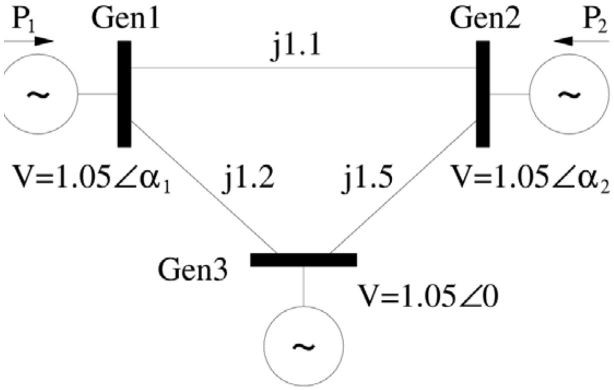

Zhao (NREL)

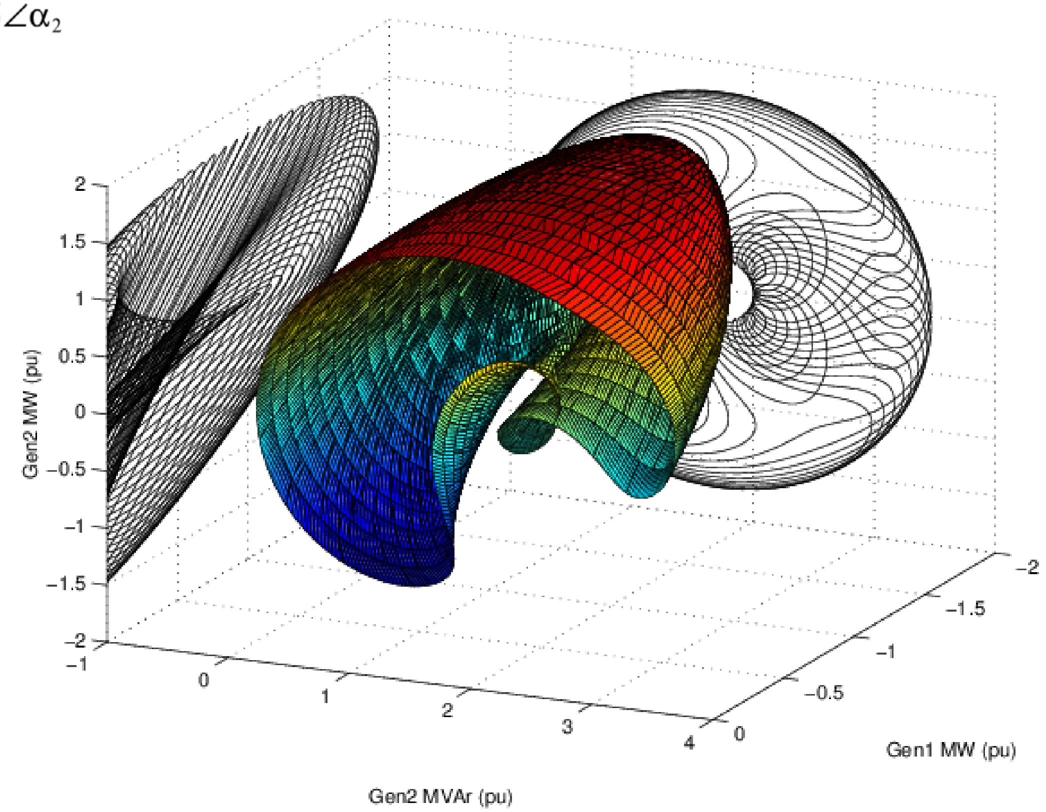
NSF arpa·e SOUTHERN CALIFORNIA EDISON RESNICKINSTITUTE science + energy + sustainability Skoltech Skolkovo Institute of Science and Technology

Optimal power flow (OPF)

OPF problem underlies numerous applications

- nonlinearity of power flow equations ➔ nonconvexity





Ian Hiskens, Michigan

How to deal with nonconvexity of power flows?

Two ideas

1. exact semidefinite relaxation

Tutorial:
L, Convex relaxation of OPF, 2014
http://netlab.caltech.edu

How to deal with nonconvexity of power flows?

Two ideas

1. exact semidefinite relaxation
2. use grid as implicit power flow solver

Relaxations of OPF

OPF: $$\min_{x \in \mathbf{X}} f(x)$$

relaxation: $$\min_{\hat{x} \in \mathbf{X}^+} f(\hat{x})$$

But traditional algorithms are all offline …
… unsuitable for real-time optimization of
network of distributed energy resources

Key message

Large network of DERs

- Real-time optimization at scale
- Computational challenge: power flow solution

Online optimization [feedback control]

- Network computes power flow solutions in real time at scale for free
- Exploit it for our optimization/control
- Naturally adapts to evolving network conditions

Examples

- Slow timescale: OPF
- Fast timescale: frequency control

Key message

Large network of DERs

- Real-time optimization at scale
- Computational challenge: power flow solution

Online optimization [feedback control]

- Network computes power flow solutions in real time at scale for free
- Exploit it for our optimization/control
- Naturally adapts to evolving network conditions

Examples

- **Slow timescale: OPF**
- Fast timescale: frequency control

Prior work

Gan and Low, JSAC 2016
Dall’Anese, Dhople and Giannakis, TPS 2016
Dall’Anese and Simonetto, TSG 2016
Arnold et al, TPS 2016

Prior work

Gan and Low, JSAC 2016

Dall’Anese, Dhople and Giannakis, TPS 2016

Dall’Anese and Simonetto, TSG 2016

Arnold et al, TPS 2016

Outline

Problem formulation

Online Newton method for OPF

Optimal power flow

$$\begin{array}{lll} \min & \operatorname{tr}\left(CVV^H\right) & \text{gen cost, power loss} \\ \text{over} & \left(V, s\right) & \\ \text{subject to} & \underline{s}_j \le s_j \le \overline{s}_j & \underline{V}_j \le |V_j| \le \overline{V}_j \\ & s_j = \operatorname{tr}\left(Y_j^H V V^H\right) & \text{power flow equation} \end{array}$$

- Y_j^H describes network topology and impedances
- s_j is net power injection (generation) at node j
- "power balance at each node j" (Kirchhoff's law)

Optimal power flow

$$\begin{array}{lll} \text{min} & \mathrm{tr}\left(CVV^H\right) & \\ \text{over} & (V, s) & \\ \text{subject to} & \underline{s}_j \le s_j \le \overline{s}_j & \underline{V}_j \le |V_j| \le \overline{V}_j \\ & s_j = \mathrm{tr}\left(Y_j^H VV^H\right) & \end{array}$$

gen cost, power loss

power flow equation

nonconvex feasible set

- Y_j^H not Hermitian (nor positive semidefinite)
- C is positive semidefinite (and Hermitian)

nonconvex QCQP

OPF

$$\begin{aligned} \min \quad & c_0(y) + c(x) \\ \text{over} \quad & x,\ y \\ \text{s. t.} \quad & \end{aligned}$$

controllable devices

uncontrollable state

OPF

$$\begin{aligned} \min \quad & c_0(y) + c(x) \\ \text{over} \quad & x,\ y \\ \text{s.t.} \quad & F(x,y) = 0 \end{aligned}$$

power flow equations

OPF

$$\begin{aligned} \min \quad & c_0(y) + c(x) \\ \text{over} \quad & x,\ y \\ \text{s. t.} \quad & F(x,y) = 0 \\ & y \le \overline{y} \\ & x \in X := \left\{ \underline{x} \le x \le \overline{x} \right\} \end{aligned}$$

power flow equations

operational constraints

capacity limits

$$\text{Assume:} \quad \frac{\partial F}{\partial y} \ne 0 \qquad \Rightarrow \qquad y(x) \quad \text{over} \;\; X$$

OPF: eliminate y

$$\begin{aligned} \min_{x} \quad & c_0(y(x)) + c(x) \\ \text{s.t.} \quad & y(x) \le \overline{y} \\ & x \in X := \{\underline{x} \le x \le \overline{x}\} \end{aligned}$$

OPF: add barrier

$$\begin{array}{ll} \min\limits_{x} & c_0(y(x)) + c(x) \\ \text{s. t.} & y(x) \leq \overline{y} \\ & x \in X := \{\underline{x} \leq x \leq \overline{x}\} \end{array}$$

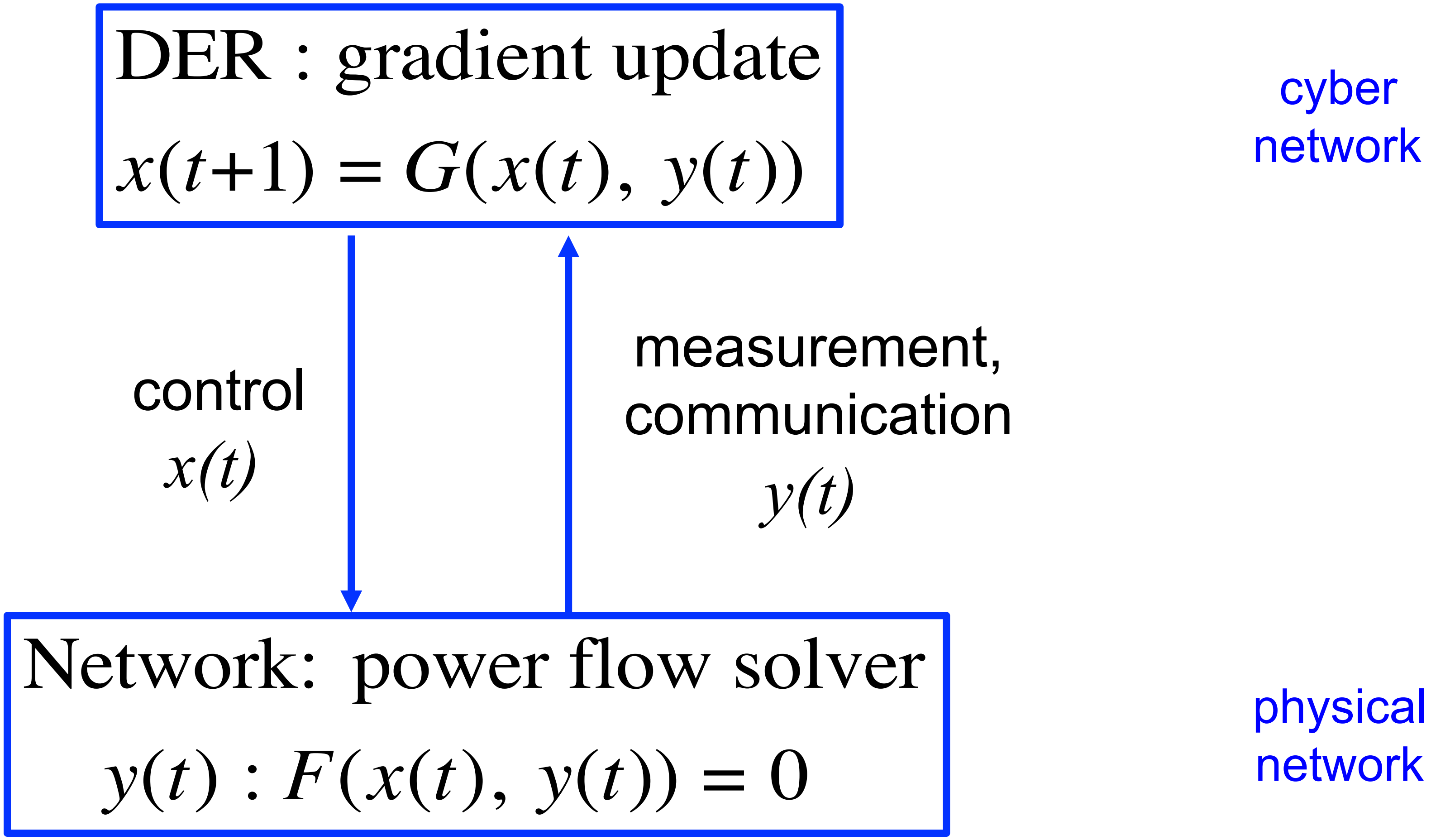
$$\big\downarrow$$

add barrier function
to remove
operational constraints

$$\boxed{\begin{array}{ll} \min & L(x, y(x);\ \mu) \\ \text{over} & x \in X \end{array}}$$

L: nonconvex

Online (feedback) perspective



DER : gradient update

$$x(t+1) = G(x(t),\ y(t))$$

cyber
network

control
$x(t)$

measurement,
communication
$y(t)$

Network: power flow solver

$$y(t) : F(x(t),\ y(t)) = 0$$

physical
network

- Explicitly exploits network as power flow solver
- Naturally tracks changing network conditions

Prior work

$$\boxed{\begin{array}{ll} \min & L(x, y(x);\ \mu) \\ \text{over} & x \in X \end{array}}$$

gradient projection algorithm:

$$x(t+1) = \left[x(t) - \eta \frac{\partial L}{\partial x}(t) \right]_X \qquad \text{active control}$$

$$y(t) = y(x(t)) \qquad \text{law of physics}$$

- First-order algorithm
- Static OPF

Gan & Low, JSAC 2016

This paper

$$\boxed{\begin{array}{ll} \min & L(x, y(x);\ \mu) \\ \text{over} & x \in X \end{array}}$$

- Quasi-Newton method
- Drifting OPF

➔ Better tracking performance

Outline

Problem formulation

Online Newton method for OPF

Drifting OPF

$$\left.\begin{aligned} \min_{x} \quad & c_0(y(x)) + c(x) \\ \text{s. t.} \quad & y(x) \le \overline{y} \\ & x \in X \end{aligned}\right\} \quad \text{static OPF}$$

$$\left.\begin{aligned} \min_{x} \quad & c_0(y(x),\gamma_t) + c(x,\gamma_t) \\ \text{s. t.} \quad & y(x,\gamma_t) \le \overline{y} \\ & x \in X \end{aligned}\right\} \quad \text{drifting OPF}$$

Newton algorithm

$$\boxed{\begin{array}{ll} \min & f_t(x, y(x);\ \mu_t) \\ \text{over} & x \in X_t \end{array}}$$

$$\begin{aligned} x(t+1) &= \left[x(t) - \eta \left(H(t) \right)^{-1} \frac{\partial f_t}{\partial x}(x(t)) \right]_{X_t} && \text{active control} \\ y(t) &= y(x(t)) && \text{law of physics} \end{aligned}$$

- (Quasi) Newton algorithm
- Drifting OPF

Tracking performance

$$\text{error} := \frac{1}{T}\sum_{t=1}^{T}\left\| x^{\text{online}}(t) - x^{*}(t) \right\|$$

control error

Tracking performance

$$\text{error} \quad := \quad \frac{1}{T}\sum_{t=1}^{T}\left\| x^{\text{online}}(t) - x^{*}(t) \right\|$$

Theorem

$$\text{error} \quad \leq \quad \sqrt{\frac{\lambda_M}{\lambda_m}} \cdot \frac{\varepsilon}{1-\varepsilon} \cdot \underbrace{\frac{1}{T}\sum_{t=1}^{T}\left(\left\| x^{*}(t) - x^{*}(t-1) \right\| + \Delta_t \right)}_{\text{avg rate of drifting}}$$

Tracking performance

$$\text{error} \;:=\; \frac{1}{T}\sum_{t=1}^{T}\left\| x^{\text{online}}(t) - x^{*}(t) \right\|$$

Theorem

$$\text{error} \;\leq\; \sqrt{\frac{\lambda_M}{\lambda_m}} \cdot \underbrace{\frac{\varepsilon}{1-\varepsilon}}_{} \cdot \frac{1}{T}\sum_{t=1}^{T}\left(\left\| x^{*}(t) - x^{*}(t-1) \right\| + \Delta_t \right)$$

"initial distance" from $x^{*}(t)$
& error in Hessian approx

Tracking performance

$$\text{error} \;:=\; \frac{1}{T}\sum_{t=1}^{T}\left\| x^{\text{online}}(t) - x^{*}(t) \right\|$$

Theorem

$$\text{error} \;\leq\; \underbrace{\sqrt{\frac{\lambda_M}{\lambda_m}}}_{\text{“condition number” of Hessian}} \cdot \frac{\varepsilon}{1-\varepsilon} \cdot \frac{1}{T}\sum_{t=1}^{T}\left(\left\| x^{*}(t) - x^{*}(t-1) \right\| + \Delta_t \right)$$

Tracking performance

$$R(x,x^*) := \sum_{t=1}^{T} c_0(y(x),\gamma_t) + c(x,\gamma_t)$$

cost of Alg

dynamic regret

$$- \sum_{t=1}^{T} c_0(y(x^*),\gamma_t) + c(x^*,\gamma_t)$$

optimal cost

Tracking performance

$$R(x,x^*) \; := \; \sum_{t=1}^{T} c_0(y(x),\gamma_t) + c(x,\gamma_t)$$

cost of Alg

dynamic regret

$$- \sum_{t=1}^{T} c_0(y(x^*),\gamma_t) + c(x^*,\gamma_t)$$

optimal cost

Theorem

$$R(x,x^*) = \underbrace{O\left(\sqrt{T}\left(1+\sum_{t=1}^{T}\left\| x_{t+1}^* - x_t^* \right\|\right)\right)}_{\text{rate of drifting}} + \underbrace{\sum_{t=1}^{T}\delta_t}_{\text{subopt of local min}}$$

first-order alg

Implementation

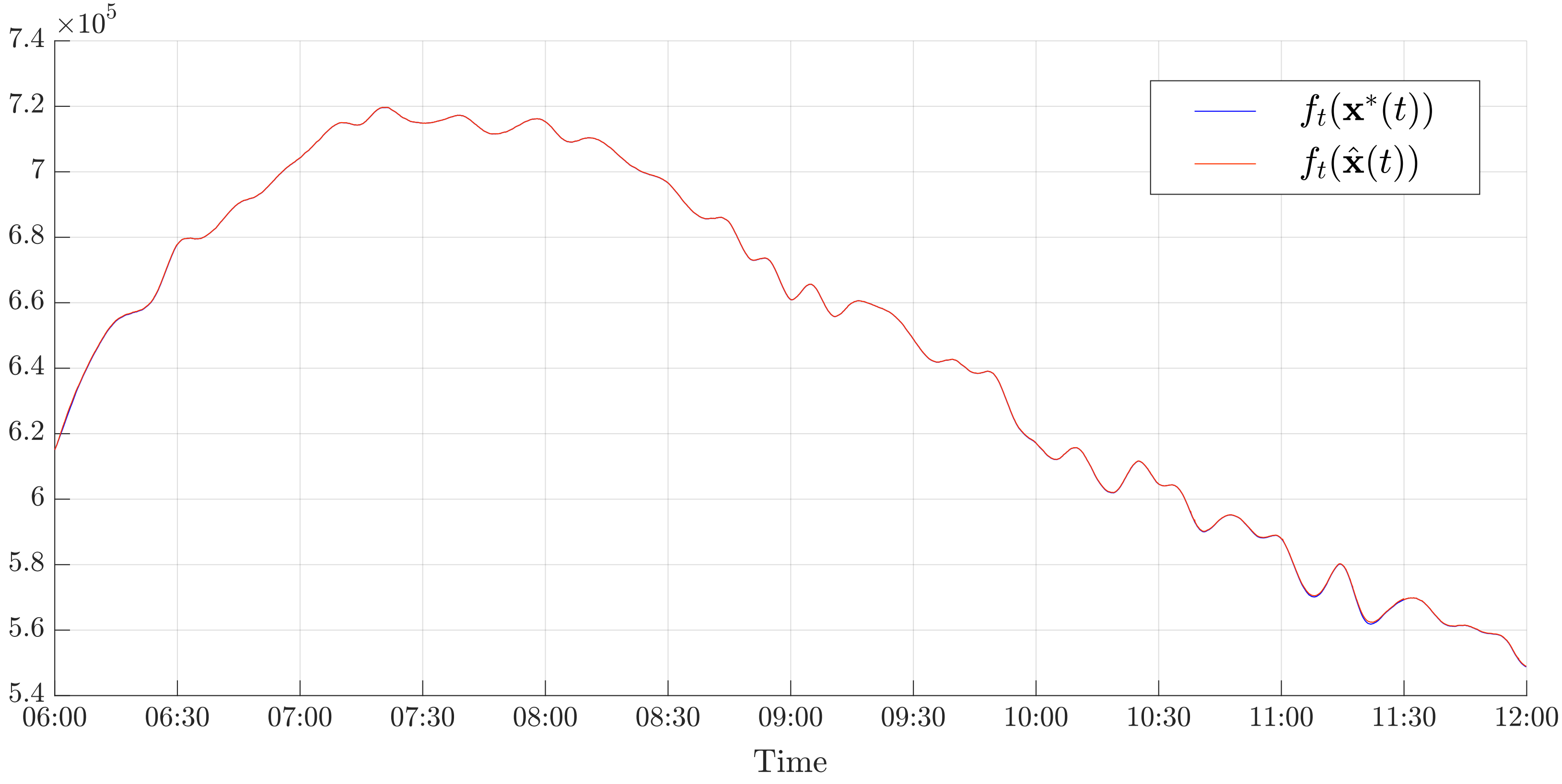
Implement L-BFGS-B

- More scalable
- Handles (box) constraints X

Simulations

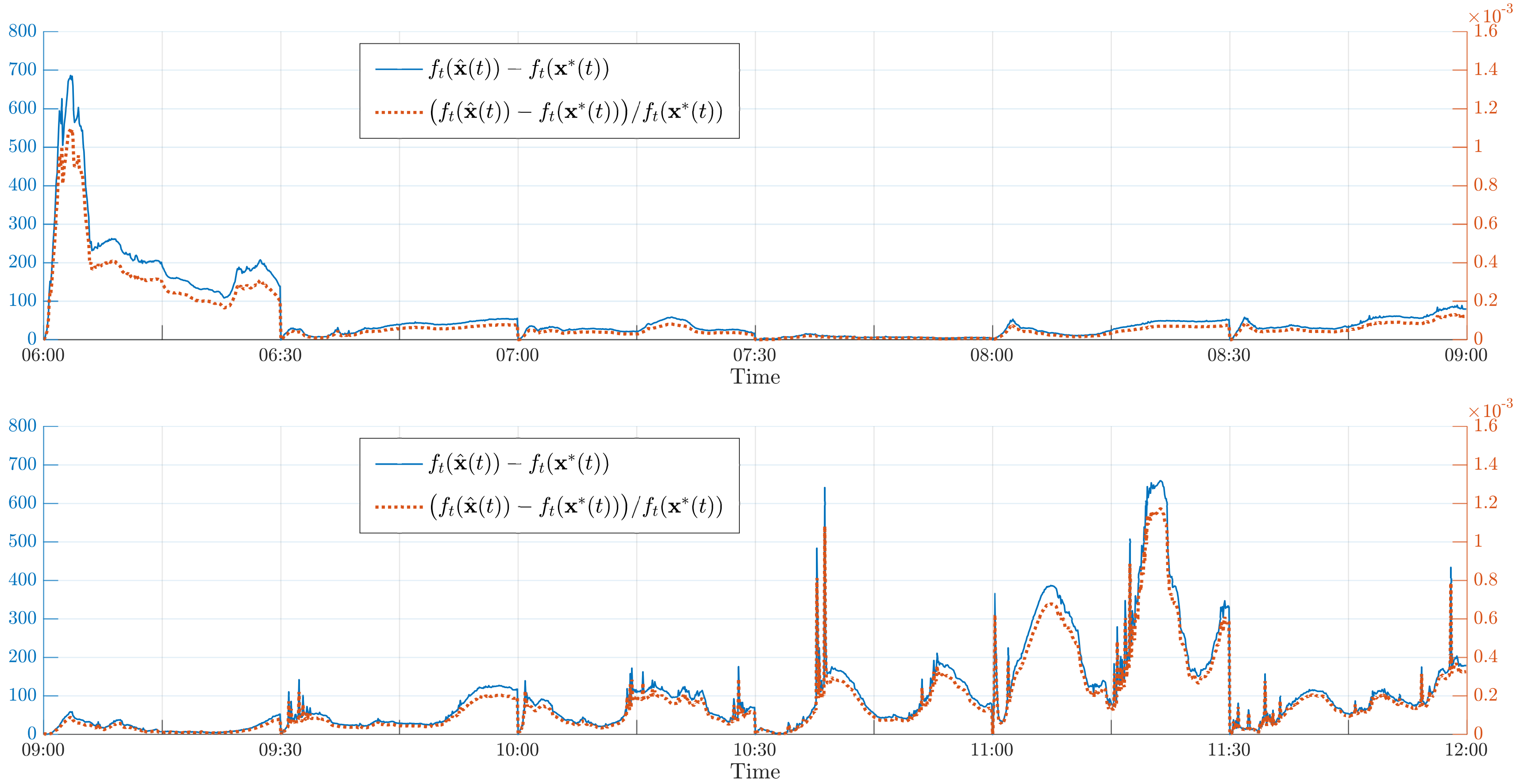
- IEEE 300 bus

Tracking performance



IEEE 300 bus

Tracking performance



IEEE 300 bus

Key message

Large network of DERs

- Real-time optimization at scale
- Computational challenge: power flow solution

Online optimization [feedback control]

- Network computes power flow solutions in real time at scale for free
- Exploit it for our optimization/control
- Naturally adapts to evolving network conditions

Examples

- Slow timescale: OPF
- Fast timescale: frequency control