A COMPOSABLE METHOD FOR REAL-TIME CONTROL OF MICROGRIDS WITH EXPLICIT POWER SETPOINTS. PART I: METHOD

IEEE Power Engineering Society, Swiss Chapter

Microgrids: Evolution And Integration In Modern Power Systems
Lausanne, Switzerland, April, 30th, 2014

Jean-Yves Le Boudec
EPFL, Lausanne, Switzerland

joint work with

Prof. Mario Paolone, Dr. Andrey Bernstein and Lorenzo Reyes, EPFL Laboratory for Communications and Applications and Distributed Electrical Systems Laboratory
Contents

1. Motivation
2. The Commelec Protocol

Reference

Andrey Bernstein, Lorenzo Reyes-Chamorro, Jean-Yves Le Boudec, Mario Paolone
Switzerland 2035: $\geq 30$ to 40% generation will be distributed and volatile

Example of daily measured power injected by solar arrays at EPFL

source: Prof. Mario Paolone, Distributed Electrical Systems Lab, EPFL
≥ 30 to 40% generation will be distributed and volatile

source: Prof. Mario Paolone, Distributed Electrical Systems Lab, EPFL
Remark#1: possibility to have phases along the day with large reduction of the net power flow on the transmission network.

Remark#2: need of faster ramping in the evening hours

Source: Terna S.p.A.
Outlook for 2035

Challenges for grids

- quality of service in distribution networks
- participation of distributed generation to frequency and voltage support (*Virtual Power Plant*)
- autonomous small scale grids with little inertia

Solutions

- fast ramping generation (fossil fuel based)
- local storage, demand response
- *real time control* of local grids
Real Time Control of Grids

- Typically done with droop controllers

- Problems:
  - system does not know the state of resources (e.g. temperature in a building, state of charge in a battery)
  - all problems made global

- Alternative: explicit control of power setpoints
Requirements for an Explicit Control Method

1. Real time
2. Bug free  
   (i.e. simple)
3. Scalable
4. Composable  
   e.g. TN1 can control DN2; DN2 can control SS1
2. COMMELEC’s Architecture

- Software Agents associated with devices
  - load, generators, storage
  - grids

- Grid agent sends explicit power setpoints to devices’ agents
Resources and Agents

- Resources can be
  - controllable (sync generator, microhydro, battery)
  - partially controllable (PVs, boilers, HVAC, freezers)
  - uncontrollable (load)

- Each resource is assigned to a resource agent

- Each grid is assigned to a grid agent

- Leader and follower
  - resource agent is follower or grid agent
  - e.g. LV grid agent is follower of MV agent
Every agent advertises its state (every $\approx 100$ ms) as PQt profile, virtual cost and belief function.

Grid agent computes optimal setpoints and sends setpoint requests to agents.

Communication is over D-TLS and IPRP – details not discussed today.
A Uniform, Simple Model

- Every resource agent exports
  - constraints on active and reactive power setpoints $P, Q$ (PQt profile)
  - virtual cost
  - belief function

I can do $P, Q$

It costs you (virtually) $C(P, Q)$
Examples of PQt profiles

Battery

PV plant

Synchronous Generator

\[ \text{cos}_{\text{min}}(\phi) = 0.8 \]
Virtual cost act as proxy for Internal Constraints

I can do $P, Q$
It costs you (virtually) $C(P, Q)$

If state of charge is 0.7,
I am willing to inject power

If state of charge is 0.3,
I am interested in consuming power
Examples of Virtual Costs

Battery

$C_b(Q) = 0$

Synchronous Generator
Commelec Protocol: Belief Function

- Say grid agent requests setpoint \((P_{set}, Q_{set})\) from a resource; actual setpoint \((P, Q)\) will, in general, differ.
- **Belief function** is exported by resource agent with the semantic: resource implements \((P, Q) \in BF(P_{set}, Q_{set})\)
- Essential for safe operation
PQt profile = setpoints that this resource is willing to receive
Belief function = actual operation points that may result from receiving a setpoint
Grid Agent’s job

- Leader agent (grid agent) computes setpoints for followers based on:
  - state estimation
  - advertisements received
  - requested setpoint from leader agent

- Grid Agent attempts to minimize:

\[
J(x) = \sum_{i} w_i C_i(x_i) + W(z)
\]

  - virtual cost of resource \( i \)
  - keeps voltages close to 1 p.u. and currents within bounds

- Grid Agent does not see the details of resources:
  - a grid is a collection of devices that export PQt profiles, virtual costs and belief functions and has some penalty function
  - problem solved by grid agent is always the same
Given estimated (measured) state $\hat{x} = (\hat{P}_i, \hat{Q}_i)$ computed next setpoint is

$$x = \text{Proj}\{\hat{x} + \Delta x\}$$

where

$\Delta x$ is a vector opposed to gradient of overall objective

Proj{} is the projection on the set of safe electrical states

This is a randomized algorithm to minimize $E(J(x))$
Setpoint Computation by Grid Agent involves gradient of overall objective = sum of virtual costs + penalty

Synchronous Generator

Battery

Voltage deviation penalty

+ line congestion penalty
A system, including its grid, can be abstracted as a single component.

Given PQt profiles of $S_1, S_2, S_3$, solve load flow and compute possible $P_0, Q_0$ + overall cost $C_0(P_0, Q_0)$
Aggregation Example

- Boiler
- Microhydro
- Non-controlled load
- Battery
- PV
Aggregated PQt profile safe approximation (subset of true aggregated PQt profile)
Aggregated Belief

safe approximation
\textit{(superset of true aggregated belief)}
Separation of Concerns

**Resource Agents**
- Device dependent
- Simple:
  - translate internal state (soc) into virtual cost
  - Implement setpoint received from a grid agent

**Grid Agents**
- Complex and real time
- But: all identical
Reliability and Security

- Grid Agent development uses Prof Sifakis’s rigorous system development approach and the BIP framework.

- Grid Agent are triplicated, Resource Agents use voting.

- Communication used authentication (D-TLS) and real time reliability protocols.
Separation of Time Scales

- real time control (grid agent)
- “trip planning” (at local resources)
  resource agent translates
  long term objective into current
  cost function

\[ \text{total energy delivered} \]

\[ \text{upper bound} \]

\[ \text{lower bound} \]

at \( t_1 \) load agent exports a cost function that expresses desire to consume energy

at \( t_2 \) load agent exports a cost function that expresses desire to stop consuming energy

Commelec = grid autopilot
Thank You!