UNIVERSITY OF CALIFORNIA,
IRVINE

# A Distributed Smart Grid Control Model for Integration of Renewables

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Computer Science

by

Kiyoshi Nakayama

Dissertation Committee:
Professor Lubomir F. Bic, Co-chair
Professor Michael B. Dillencourt, Co-chair
Associate Professor Jacob Brouwer

2014

# DEDICATION

To my wife

To my parents

In recognition of their worth.

To my mentor, Daisaku Ikeda,

who wrote the following:

"A great human revolution in just a single individual

will help achieve a change in the destiny of a nation and, further,

will enable a change in the destiny of all humankind."

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# NOMENCLATURE

## ACRONYM

**AdT** Adjacent Tie-set

**BFS** Breadth First Seach

**BPC** Battery Penalty Cost

**CGF** Centralized Generation Facility

**CAT** Communication among Tie-sets

**DAORDER** Distributed Algorithm for Optimal Real-Time Distribution of Renewables

**DAPLM** Distributed Algorithm for Power Loss Minimization

**DATIC** Distributed Algorithm for Tie-set Information Configuration

**DER** Distributed Energy Resource

**DFS** Depth First Search

**DLP** Distributed Linear Programming

**FTM** Find Tie-set Message

**LEAT** Leader Election Algorithm in Tie-set

**LEL** Lower Energy Limit

**MV** Measurement Vector

**OPF** Optimal Power Flow

**ORDER** Optimal Real-time Distribution of renewable Energy Resources

**ORDER-RPF** Optimal Real-Time Distribution of renewables Energy Resources with Real Power Flow

**ORDER-T** Optimal Real-Time Distribution of renewable Energy Resources in Tie-set

**ORDER-RT** Optimal Real-Time Distribution of renewable Energy Resources with Real Power Flow in Tie-set

**ORPF** Optimal Real-Time Power Flow

**ORPF-E** Extended model for Optimal Real-Time Power Flow

**PLM** Power-flow Loss Minimization

**RPF** Real Power Flow

**RPR** Renewable Penetration Rate

**SOP** System Operating Point

**STA** Spanning Tree Algorithm

**TA** Tie-set Agent

**TADiC** Tie-set based Autonomous Distributed Control

**TBO** Tie-set Based Optimization

**TEF** Tie-set Evaluation Function

**TEFM** Tie-set Evaluation Function Message

**TF** Tie-set Flag

**TFS** Tie-set Flag Signal

**TPL** Tie-set Power Loss

# NOTATIONS

### Notations for All Chapters

$D$**:** The diameter of a graph $G$.

$E = \{e_1, ..., e_m\}$**:** The set of edges of $G$.

$e_k$**:** An arbitrary edge of a graph $G$.

$e(v_i, v_j)$**,** $i \to j$**:** An edge from node $v_i$ to node $v_j$.

$G = (V, E)$**:** Bi-connected and undirected graph as networks are assumed redundant and links are bidirectional.

$j : j \to i$**:** Set of predecessors of node $v_i$.

$k : i \to k$**:** Set of successors of node $v_i$.

$l$**:** An edge of $\overline{T}$, where $l = (a, b) \in \overline{T}$.

$L_i$**:** Tie-Set, a set of all the edges $\{e_1^i, e_2^i, ...\}$ in a loop of $G$.

$L(l)$**:** A fundamental tie-set determined by an edge $l = (a, b) \in \overline{T}$ and a path $P_T(a, b)$ on $G_T$.

$\mathbb{L}_B$**:** A fundamental system of tie-sets $\mathbb{L}_B = \{L_1, L_2, ..., L_i, ..., L_\mu\}$.

$T$**:** The spanning tree within $G$.

$\overline{T}$**:** Cotree of $G$, where $\overline{T} = E - T$.

$V = \{v_1, ..., v_n\}$**:** The set of vertices of $G$.

$v_i$**:** An arbitrary node in a network.

$V(L_i)$**:** Set of all the nodes included in tie-set $L_i$.

$\rho$**:** The rank of $G$, where $\rho = \rho(G) = |T|$.

$\mu$**:** The nullity of $G$, where $\mu = \mu(G) = |\overline{T}|$.

**Chapter 2: TADiC**

$\mathcal{A}(v_i)$**:** Adjacent nodes of node $v_i$.

$\mathcal{E}(v_i)$**:** Incident links of node $v_i$.

$G_T$**:** A subgraph $G_T = (V, T)$, i.e. $G_T = (V, T)$, where $T \subseteq E$.

$\underline{G} = (\underline{V}, \underline{E})$**:** Tie-set graph, where a set of vertices $\underline{V}$ corresponds to a fundamental system of tie-sets $\{L_1, L_2, ..., L_\mu\}$, and a set of edges $\underline{E}$ corresponds to a set of edges $\{\underline{e}(L_i, L_j)\}, (i \neq j)$, which represent the connections among tie-sets.

$\underline{G}_e = (\underline{V}, \underline{E}_e)$**:** E-tie-set graph. If $L_i \cap L_j \neq \emptyset$, $L_i$ and $L_j$ have an edge $\underline{e}(L_i, L_j)$.

$\underline{G}_v = (\underline{V}, \underline{E}_v)$**:** V-tie-set graph. If $V(L_i) \cap V(L_j) \neq \emptyset$, $L_i$ and $L_j$ have an edge $\underline{e}(L_i, L_j)$.

$\mathbb{L}_i^a$**:** Adjacent tie-sets of tie-set $L_i$.

$\mathbb{L}(v_i)$**:** Tie-set Information of node $v_i$.

$P_T$**:** One elementary path whose origin is $b$ and terminal is $a$ of $l = (a, b)$ in $G_T$.

$\Re(L_i, L_j)$**:** Relation between two fundamental tie-sets.

$v_a$**:** An adjacent node of an arbitrary node.

$v_l^i$**:** Leader node of tie-set $L_i$.

$v_o$**:** A node that creates *Find Tie-set* messages.

$v_r$**:** A node that receives a *Find Tie-set* message.

$y_i(t)$**:** Measurement vector of tie-set $L_i$ at time $t$.

$\Phi(L_i, t)$**:** Tie-set Evaluation Function.

$\zeta(L_i)$**:** Tie-set Flag.

**Chapter 3: ORDER**

$\overline{B}(t)$**:** The average value of node energies of all the nodes in $G$ defined as $\overline{B}(t) = \frac{\sum_{v_i \in V} b_i(t)}{|V|}$.

$b_i(t)$**:** Node energy that is battery energy level at node $v_i$ at time $t$ where $b_i(t) = b_i(t-1) - p_i(t)\Delta\tau$.

$b^*$**:** Minimum energy level of nodes' storage.

$b_i^d(t)$**:** Net node energy defined as $b_i^d(t) := b_i(t-1) - d_i(t)\Delta\tau$.

$\overline{b}_i$**:** Finite battery capacity.

$\overline{B}_l(\tau)$**:** Average value of node energies in tie-set $L_i$ where $\overline{B}_l(t) = \frac{\sum_{v_i \in V(L_i)} b_i(t)}{|V(L_i)|}$.

$\Delta B_i(t)$**:** Spare battery space defined as $\Delta B_i(t) = \overline{b}_i - b_i(t)$ that is preserved for unexpected renewable generation peak when $r_i(t)$ is larger than the load process $l_i(t)$.

$d_i(t)$**:** Net demand, which is the difference between load and renewable generation.

$f_{ij}(t)$**:** Power flow on the link $e(v_i, v_j) \in E$.

$\overline{f}_{ij}$**:** Link capacity for the power flow.

$f_{P(i,j)}(t)$**:** Path flow on a path $P(i,j)$ from node $v_i$ to $v_j$ at time $t$.

$F_i(t)$**:** Net power flow where $F_i(t) := \sum_{k:i\to k} f_{ik}(t) - \sum_{j:j\to i} f_{ji}(t)$.

$g_i(t)$**:** Amount of real power injection from Centralized Generation Facility (CGF) to node $v_i$ at time $t$.

$\overline{g}_i$**:** CGF Generation capacity at node $v_i$.

$H_i(t)$**:** Battery Penalty Cost of node $v_i$ at time $t$.

$LB_i(t)$**:** Lower Energy Limit (LEL) of node $v_i$ at time $t$.

$l_i(t)$**:** Total load of node $v_i$ at time $t$.

$\overline{l}_i$**:** Expected maximum load.

$\overline{l}_i(t)$**:** $\max_{t\in\mathcal{T}}\{l_i(t)\}$ among the load profile $\{l_i(0), l_i(1), ..., l_i(t)\}$ until time $t$ at node $v_i$.

$p_i(t)$**:** Net power export that is the total amount of discharging power from node $v_i$ at time $t$ where $p_i(t) = d_i(t) - g_i(t) + F_i(t)$.

$r_i(t)$**:** Amount of real power injection of node $v_i$ at time $t$ provided by renewable energy resources.

$\overline{r}_i$**:** Expected maximum renewable generation.

$V_i$**:** Voltage at node $v_i$.

$Y_{ij}$**:** Admittance on link $e(v_i, v_j)$.

$\delta(t)$**:** Average deviation of node powers at time $t$ where $\delta(t) = \frac{\sum_{v_i \in V} |b_i(t) - \overline{B}(t)|}{|V|}$.

$\mathcal{T}$**:** Time sequence given to each problem.

$\Delta\tau$**:** Time interval between time $t$ and $t-1$.

$\theta_i(t)$**:** Phase angle at node $v_i$ at time $t$.

**Chapter 4: ORPF**

$b_i(t)$, $\overline{b}_i$, $\overline{B}(t)$, $d_i(t)$, $f_{ij}(t)$, $\overline{f}_{ij}$, $f_{P(i,j)}(t)$, $F_i(t)$, $g_i(t)$, $\overline{g}_i$, $l_i(t)$, $\overline{l}_i$, $p_i(t)$, $r_i(t)$, $\overline{r}_i$ from chapter 3 are also used in chapter 4.

$c_i(g_i(t), t)$**:** Cost of CGF generation $g_i(t)$ at node $v_i$ at time $t$.

$\mathcal{D}$**:** Set of demand nodes.

$\mathcal{G}$**:** Set of generation nodes.

$h_i(b_i(t))$**:** Cost of using battery with energy level $b_i(t)$ at time $t$.

$\alpha_i$**:** Weight on the cost of battery where $\alpha_i > 0$.

$\gamma_i(t)$**:** Time-varying coefficient for CGF generation.

$\phi_\lambda$**:** Flow optimization function where $\phi_\lambda = \sum_{v_j \in V(L_\lambda)} \alpha_j \left(\overline{b}_j - (b_j(g_j^*(t), t) - F_j(t))\right)$.

$\psi(g_i(t), t)$**:** The sum of costs of CGF generation and of using battery where $h_i(b_i(t)) + c_i(g_i(t), t)$.

**Chapter 5: PLM**

$A$**:** $n \times m$ incidence matrix where $n$ and $m$ are the number of nodes and links.

$b_{k\lambda}$**:** Element of tie-set matrix.

$B$**:** Tie-set matrix.

$c_k$**:** Link capacity.

$C$**:** Vector of link capacities.

$i_k$**:** Link current.

$I$**:** Vector of link currents.

$J$**:** Current injections.

$p_k$, $p(i_k)$: Power loss on link $e_k$.

$p_\lambda(x_\lambda)$: Tie-set Power Loss function of tie-set $L_\lambda$.

$P_G$: Power loss of an entire power network, which is given by the sum of power losses on all the links.

$r_k$: Link resistance.

$R$: Vector of link resistances.

$R^d$: Diagonal matrix diag$(r_1, r_2, ..., r_m)$ whose elements are resistances.

$S$: Power network and its operating conditions where $S = (V, E, I, C, R)$.

$x_\lambda$: Tie-set current (flow) that circulates along a tie-set.

$x$: Vector of tie-set currents.

$\xi_k$: Element of initial current vector.

$\xi$: Initial current vector.

$\Gamma(t)$: Communications complexity at time $t$, which is the number of messages used at time $t$.

# ACKNOWLEDGMENTS

I would like to thank Professor Lubomir F. Bic and Professor Michael B. Dillencourt, my advisors and mentors, for their guidance and support in all the aspects of my Ph.D. life. I would also like to express my appreciation to Professor Jacob Brouwer for providing me a lot of advice not only on the dissertation but also on the entire research work. My gratitude is also to Professor Nalini Venkatasubramanian and Professor Elaheh Bozorgzadeh for their professional and thoughtful advice throughout the research project.

I also owe a great deal to colleagues in Information and Computer Sciences (ICS) and Electrical Engineering and Computer Science (EECS) for their contributions to the project: Kyle E. Benson, Nga Dang, Vahe Avagyan, Yi-Hsi Huang, and Yong hun Eom.

This thesis has greatly benefited from the discussion and useful advice of Professor Steven H. Low and Changhong Zhao in Smart Grid Research Group at California Institute of Technology.

I am grateful to Professor Norihiko Shinomiya and Professor Emeritus Hitoshi Watanabe at Soka University who have kindly collaborated with me to sophisticate the quality of research.

The research in Ph.D. program at UC Irvine has been supported by: Japanese Student Services Organization (JASSO) and Information and Computer Sciences (ICS) Fellowship at University of California, Irvine.

Finally, I would like to thank my wife, Sachiko Nakayama, who has dedicatedly supported my entire Ph.D. life. Without her support, I would not have been able to complete my Ph.D. program. I thank my parents, Seiichi Nakayama and Sayoko Nakayama, for their love and constant encouragement. Despite the geographical distance, I felt strong family bond, which let me overcome the difficulties I have encountered.

# CURRICULUM VITAE

## Kiyoshi Nakayama

### EDUCATION

**Doctor of Philosophy in Computer Science** **2014**
University of California, Irvine

**Master of Science in Engineering** **2011**
Graduate School of Engineering, Soka University, *Tokyo, Japan*

**Bachelor of Science in Engineering** **2009**
Department of Information Systems Science, Soka University, *Tokyo, Japan*

### SELECTED HONORS AND AWARDS

**Best Paper Award** **2012**
IEEE International Conference on Smart Grid Communications (SmartGridComm)

**Best Paper Award** **2010**
IEEE International Congress on Ultra Modern Telecommunications (ICUMT)

**Graduate Research Fellowship** **2011–2014**
Japanese Student Services Organization (JASSO)

**Da-Vinci Award** **2009–2011**
Soka University

## PUBLICATIONS

### Journals (Refereed)

1. **K. Nakayama**, C. Zhao, L. Bic, M. Dillencourt, J. Brouwer, *"Distributed Power Flow Loss Minimization Control for Future Grid,"* Wiley International Journal of Circuit Theory and Applications, 2014. DOI: 10.1002/cta.1999.

2. **K. Nakayama**, N. Shinomiya, H. Watanabe, *"An Autonomous Distributed Control Method Based on Tie-set Graph Theory in Future Grid,"* Wiley International Journal of Circuit Theory and Applications, Vol. 41, No. 11, 2013, pp. 1154 - 1174.

3. **K. Nakayama**, N. Shinomiya, H. Watanabe, *"An Autonomous Distributed Control Method for Link Failure Based on Tie-Set Graph Theory,"* IEEE Transactions on Circuits and Systems-1: Regular Paper, Vol. 59, No. 11, 2012, pp. 2727 - 2737.

### Conference and Symposium Proceedings (Refereed)

1. **K. Nakayama**, N. Dang, L. Bic, M. Dillencourt, E. Bozorgzadeh, N. Venkatasubramanian, *"Distributed Flow Optimization Control for Energy-Harvesting Wireless Sensor Networks,"* IEEE International Conference on Communications (ICC), Sydney Australia, June 2014.

2. **K. Nakayama**, T. Koide, *"A Decentralized Algorithm for Network Flow Optimization in Mesh Networks,"* IEEE Global Communications Conference (Globecom), pp. 1554 - 1559, Atlanta GA USA, Dec. 2013.

3. **K. Nakayama**, C. Zhao, M. Dillencourt, L. Bic, J. Brouwer, *"Distributed Real-Time Power Flow Control with Renewable Integration,"* IEEE Conference on Smart Grid Communications (SmartGridComm), pp. 516 - 521, Vancouver Canada, Oct. 2013.

4. **K. Nakayama**, K. Benson, V. Avagyan, M. Dillencourt, L. Bic, N. Venkatasubramanian, *"Tie-set Based Fault Tolerance for Autonomous Recovery of Double-Link Failures,"* IEEE Symposium on Computers and Communications (ISCC), Split Croatia, July 2013.

5. Y. Sakai, **K. Nakayama**, N. Shinomiya, *"A Node-Weight Equalization Problem with Circuit-Based Computations,"* IEEE International Symposium on Circuits and Systems (ISCAS), pp. 2525-2528, Beijing China, May 2013.

6. **K. Nakayama**, K. Benson, L. Bic, M. Dillencourt, *"Complete Automation of Future Grid for Optimal Real-Time Distribution of Renewables,"* IEEE International Conference on Smart Grid Communications (SmartGridComm), pp. 418 - 423, Tainan Taiwan, Nov. 2012 (**Best Paper Award**).

7. Y. Sakai, **K. Nakayama**, N. Shinomiya, *"A Property Verification of Node-Weight Equalization Focusing on Cycles of a Graph,"* 27th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), Jul. 2012, P-T2-236.

8. **K. Nakayama**, N. Shinomiya, *"Autonomous Recovery for Link Failure Based on Tie-Sets in Information Networks,"* IEEE Symposium on Computers and Communications (ISCC), pp. 671 - 676, Corfu Greece, June 2011.

9. K. Kadena, **K Nakayama**, N. Shinomiya, *"Network Failure Recovery with Tie-Sets,"* IEEE International Conference on Advanced Information Networking and Applications Workshops (WAINA), pp. 467 - 472, Biopolis Singapore, March 2011.

10. **K. Nakayama**, N. Shinomiya, *"Distributed Control Based on Tie-Set Graph Theory for Smart Grid Networks,"* IEEE International Congress on Ultra Modern Telecommunications and Control Systems (ICUMT), pp. 957 - 964, Moscow Russia, October 2010 (**Best Paper Award**).

11. **K. Nakayama**, N. Shinomiya, H. Watanabe, *"Distributed Control for Link Failure Based on Tie-Sets in Information Networks,"* IEEE International Symposium on Circuits and Systems (ISCAS), pp. 3913 - 3916, Paris France, May 2010.

**Conference and Symposium Proceedings (Non-refereed)**

1. **K. Nakayama**, N. Shinomiya, *"Tie-set Graph Theory and its Application to Smart Grid,"* IEICE General Conference, Miyagi Japan, March 2010, BS-3-8, "S-38"-"S-39".

2. K. Kadena, **K Nakayama**, N. Shinomiya, *"A Way of Determining a Fundamental System of Tie-sets Considering a Link Failure Recovery,"* IEICE General Conference, Miyagi Japan, March 2010, BS-3-1, "S-24"-"S-25".

3. **K. Nakayama**, N. Shinomiya, *"A Distributed Control Method based on Tie-sets on a Network,"* Multimedia, Distributed, Cooperative and Mobile Symposium (DICOMO), Oita Japan, July 2009, No. 1, pp. 788 - 796.

4. **K. Nakayama**, N. Shinomiya, *"Autonomous Distributed Control for Optical Sensory Nerve Networks,"* IEICE General Conference, Ehime Japan, March 2009, B-20-32.

## CONFERENCE AND INVITED TALKS

- June 2014: Invited Talk, Communications Research in Signal Processing (CRISP) Group, UC Davis, CA.
- January 2014: Invited Talk, Rigorous Systems Research Group, Caltech, Pasadena, CA.
- December 2013: Invited Talk, Center for Nonlinear Studies, Los Alamos National Laboratory, NM.
- December 2013: Oral Presentation, IEEE Globecom, Atlanta, GA.
- October 2013: Oral Presentation, IEEE SmartGridComm, Vancouver, Canada.
- November 2012: Invited Talk, Global Citizenship Program, Soka University, Tokyo, Japan.
- November 2012: Oral Presentation, IEEE SmartGridComm, Tainan, Taiwan.
- June 2011: Oral Presentation, IEEE ISCC, Corfu Greece.
- October 2010: Oral Presentation, IEEE ICUMT, Moscow, Russia.
- March 2010: Oral Presentation, IEICE General Conference, Miyagi, Japan.
- March 2009: Oral Presentation, IEICE General Conference, Ehime, Japan.

## PROFESSIONAL MEMBERSHIPS

- IEEE Graduate Student Member
- IEEE Communications Society
- IEEE Power and Energy Society
- IEEE Smart Grid Society
- ACM Student Member

# ABSTRACT OF THE DISSERTATION

A Distributed Smart Grid Control Model for Integration of Renewables

By

Kiyoshi Nakayama

Doctor of Philosophy in Computer Science

University of California, Irvine, 2014

Professor Lubomir F. Bic, Co-chair

Professor Michael B. Dillencourt, Co-chair

Future smart grids will likely support bi-directional flow of electricity and include power production from multiple, disparate, and uncontrollable sources due to a high penetration of distributed renewable energy resources. Some of the more challenging problems for the future grid include maximizing the use and efficiency of renewable resources, and realizing optimal demand and power production responses that can complement renewable intermittency. Integration of renewables together with energy storage systems has been motivated by the increasing attention to feature renewable energies not only from solar and wind power but also from the excess generation from many customers. Effective use of renewable resources using battery systems can be realized by balancing distributed energy resources (DERs) with complementary demand and dispatchable generation responses. The spatial distribution, intermittency, and uncontrollability of most renewable resources, however, make stable and reliable electricity transmission and distribution difficult especially with high renewable market penetration in large-scale complex power networks.

In order to use energy storage systems effectively for optimizing DERs and realizing a reliable and sustainable future grid with a lot of real-time end-use devices that anticipate demands automatically, we present an autonomous distributed control model that can realize opti-

mum power flow control together with demand and power response. The proposed model, which integrates tie-set graph theory with an autonomous agent system, effectively divides the power network into a set of independent loops (tie-sets). Autonomous agents constantly navigate the network to dynamically synchronize state information within tie-sets and completely automate the future power network. Due to the theoretical basis of a tie-set graph, the supply and load of electric power at every instant can be balanced even if the future load is uncertain and renewable generation is highly variable and unpredictable.

In this thesis, we developed a distributed control model to deal with the following important problems: Optimal Real-time Distribution of renewable Energy Resources (ORDER), Optimal Real-time Power Flow (ORPF), and Power-flow Loss Minimization (PLM).

The objective of the ORDER problem is minimizing the power supply from a Centralized Generation Facility (CGF), which is a power production station that uses fossil fuels or any other resources except renewables, with balanced distribution and allocation of DERs. To minimize the power supply from CGF, the net demands (loads minus renewables) and power from CGF should be constantly balanced under the condition that the storage system at each node has at least certain amount of energy within the fixed range of the capacity. If all the net demands are balanced, power supply from CGF can be minimized and blackouts can be prevented.

In ORPF problem, we expand the model by considering cost functions of using CGF and battery. The goal is to minimize the total cost, which consists of (1) the cost of power production by the CGF using fossil fuels and (2) the cost of using batteries across multiple time periods to balance the fluctuation of renewable power generation and loads. To solve the ORPF problem, we present a novel decentralized algorithm to autonomously allocate the CGF generations and battery charges/discharges.

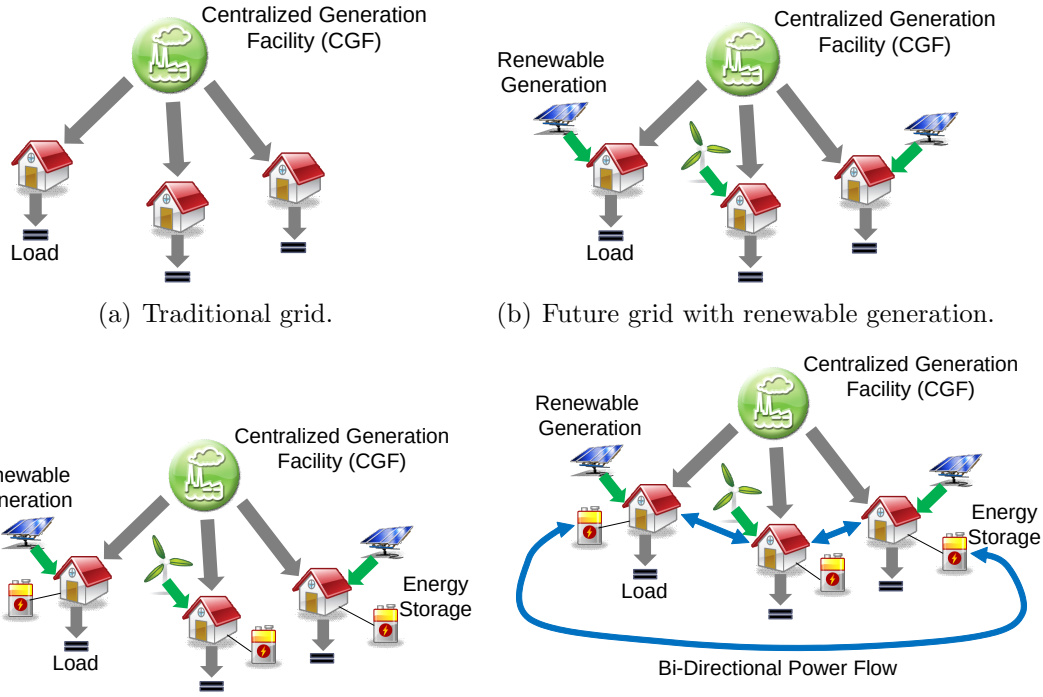
In PLM problem, power flow loss is taken into account on top of the model. A distributed algorithm is proposed to minimize the loss while distributing energy resources to consumers in a future grid that connects many real-time distributed generation systems. We employ the notion of tie-sets to create a linear vector space represented by a set of loops in the power network where the power loss function can be simply formulated and solved. As finding a solution for each tie-set enables global optimization, we propose parallel computing based on tie-sets by integrating autonomous agents; the power loss on every link can be minimized with iterative optimization within a system of independent tie-sets.

# Chapter 1

# Introduction

## 1.1 Background of Smart Grid

In traditional power grids, transmission and distribution processes have been organized by large centralized generation facilities (CGFs) where the power lines catering to bulk electricity transmission have played a significant role 1.1(a). In a decentralized power distribution architecture, however, where small-scale distributed generation units are installed near consumer premises, the transmission level is increasingly circumvented so that distribution and micro grids could take a more important role [1]. Therefore, future smart grids will likely support bi-directional flow of electricity and include power production from multiple, disparate, and uncontrollable sources due to a high penetration of Distributed Energy Resources (DERs) [2, 3, 4] as shown in Figure 1.1. Some of the more challenging problems for the future grid include maximizing the use and efficiency of renewable resources, and realizing optimal demand responses and power flow control. In addition, local autonomy is required due to communication times that are longer than those required to communicate grid perturbations and actuate hardware to prevent widespread outages. Local authority regimes also contain




(a) Traditional grid.

(b) Future grid with renewable generation.

(c) Future grid with renewable generation and energy storage.

(d) Future grid with renewable generation, energy storage, and bi-directional power flow switches.

Figure 1.1: General difference between traditional grid and future grid.

disparate policy and market frameworks in which the optimization and control infrastructure must operate. For this reason, development of an autonomous distributed architecture that can realize optimum allocation of complementary demand and power response together with dispersed renewable energy resources is inevitably required to produce highly efficient and reliable operation of future power networks that will integrate millions of state-of-the-art real-time end-use devices.

### 1.1.1 Intermittency and Uncontrollability of Renewable Energies

The future power grid features a lot of renewable generation systems together with smart meters that anticipate the demands at customer premises as in Figure 1.1(b). Figure 1.1(b) describes the model that the power by renewable generation is either consumed at the cus-

tomer premises or sold to the power companies where CGFs are located. Over the past few years, renewable energy generation by wind and solar power has become one of the fastest growing sectors of the U.S. renewable energy portfolio [5, 6]. In particular, the wind power installation in the U.S. in 2010 was about 5 Gigawatt (GW) and comprised 25% of U.S. electric generating capacity additions [6]. In addition, 12,100 Megawatt (MW) of new wind capacity is expected to be added by the year 2013, with forecasts of meeting 20% of the nation's electricity demand from wind energy by the year 2030 [6]. However, the spatial distribution, intermittency and uncontrollability of most renewable resources make stable and reliable electricity transmission and distribution difficult especially with high renewable market penetration in large-scale complex power networks. Figure 1.2 shows the demand and wind power in 2006 reported in [7]. As in the Figure 1.2, the process of demand and wind power generation shows the random nature during the entire year, although those processes are bounded by 800 MW and 1500 MW, respectively. The wind generation demonstrates its randomness within one day span according to Figure 1.3, even though demand can be anticipated beforehand on the basis of seasonal profiles. Moreover, energy harvesting, which is also known as power harvesting or energy scavenging, is also being integrated especially into micro grids by which energy is derived from external sources (e.g. solar power, thermal energy, salinity gradients, and kinetic energy). Therefore, the intermittent nature of wind generation gives rise to new technical development for integration of renewables into power networks, especially as the market share of wind power becomes large.

### 1.1.2 Development of Energy Storage

Integration of renewables together with energy storage systems has been motivated by the increasing attention to feature renewable energies from not only solar and wind power but also the excess generation from many customers as seen in Figure 1.1(c). According to [8], if we can take full advantage of 20% of energy from renewable harvesting systems, all the demands

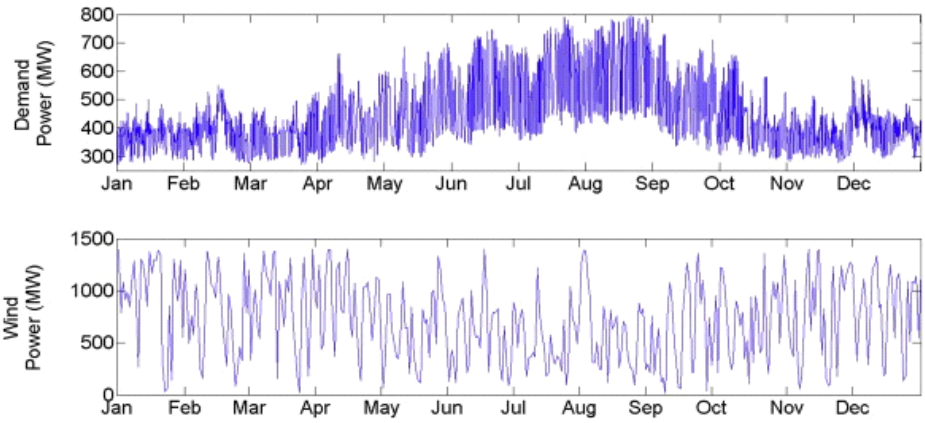



Figure 1.2: Demand and wind power in 2006 [7].

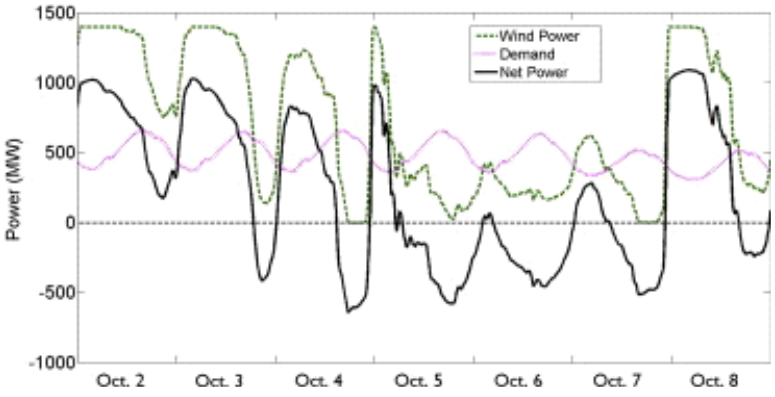



Figure 1.3: Power profiles of wind, demand and net power in October 2006 [7].

in the world will be met sufficiently. Deployment of energy harvesting systems to exploit renewable energies gives rise to the intensive research and development of energy storage technologies [9] as realizing reliable and sustainable power production responses are required to balance the demand and supply. Battery systems save a large amount of generation cost without hurting utility at customers [10]. Although energy storage is gaining its attention with renewable integration, it has not been fully explored [11]. One of the technical and commercial barriers is that availability of renewables such as solar and wind is not easily predictable due to their intermittency, which makes it difficult to evaluate cost efficiency in integrating batteries and deciding their capacities. Nonetheless, the energy storage appears to be the only solution to the issue of intermittency when integrating renewables [12], since the storage can relax the random nature of renewables and enable us to optimize the use of them with certain time range, where temporally stored renewables are soon dispatched to meet demands. Furthermore, energy storage can make a power network stable with a high renewable penetration level. To include renewables as primary sources of energy, we will explore effective use of DERs using energy storage devices that can be realized by real-time balanced transmission and distribution of such distributed resources with complementary demand and dispatchable generation responses.

Energy storage technologies can generally be classified into four main categories [13]: mechanical process, electrochemical process, electromagnetic process, and thermal processes as in Figure 1.4. The energy storage systems in use for electrical energy usually include the first three types. Advanced storage technologies and systems continue to be developed that will increase the reliability, performance, and competitiveness of electricity generation and transmission in a power network.

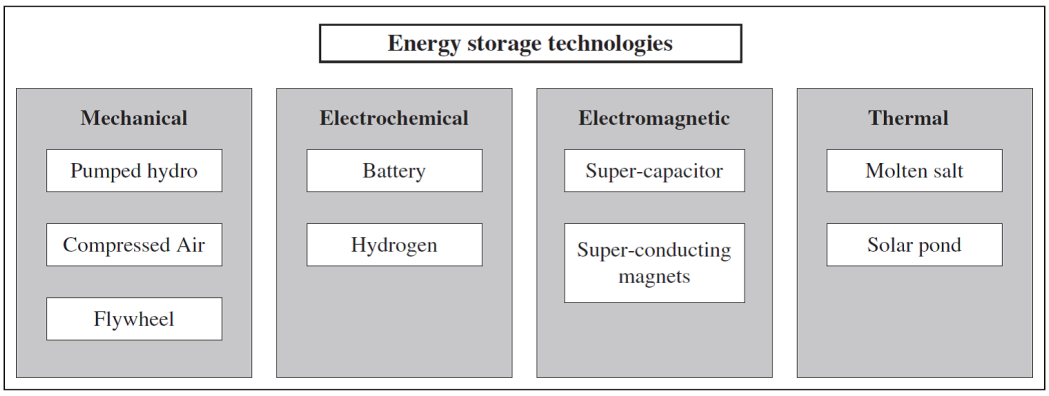



Figure 1.4: Classification of energy storage systems [13].

### 1.1.3 Topology

The topology of today's traditional grids must be modified to integrate and manage DERs when creating future power networks. Traditional grid topology employs a "tree" structure in the distribution system as indicated in Figure 1.1(a). However, the tree topology will not hold anymore when DERs will be integrated into the grid since the topology is the worst in communication delay and power loss as the number of hops between nodes grows in the order of the size of the network [14, 15]. While an interconnected or meshed network would provide higher reliability and flexibility, such meshed power networks are only found in a few urban areas.

Nonetheless, recent attention has been focused on mesh topologies for development of new power network infrastructure because of the increased flexibility, efficiency, and resiliency they provide and the familiarity, good understanding and technology associated with the analogous information and communications network of the Internet [16]. The benefit of using meshed networks also lies in preventing the grid from being isolated with each other in the event of a failure, natural disaster, or required maintenance. For this reason, the future grid would need to be much more connected to support bi-directional power flows as shown in Figure 1.1(d).

Besides those factors, it has also been studied how the minimum power loss in a power system is related to its network topology [17]. In the simulation of chapter 5, we assess how much power loss we can reduce by outputting random resistance and current value on each link and comparing the result using a mesh topology with the one using a tree topology in order to motivate the use and design of mesh topological grids.

## 1.2 Problem Statement

The primary goal of this thesis is providing a completely autonomous distributed control model that integrates

- Millions of real-time end-use devices;
- Individual energy storage devices;
- Distributed energy resources produced locally;
- Bi-directional power flow systems among individual nodes;

in order to maximize the efficiency of renewable energy resources as well as realize a reliable and sustainable future grid even if the future load is uncertain and renewable generation is highly variable and unpredictable.

Given a power network, we define the following terms that are used throughout the thesis.

**Definitions**

- *Node* can be considered as either a bus, system operating point, or customer premise such as house and factory.

- *Link* or *edge* is wired power line between two nodes, or wireless power route on which power packets pass through as network flow.

- *Power flow*, the flow of power that is exchanged between nodes. There are two major types of power flow here, which are independent power flows and dependent power flows.

- *Net power export* is the total amount of power going out from a node, which is total amount of discharging power. When it is positive, it supplies power to the other nodes. Otherwise, it obtains power, which can be considered as net power injection.

- *Load* is consumption of power whose information is obtained by real-time end-use devices such as smart meters. The process of load at node can be predicted by the consumption data from customer premises.

- *Renewable generation* is the power obtained by energy harvesting systems such as solar panels, wind-power generation systems, etc.

- *Net demand* is the delta between load and renewable generation. When the net demand is positive, it becomes consumption, otherwise it is considered to be injection or charging process.

- *Centralized Generation Facility (CGF)* supplies power exploiting traditional energy resources such as fossil fuels, nuclear resources, natural gas, etc.

- *Node energy* is the battery energy level at a node at a time, which is the algebraic sum of the battery energy level from a previous time step, load, renewable generation, CGF power supply, and in-and-out power flows. If the energy level is properly scaled according to each time step with 1-hour interval, we do not need to distinguish between energy and power.

The load and renewable generation are uncontrollable processes, whereas the power flow and CGF power supply are controllable processes. Chapters 3, 4, and 5 deal with the important problems to achieve the primary goal using the definitions above.

### 1.2.1 Real-Time Distribution of Renewable Energies

In order to maximize the efficiency of renewable energies, we need to realize balanced allocation of DERs to cope with renewable intermittency. At the same time, we should minimize the power production using fossil fuels, nuclear, and other resources except renewables as well as constantly balance the net demands and power supply from those traditional resources. (the subtraction of loads and renewables)

In chapter 3, we formulate *Optimal Real-time Distribution of renewable Energy Resources (ORDER)* to conduct power production responses and power-flow distributions both in real-time and distributed manner with the following inputs, constraints, and goal in addition to the definitions above.

**Inputs**

- *Link capacities*, the maximum power flows that can be passed through links, are given with the properties of power lines connected among nodes in a power network.
- *CGF generation capacities*, the maximum amount of power that can be generated at CGFs, depend on the configuration when designing power plants.
- *Battery capacities*, the sizes of batteries installed at nodes, are also dependent on installation of energy storage.

Those capacities are decided when designing a power network. In simulation, those properties

are assigned in an initialization procedure. On the other hand, the following inputs are determined dynamically at each time step.

- *Load process* at each node where the process is supposed to be random, uncertain, and uncontrollable.
- *Renewable generation* at each node where the generation is intermittent, uncontrollable, and not easily predictable.

**Constraints**

We have the following constraints for the ORDER problem.

- The amount of power flowing through a link is bounded by the link capacity.
- The amount of power supply by the CGF to each node is limited by the CGF generation capacity.
- The amount of energy that can be stored in a battery at each node is bounded by the the size of energy storage, that is, the battery capacity.

In addition to the constraints above, there is a particular constraint that plays important role in the ORDER problem.

- *Lower Energy Limit (LEL)* is the minimum energy level in an energy storage that each of the nodes must satisfy to prevent running out of energy in case of sudden load peaks. The amount of energy stored in a battery at each node should be more than or equal to LEL. The LEL can be updated dynamically by an unexpected load peak.

In simulation, the load and renewable generation are uncontrollable processes given at random at each time step, whereas the power flow, CGF power supply, and LEL are controllable processes so that the decentralized model tries to optimize them in a real-time manner.

Therefore, we aim at optimizing those controllable processes to minimize the effect of using traditional resources, and realize reliable and sustainable power networks.

**Goal**

The goal of the ORDER problem is to reduce the utilization of traditional resources as much as possible by assuring reliable use of energy storage systems. Therefore, we define the following objective for ORDER problem.

- Minimize total amount of power generated at CGF of all the nodes at all times, subject to the constraints of ORDER, assuming that each node has a battery with LEL condition.

As the purpose of the ORDER problem is to minimize the total amount of power generated at CGF at all times, we have to distribute DERs stored in the energy storages in a balanced manner. In other words, the nodes that store excess amount of energy from renewable generation ought to distribute their power to the other nodes that are in short of energy, which leads to efficient use of renewables. Therefore, node energies should be allocated in balanced manner so that the battery at each node stores energy with the amount of LEL or greater at all times. By distributing DERs in that way, we can achieve minimizing excessive CGF power supply as ORDER problem deals with constant allocation of those DERs to be stored at batteries of nodes.

The ORDER problem in Chapter 3 describes how we realize real-time distribution of DERs by power flow control in a mesh power network.

### 1.2.2 Dynamic Power Flow with Renewable Integration

On the basis of the model of Chapter 3, we define an *Optimal Real-time Power Flow (ORPF)* Problem to integrate renewable generation by considering the costs of producing CGF power using fossil fuels and of using battery systems.

Optimal power flow (OPF) problems, first formulated and studied by Carpentier, have been studied for more than half a century. Many of the earlier models for OPF focused on static optimizations, i.e., optimizations for isolated periods of time in which power supply and demand must be balanced at every period. Today, it is increasingly common for energy storage devices such as batteries to be installed and used in power grids. By charging and discharging the batteries, mismatches between instantaneous power supply and demand can be balanced. As a result, the requirement of power balance at every instant in the OPF formulation can be relaxed. Multi-period versions of OPF problems can be considered in which the battery charging/discharging profile becomes one of the control variables over which the optimization is defined.

Whereas minimizing the amount of power supply from traditional non environmentally-friendly resources under the constraint of the lower power limit at the battery of a node in the ORDER problem, the ORPF problem focuses on the costs defined as follows:

- *CGF cost* is the quadratic function of CGF power supply with time-varying coefficient. Therefore, the more we produce power at CGF, the larger the CGF cost becomes with quadratic increase.
- *Battery cost* is the penalty cost imposed on the difference between the capacity of the battery and the current energy level (node energy). The more a battery is charged, the less the battery cost becomes.

Here, we clarify the inputs, constraints, and goal for the ORPF problem.

**Inputs**

The inputs are exactly the same as those of ORDER problem, namely: link capacity, CGF generation capacity, battery capacity, load process, and renewable generation.

**Constraints**

Except LEL, the constraints are the same as those of ORDER problem, namely: power flow is bounded by a link capacity, CGF power supply is limited by its capacity, and energy level of a battery is bounded by its capacity.

**Goal**

The goal of the ORPF problem is as follows:

- Minimize the summation of CGF cost and battery cost of all the nodes at all times, subject to the constraints of ORPF assuming that each node has a battery.

The chapter 4 on ORPF formulates the problem and how we minimize the summation of CGF cost and battery cost by deciding power flows among individual nodes and CGF power supply at each node. As the battery cost is taken into account, we should also consider to quantify how efficiently the battery is utilized, which contributes to deciding the size of a battery as well.

### 1.2.3 Power-Flow Loss Minimization Problem

The models studied so far do not take power flow loss into account. Effective use of renewable resources can be realized by grid topology, controls, and optimization that can handle the

dispersed and uncontrollable sources while minimizing the transmission and distribution losses, which offsets the dependence on fossil fuels. As power flow loss minimization problem is a critically important issue as often studied in the area of power transmission loss systems, we formulate a *Power-flow Loss Minimization (PLM)* problem that minimizes the total power distribution loss that can be integrated into the architecture presented above. In chapter 5, we consider the model that minimizes the power flow loss when distributing powers among nodes on top of the model used in Chapters 3 and 4.

Here, we consider power flow loss focusing on current and resistance on a link, which is defined as follows:

- *Power flow loss* is proportionate to the product of the square of current (A: Ampere) and resistance (Ω: Ohm) on a link affected by the distance and thickness.

Then, we clarify the inputs, constraints, and goal for the PLM problem.

**Inputs**

- Link capacities in this case are the maximum currents that can be passed through links given with the properties of power lines in a power network.
- *Resistances* are given with properties of power lines.
- Initial currents are given as current injections to nodes.

The resistances are assigned according to IEEE bus test systems or randomly given when configuring a power network.

**Constraint**

The current on a link is bounded by its capacity.

**Goal**

The goal of PLM problem is as follows:

- Minimize the summation of power-flow losses on all the links, subject to the link capacities, assuming that current and resistance are given for each link.

In the PLM problem, we try to calculate the optimal currents on all the links by redirecting the currents from the links with larger losses to the loss-less links. The PLM can be solved as a convex optimization problem since the power loss on a link forms a quadratic function.

In this thesis, we first propose an autonomous control method on the basis of tie-set graph theory that deals with the nature of loops in a graph as a basic architecture for those problems above, then provide the solution to each problem.

## 1.3 Research Objective

The objective of this thesis is to design a future power network and propose an distributed control model for the future grid to solve the problems stated in Section 1.2. We exploit tie-set graph theory used in realizing Kirchhoff's laws in order to achieve the efficient design and control architecture to utilize energy storage and DERs effectively aiming at a reliable and sustainable future grid with millions of real-time end-use devices. The autonomous distributed control model can realize optimum power flow control together with power responses by integrating tie-set graph theory and the notion of an autonomous agent systems. The

theory of tie-set provides the mathematical basis for effectively dividing the power network into a set of independent loops (tie-sets), and thus creates a linear vector space within a network, which greatly simplifies the structure of mesh networks. Autonomous agents constantly navigate the network to dynamically synchronize state information within tie-sets and completely automate the future power network. Due to the theoretical basis of a tie-set graph, the supply and load of electric power at every instant can be balanced even if the future load is uncertain and renewable generation is highly variable and unpredictable. In addition, our solution towards those problems provides the idea about how we design a power network especially from a topological perspective. In what follows, we will discuss both design and control aspects of future power networks.

# Chapter 2

# TADiC: Tie-set based Autonomous Distributed Control

In order to establish the models, we have particularly integrated tie-set graph theory, which is a principle theory that has played a significant role in realizing Kirchhoff's laws in the field of circuits and power systems. The tie-set graph creates independent logical loop structure in an underlying graph of the network; each loop can be seen as a tie-set representing a set of all edges comprising a loop. The set of loops (tie-sets) forms a linear vector space in the network, and thus the complexity of a target network can be significantly reduced. On the basis of tie-set graph theory, a variety of decentralized architectures have been developed to realize global optimization of an entire network with iterative local autonomous distributed control based on individual tie-sets. As an approach to realize optimal future grid, we provide a completely autonomous distributed control method and decentralized algorithms that integrate tie-set graph theory [18] and an autonomous agent system [19].

## 2.1 Tie-set Graph Theory

The tie-set graph theory is described in [18, 20, 21] in detail, we introduce the basis for the unfamiliar reader.

### 2.1.1 Notion of Tie-sets and its Applications

Since the birth of circuit theory traced back to Kirchhoff's current and voltage laws in 1845 [22], the concept of tie-sets has been applied to core fields of circuits and systems in formulating, solving, and characterizing properties of various problems from reliability and optimization perspectives [23]. [24] discusses a formulation of a basic theory of the information systems, where an interesting example of generalized circuit theory can be the basics in the theory of information networks. In [24], a complicated computer and communication system is simplified using circuit theory by exploiting new quantity information vitality as well as traditional information entropy. The partition problem of a network has also been an important problem as in [25] where the principal partition theorem is presented, which leads to the significant development of circuits and systems by representing them in a simplified form. The partition theorem has been applied to creating tie-set graph theory in which the underlying graph of an information network has been captured. The nature of tie-set graph is verified to be helpful to conduct more theoretic analysis for the network management [18].

#### Relationship between a tie-set graph and a power grid

The result of the recent Pacific Crest Mosaic Smart Grid Survey [16] shows that when asked about the relative importance of communication technologies, respondents selected mesh networks as the technology they will most likely use in their service territory (45%) as shown in Figure 2.1. Point-to-point radio frequency and WiMax were listed as most important by

25% and 10% of respondents, respectively. Tie-set graph theory is discussed on a biconnected graph and applicable to a mesh topological network. Therefore, the distributed control method proposed in this paper has potential to be applied to future smart grid networks. Application of the tie-set concept to distributed control has two major advantages as follows:

1. Simplicity in Synchronization: Now that electricity is able to flow bidirectionally on a power grid, there is a need to communicate the information of demands and supplies in a certain unit. On the basis of loop structures, message passing can be realized within a loop. Therefore, by sending a message around on a circle periodically, each node can recognize the information of other nodes in a loop defined by the tie-set. That leads to the simplicity and consistency in terms of synchronization of state information among nodes in a tie-set.

2. Reduction of Electricity Loss: If the distance between a point of demand and supply becomes larger, the power loss increases. Hence, the distance of power transport should be shortened to minimize the loss of electricity. If electricity is supplied within a loop, the distance of power transmission is greatly shortened in comparison with the case that electricity once converses on a substation and is distributed to points of demand [26].

**Application to Smart Grid**

We utilize the nature that optimization within a system of tie-sets in a power network leads to a global solution [27, 28]. In [29, 30], a novel technique based on tie-sets is proposed to realize global optimization of balanced distribution of dispersed static energies with local optimizations. The overview of autonomous control based on tie-set is introduced in [31, 32] with simple simulation results. The tie-set-based approach has been proved to be more effective than traditional tree-based centralized approaches in making decisions in terms of

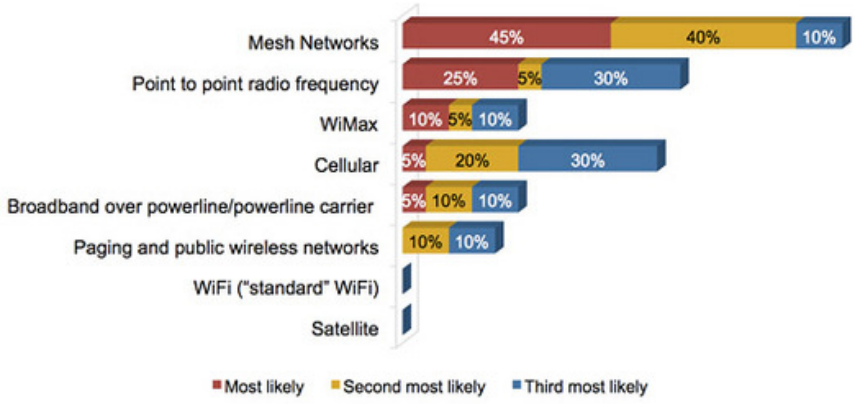



Figure 2.1: The result of surveys by Pacific Crest Mosaic [16].

resiliency [33, 34, 35, 36, 37, 38]. One of the contributions of this thesis is providing a completely autonomous control method integrating tie-set graph theory with an autonomous intelligent agent system [19] to realize optimal real-time distribution of renewables and minimizing power losses in a prospective meshed power network connecting emerging storage systems and millions of end-use devices [39]. The comprehensive simulation results on the basis of TADiC, which are introduced in chapters 3, 4, and 5, demonstrate that the proposed method realizes sustainable real-time distribution of renewables, reduces the excessive CGF power, and minimizes the power flow losses.

### 2.1.2 Fundamental System of Tie-sets

For a given connected graph $G = (V, E)$ with a set of vertices $V$ and a set of edges $E$, let $L_i = \{e_1^i, e_2^i, ...\}$ be a set of all edges that constitutes a loop in $G$. The set of edges $L_i$ is called a *tie-set* [40]. Let $T$ and $\overline{T}$ respectively be a spanning tree and a cotree of $G$, where $\overline{T} = E - T$. $\mu = \mu(G) = |\overline{T}|$ is called the *nullity* of a graph. As $T$ on a graph $G = (V, E)$ is a spanning tree, $T$ does not include any tie-set. In other words, for $l \in \overline{T}$, $T \cup \{l\}$ includes one tie-set. Focusing on a subgraph $G_T = (V, T)$ of $G$ and an edge $l(a, b) \in \overline{T}$, there exists only one elementary path $P_T$ whose origin is $b$ and terminal is $a$ in $G_T$. Then, a *fundamental*

*tie-set* that consists of the path $P_T$ and the edge $l(a,b)$ is uniquely determined as

$$L(l) = \{l(a,b)\} \cup P_T(b,a). \tag{2.1}$$

It is known that $\mu = |\overline{T}|$ fundamental tie-sets exist in $G$, and they are called a *fundamental system of tie-sets* $\mathbb{L}_B = \{L_1, L_2, ...L_\mu\}$.

In network segmentation by tie-sets, a topology of tree $T$ becomes important. Figure 2.2(a) utilizes a tree created by Breadth-First Search (BFS); Figure 2.2(b) uses a tree by Depth-First Search (DFS). Therefore, formation of tie-sets differs by tree structure.

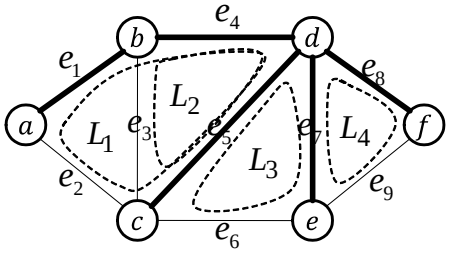
We define a system of *simplest tie-sets* as the system of tie-sets where the number of edges that constitute each loop is minimum compared with other systems of tie-sets. The example of a system of simplest tie-sets can be found in Figure 2.2(c).

If $G$ is bi-connected, a fundamental system of tie-sets $\mathbb{L}_B = \{L_1, L_2, ...L_\mu\}$ guarantees that it covers all the vertices and edges of $G$ as shown in Figure 2.2. Even though a given graph is non-planar, a fundamental system of tie-sets creates a set of $\mu$ independent loops as seen in Figure 2.3.

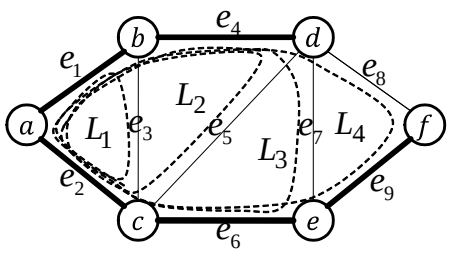
### 2.1.3 Independency of Tie-sets

Any tie-set in a fundamental system of tie-sets is independent of each other. That is, any tie-set cannot be obtained by calculus $\oplus$[1] among other tie-sets. As $l_i \in \overline{T}$ is only included in a fundamental tie-set $L(l_i)$ of a fundamental system of tie-sets $\mathbb{L}_B$, a tie-set that includes $l_i$ cannot be created even if the calculus $\oplus$ is applied to other fundamental tie-sets than $L(l_i)$. For any $\{L(l_i), L(l_j)\} \subseteq \mathbb{L}_B, (l_i \neq l_j)$, if $L(l_i) \cap L(l_j) \neq \emptyset$, a tie-set $L'(l_i)$ replaced by

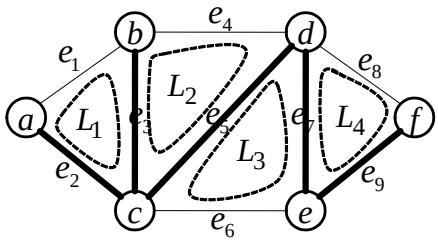
[1]The definition of $\oplus$ for a set $A$ and a set $B$ is defined as follows: $A \oplus B = (A - B) \cup (B - A) = (A \cup B) - (A \cap B)$.




(a) Tie-sets with Breadth-First Search (BFS).




(b) Tie-sets with Depth-First Search (DFS).




(c) Simplest tie-sets.

Figure 2.2: Examples of fundamental systems of tie-sets in a planar graph

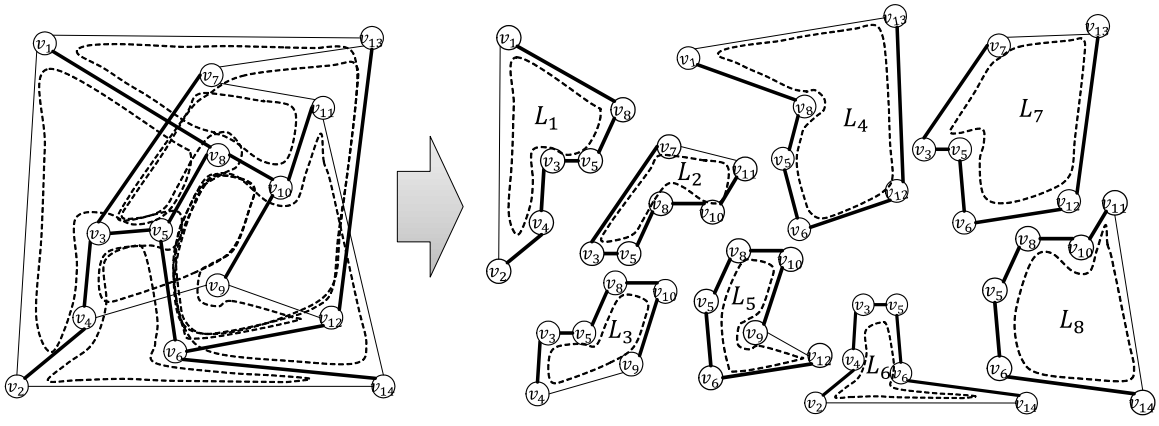



Figure 2.3: A fundamental system of tie-sets in a non-planar graph.

$L'(l_i) \leftarrow L(l_i) \oplus L(l_j)$ is also independent of other fundamental tie-sets. This transformation of a tie-set is done by the calculus $\oplus$ and called *L-Transformation*.

There are systems of independent tie-sets that do not correspond to any fundamental systems of tie-sets constructed by spanning trees of a graph. In Figure 2.4, the tie-sets $\{L_1, L_2, L_3, L_4, L_5, L_6\}$ of the network on the left side forms a fundamental system of tie-sets as it can be created a spanning tree expressed as thick links. Then, in Figure 2.4, L-Transformation is conducted as follows:

$$L'_3 \leftarrow L_2 \oplus L_3,\ L'_6 \leftarrow L_5 \oplus L_6.$$

As $L'_3$ and $L'_6$ are created by L-Transformation, they are also independent tie-sets. However, we cannot find any spanning tree that forms the system of independent tie-sets $\{L_1, L_2, L'_3, L_4, L_5, L'_6\}$ after the L-Transformation in Figure 2.4.

Let $\mathbb{L}'$ be the transformed system of tie-sets from $\mathbb{L}_B$ by replacing $L_i$ by the L-Transformation $L'_i \leftarrow L_i \oplus L_j$. Then, the following holds true between fundamental systems of tie-sets $\{\mathbb{L}_B\}$ and systems of independent tie-sets $\{\mathbb{L}'\}$.

$$\{\mathbb{L}_B\} \subseteq \{\mathbb{L}'\} \tag{2.2}$$

Therefore, by repeatedly conducting L-Transformations, a system of independent tie-sets suitable for a target network management is obtained.

### 2.1.4 Global Optimization within Tie-sets

Tie-set graph theory realizes global optimization on the basis of local optimization within the system of fundamental tie-sets. Let $x_i$ be a variable assigned to a tie-set $L_i$, and $\mathbf{x}$ be a

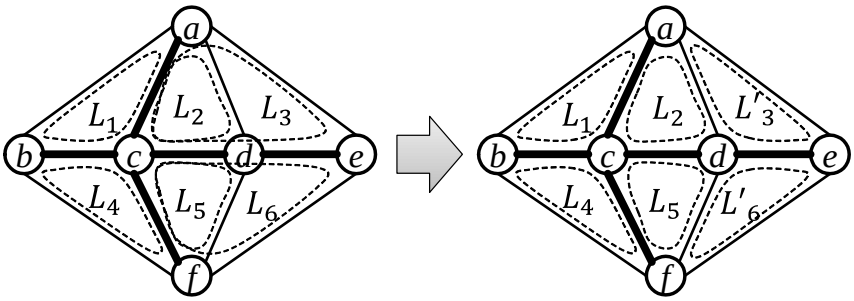



Figure 2.4: An example of a system of independent tie-sets that does not correspond to any fundamental system of tie-sets. Thick and thin lines are links of a tree $T$ and a cotree $\overline{T}$, respectively.

set of variables $\{x_1, x_2, ..., x_i, ..., x_\mu\}$ for all the tie-sets $\{L_1, L_2, ..., L_i, ...L_\mu\}$. Suppose $f_G(\mathbf{x})$ is a function of $\mathbf{x}$ in a graph $G$ that is differentiable in $\mathbf{R}^\mu$, and $f_i(\mathbf{x})$ is a function of $\mathbf{x}$ with regard to $L_i$. At $\mathbf{x} = \mathbf{x}^*$, $f_G(\mathbf{x})$ becomes minimum if the function $f_i(\mathbf{x})$ for each tie-set satisfies the following condition.

$$\left(\frac{\partial f_1(\mathbf{x}^*)}{\partial x_1}, \frac{\partial f_2(\mathbf{x}^*)}{\partial x_2}, ..., \frac{\partial f_\mu(\mathbf{x}^*)}{\partial x_\mu}\right) = 0. \tag{2.3}$$

All the proofs are provided in [20].

### 2.1.5 Tie-set Graph

A graph $\underline{G} = (\underline{V}, \underline{E})$ is defined as a tie-set graph, where a set of vertices $\underline{V}$ corresponds to a fundamental system of tie-sets $\{L_1, L_2, ..., L_\mu\}$, and a set of edges $\underline{E}$ corresponds to a set of edges $\{\underline{e}(L_i, L_j)\}, (i \neq j)$, which represent the connections among tie-sets. There are two ways to determine $\underline{e}(L_i, L_j)$ by focusing on a relation between two fundamental tie-sets $\Re(L_i, Lj)$.

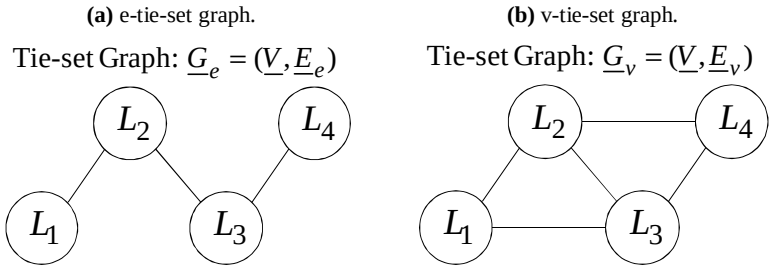



Figure 2.5: The tie-set graphs of Figure 2.2. (a) e-tie-set graph and (b) v-tie-set graph.

**E-tie-set graph** $\underline{G}_e = (\underline{V}, \underline{E}_e)$

If $\Re(L_i, L_j)$ is determined by the set of common edges of $L_i$ and $L_j$, $\underline{G}_e = (\underline{V}, \underline{E}_e)$ is defined as *e-tie-set graph* [41] as shown in Figure 2.5 (a). That is, if $L_i \cap L_j \neq \emptyset$, $L_i$ and $L_j$ have an edge $\underline{e}(L_i, L_j)$.

**V-tie-set graph** $\underline{G}_v = (\underline{V}, \underline{E}_v)$

If $\Re(L_i, L_j)$ is determined by the set of common vertices of $L_i$ and $L_j$, $\underline{G}_v = (\underline{V}, \underline{E}_v)$ is defined as *v-tie-set graph* [41] as shown in Figure 2.5 (b). Let $V(L_i)$ be a set of all the vertices included in a tie-set $L_i$. If $V(L_i) \cap V(L_j) \neq \emptyset$, $L_i$ and $L_j$ have an edge $\underline{e}(L_i, L_j)$.

Each fundamental tie-set of a given graph $G$ is uniquely mapped to the specific tie-set graph $\underline{G}_e$, and $\underline{G}_v$.

## 2.2 Distributed Algorithm for Tie-set Information Configuration (DATIC)

The fundamental system of tie-sets defined in Section 2.1.2 is constructed at each node as state information by using limited information of incident links and adjacent nodes. As de-

scribed in 2.2.1, each node has information of fundamental tie-sets to which the node belongs so as to solve any problems within some loops. In this section, we propose a Distributed Algorithm for Tie-set Information Configuration (*DATIC*). By automatically configuring state information of tie-sets with DATIC, the burden on network managers of network initialization is greatly lessened, especially in large-scale networks.

### 2.2.1 State Information of Node

Each node $v$ mainly has three types of state information as follows:

**Incident Links** $\mathcal{E}(v)$

Link information connected to $v$.

**Adjacent Nodes** $\mathcal{A}(v)$

Node information connected through incident links of $v$.

**Tie-set Information** $\mathbb{L}(v)$

Information of fundamental tie-sets to which $v$ belongs. When a fundamental tie-set $L_i$ contains $e_i$ that includes $v$ in its two vertices, namely $v \in V(L_i)$, it is defined that $v$ belongs to $L_i$ and has information of $L_i$.

An example of state information of a node $c$ in Figure 2.2 is as follows:

- $\mathcal{E}(c) = \{e_2, e_3, e_5, e_6\}$ as incident links.

- $\mathcal{A}(c) = \{a, b, d, e\}$ as adjacent nodes.
- $\mathbb{L}(c) = \{L_1, L_2, L_3\}$ as Tie-set Information.

The node $v_3$ in Figure 2.3 has information of fundamental tie-sets $\{L_1, L_2, L_3, L_6, L_7\}$ as Tie-set Information.

The bound of state information of each node is simply calculated. Each node has at most $|V|$ incident links and $|V|$ adjacent nodes. In addition, a node belongs to at most $\mu$ tie-sets so that the bound of Tie-set Information is $\mu|V|$. Therefore, the bound of state information of a node is $O(\mu|V|)$.

With the Tie-set Information of a node, more efficient data transfer is also possible than message massing using tree paths. When a node $v_i$ transfers a message to a node $v_j$, $v_i$ can select the shortest path to the node to which a message should be sent using its Tie-set Information $\mathbb{L}(v_i)$ if the information of $v_j$ is included in $\mathbb{L}(v_i)$. The routing algorithms on the basis of Tie-set Information in comparison with traditional tree-based data transfer will be discussed in another work.

### 2.2.2 Mechanism of DATIC

Communication paths are represented by a set of links of a spanning tree $T$, which we call "tree links". On the other hand, other links are considered as a set of links of a cotree $\overline{T}$, which we call "cotree links". As in Figure 2.2, thick edges stand for tree links, whereas thin edges stand for cotree links. First, a spanning tree is constructed in a decentralized manner by using one of the distributed tree construction algorithms [42]. In this thesis, the distributed Dijkstra algorithm is exploited.

We define a *Find Tie-set Message (FTM)* to obtain the information of a fundamental tie-set

(Tie-set Information in state information of a node). FTM contains the following information:

- *EdgeTable:* A set of links through which a FTM passed
- *NodeTable:* A set of nodes through which a FTM passed

Here is the overview of the DATIC. After a spanning tree is created, the node $v_i$ with smaller node ID that is connected to each non-tree link $l(v_i, v_j) \in \overline{T}$ sends a FTM to the other node $v_j$ connected to $l(v_i, v_j)$. Then, the node $v_j$ copies FTM and broadcasts them on all the tree links. When $v_i$ receives the FTM that was sent out by $v_j$ and includes information of $l(v_i, v_j)$, $v_i$ stores the Tie-set Information written in the FTM. After storing the Tie-set Information of $L_i$ in $v_i$, the node shares the information of $L_i$ to the nodes that are included in $L_i$.

The procedure of Distributed Algorithm for Tie-set Information Construction, which is called DATIC [34], to obtain *Tie-set Information* $\mathbb{L}(v)$ of node $v_i$ is described as follows:

*Initialization:* Each node $v_i$ checks its incident links $E(v_i)$. For any cotree link $l(v_i, v_j) \in E(v_i)$, $v_i$ negotiates with $v_j$ to decide which node conducts DATIC for Tie-set Information of $L(l)$ based on Node Priority such as Physical Address of a node.

Let $v_o$ be a node that executes DATIC to construct Tie-set Information of $L(l)$ where $l \in \overline{T}$. $v_o$ creates a FTM, and then sends the FTM to the node $v_j$, where $v_j$ is included in a set of two vertices of a cotree link $l(v_o, v_j)$. When sending a FTM to $v_j$, $v_o$ adds node information of $v_o$ to *NodeTable* and link information $e(v_o, v_j)$ to *EdgeTable*.

From here, let $v_r$ be a node that receives a FTM. After receiving a FTM, $v_r$ executes different procedure by the following cases.

*Case 1:* $v_r \neq v_o$ If $v_r$ ($\neq v_j$) is a leaf[2], $v_r$ discards the FTM. If $v_r$ is not a leaf or $v_j$, $v_r$ copies

[2]A leaf $x$ exactly has one tree link in its incident links and other incident links are cotree links.

the FTM and sends the copied FTM to adjacent nodes that are connected through tree links of $E(v_r)$. When sending a copied FTM to an adjacent node $v_a$ that is connected through a tree link, $v_r$ adds node information of $v_r$ to *NodeTable*, and adds link information $e(v_r, v_a)$ to *EdgeTable*.

*Case 2:* $v_r = v_o$ In this case, the FTM has passed through a fundamental tie-set $L(l)$ in a network. The information of *EdgeTable* and *NodeTable* included in the FTM is stored in $v_o$. $v_o$ also notifies the Tie-set Information of $L(l)$ to the nodes that are included in *NodeTable* of the received FTM by just passing the FTM around on a tie-set $L(l)$.

For instance, when constructing Tie-set Information of $L_1$, node $v_1$ starts to create FTM that to be sent across the non-tree link $l(v_1, v_2)$. Then, $v_2$ broadcasts FTMs to all the tree links expressed as thick lines in Figure 2.3. One of the broadcast FTMs is sent back to $v_1$ so that $v_1$ holds the information of $L_1$. Aster that, $v_1$ shares the Tie-set Information of $L_1$ to $v_2$, $v_3$, $v_4$, $v_5$, and $v_8$ that are included in $L_1$. The same procedures are conducted in parallel to construct the other tie-sets in Figure 2.3.

If failure occurs in the middle of DATIC, the node that detects the failure broadcasts failure notification messages, which correspond to Bridge Protocol Data Units (BPDUs) in Spanning Tree Protocol, in order to notify the root node of it. Then, the root creates a spanning tree again followed by DATIC above.

### 2.2.3 Complexity Analysis

Here, we discuss the Communication Complexity and Time Complexity of DATIC from the perspective of a distributed algorithm [42].

### Time Complexity

Analyzing the Time Complexity of DATIC is simple as a FTM is passed through one cotree link and several tree links until a FTM is stored in an original node that sent the FTM or discarded at a leaf node. To analyze the Time Complexity of DATIC, we focus on the diameter of a graph $D$, which is defined as the greatest distance between any pair of nodes in the graph. We use a spanning tree algorithm (STA) to create a path between each pair of nodes. The one with the largest distance is the diameter of a graph. In the first step of DATIC, each node passes its information to its adjacent nodes. Its adjacent nodes add their information on the *NodeTable* and *EdgeTable* and pass to their adjacent nodes and so on and so forth. So the message of Tie-set Information is modified and passed through at most the diameter $D$ of a graph. This step is taken concurrently by every node. Finally, the message comes back to the source node $v_i$. Therefore the maximum number of times a FTM will be forwarded is equal to $|V|$, and therefore the Time Complexity of DATIC is $O(|V|)$ as FTMs are sent simultaneously. As to the size of FTM, a message includes the tie-set information, thus $O(|V|)$.

### Communication Complexity

The Communication Complexity is the total number of messages produced by the entire distributed system to configure Tie-set Information. Only one node $a$ or $b$ of each $l(a,b) \in \overline{T}$ sends a FTM on $l(a,b)$, and there are $\mu$ such links in the graph, where $\mu$ is the nullity of the graph, as described in Section 2.1. Because the FTM is copied and sent along only tree links, it will be copied at most $|V|-1$ times. That is $v_i$, in the worst case, has to pass $|V|-1$ messages to the other nodes. The node, $v_j$ which receives tie-set information configuration message is in charge of adding its information and passing this message back to the source node $v_i$. Most of the messages passed by the nodes will be discarded including that are

passed by two cotrees or form a loop but not go back to the source node. No matter what messages are discarded, the node $v_j$ which receives the message needs to pass at least $|V|-2$ messages to the other nodes except itself and $v_i$. In addition, there are $\mu$ nodes in the system to create FTM messages. Therefore, DATIC completes its procedures with Communication Complexity of $O(\mu|V|)$.

## 2.3 Fundamental Distributed Algorithms based on Tie-set

### 2.3.1 Leader Election Algorithm in Tie-set

*Leader Election Algorithm in Tie-set (LEAT)* [29] selects a leader node to solve a dilemma that occurs when using an overlapping resource among other tie-sets. If two control units come up with two different operating points, the leaders of the two units negotiate with each other to gain priority of control. For example, a local control unit defined by a tie-set shares control of overlapping resources with other units. If two control units come up with two different operating points, the leaders of the two units have to negotiate with each other to gain priority of control.

There are some criteria to choose a leader node in a tie-set. Given the state information defined above, major criteria are as follows:

1. Magnitude relation of node ID (physical addresses)
2. The number of adjacent nodes (incident links)
3. The number of tie-sets to which a node belongs

Use of the criterion 1 enables each tie-set to decide its leader node uniquely, while criteria 2 and 3 do not. When focusing on criteria 2 or 3 to determine a leader node, there is a need to combine with criterion 1. In this study, a node with the largest number of tie-sets as well as with the smallest node ID is selected as a leader node. If a leader node is involved in a lot of tie-sets, communication load can be reduced. As each node has Tie-set Information, it is possible for a node to send a message around on a tie-set and recognize state information of other nodes. A leader node can simply be determined by sending a message around on a tie-set, and comparing the above value individually. In order to decide a leader node in each fundamental tie-set $L_i$, each node sends a *Decide Leader* message to a tie-set. A *Decide Leader* message contains the information as follows:

- $id_m$: The smallest (or largest) node ID among a set of nodes through which a *Decide Leader* message has passed.
- $value_m$: The optimal value of criteria among a set of nodes through which a *Decide Leader* message has passed. The value of criteria is, for instance, the value of above criteria 2 or 3.
- *AddressTable*: Addresses of nodes that belong to $L_i$. A string of addresses must be sorted to satisfy the order as follows: $L_i = (v_0^i, e(v_0^i, v_1^i), v_1^i, ..., v_h^i, e(v_h^i, v_0^i), v_0^i)$ where $V(L_i) = \{v_1^i, \ldots, v_h^i\}$ represents a set of vertices of $L_i$.

**Initialization:**

For each fundamental tie-set $L_i$ of Tie-set Information, each node $v_o$ creates a *Decide Leader* message, and then sends the message to an adjacent node $v_a$ that belongs to $L_i$. When sending a *Decide Leader* message to an adjacent node $v_a$, $v_o$ set its node ID to $id_m$, and assigns its criteria information $value_{v_o}$ to $value_m$. Let $v_r$ be a node that receives a *Decide*

*Leader* message. After receiving a *Decide Leader* message, $v_r$ executes a different procedure by the following cases:

**Case 1:** $v_r \neq v_o$

In this case, $v_r$ compares its criteria information $value_{v_r}$ with $value_m$ of the received *Decide Leader* message.

If $value_{v_r}$ of $v_r$ is better than $value_m$ of the received message, $v_r$ substitutes $value_{v_r}$ for $value_m$. $v_r$ also substitutes its node ID $id_{v_r}$ for $id_m$.

If $value_{v_r} = value_m$, $v_r$ compares its node ID $id_{v_r}$ with $id_m$. Given that smaller node ID is better, if $id_{v_r} < id_m$, $v_r$ substitutes $value_{v_r}$ for $value_m$ as well as substitutes its node ID $id_{v_r}$ for $id_m$.

Otherwise $v_r$ does not change information of the received *Decide Leader* message. Then $v_r$ sends the message to adjacent node $v_a$.

**Case 2:** $v_r = v_o$

In this case, the *Decide Leader* message has passed through on a tie-set $L_i$. Then $v_o$ recognizes a node that holds Node ID $id_m$ as a leader of $L_i$.

In this algorithm, communication complexity is $O(\mu \times |V|)$. As for execution time, a message passes through on a tie-set simultaneously. Therefore, time complexity is $O(D)$ where $D$ is defined as a diameter of $G$.

The leader node $v_l^i$ constantly sends a token message around its tie-set, which we later introduce as a Tie-set Agent, and collects state information of the other nodes within $L_i$. If the leader fails, every node within a tie-set can be a new leader node by just conducting

LEAT again.

### 2.3.2 Communications among Tie-sets

A leader node of a tie-set determines the optimal control for distributed problems and solves them. A leader also decides how to use distributed resources allocated to each node in a tie-set domain. When solving distributed problems, *Communications Among Tie-sets (CAT)* are frequently required. In order to realize communications among tie-sets, each leader of a tie-set must recognize connection information with adjacent tie-sets. In addition to state information described in 2.2.1, a leader node $v_l^i$ of a tie-set $L_i$ has information below:

- Adjacent Tie-sets $\mathbb{L}_i^a = \{L_j\}$

  Based on the concept of $\Re(L_i, L_j)$ stated in the tie-set graph section, an adjacent tie-set $L_j$ of $L_i$ is determined according to the relation of connection $\underline{e}(L_i, L_j) \in \underline{E}$ of $\underline{G}$.

Let $v_l^i$ and $v_l^j$ be a leader node of $L_i$ and $L_j$, respectively. As mentioned, $V(L_j)$ stands for a set of all the nodes included in $L_j$. When the leader node $v_l^i$ of $L_i$ communicates with the leader node $v_l^j$ of an adjacent tie-set $L_j$, $v_l^i$ executes a different procedure according to the following cases.

**Case 1:** $v_l^i = v_l^j$

In this case, there is no need for $v_l^i$ to send any message since $v_l^i$ itself is the leader of the adjacent tie-set $L_j$. Instead, $v_l^i (= v_l^j)$ decides an proper procedure considering state information of both $L_i$ and $L_j$.

**Case 2:** $v_l^i \in V(L_j), (v_l^i \neq v_l^j)$

In this case, $v_l^i$ has state information of $L_j$. Thereby, $v_l^i$ only sends a *Tie-set Communication* message to $v_l^j$ using the topology information of $L_j$.

**Case 3:** $v_l^i \notin V(L_j)$

In this case, $v_l^i$ first detects a node $v_h$ that satisfies a condition $v_h \in V(L_j)$, and then sends a *Tie-set Communication* message to $v_h$. Next, $v_h$ sends the *Tie-set Communication* message to $v_l^j$ using topology information of $L_j$.

For example, adjacent tie-sets of $L_1$ in Figure 2.2 are $\mathbb{L}_1^a = \{L_2, L_3\}$ so that the leader node of $L_1$ constantly communicates with leaders of $\mathbb{L}_1^a$ using its precomputed routing table as explained in [29].

A routing table for communication among tie-sets should be computed according to the above rules before conducting certain procedure so that each leader node can quickly communicate with other leaders using an appropriate path to them.

## 2.4 Tie-set Agent, Tie-set Evaluation Function, and Tie-set Flag

### 2.4.1 Tie-set Agent

*Tie-set Agent (TA)* is an autonomous agent that constantly navigates a tie-set $L_i$ around and brings state information of nodes in $L_i$ to its leader node $v_l^i$. TA contains a measurement vector $y_i(t)$ that provides the value of power level and power flow at each node and link in a

tie-set. The measurements may also include current load level, renewable power generation, weather forecasts, pricing info, among others. The leader node of a tie-set $L_i$, which is considered to be a System Operating Point (SOP) in a tie-set, receives a measurement vector $y_i(t)$ and decides the procedure in $L_i$ based upon $y_i(t)$. TA constantly brings the measurement data $y_i(t)$ of $L_i$ to its leader node with certain time interval.

### 2.4.2 Tie-set Evaluation Function

*Tie-set Evaluation Function (TEF)* is the function that evaluates a tie-set with certain predefined criteria. Obviously, many tie-sets share some overlapping resources with adjacent tie-sets. To avoid contradictions of state information within or among tie-sets, each tie-set communicates with adjacent tie-sets to decide which tie-set processes their overlapping resources first. When deciding the process priority for overlapping resources shared by several adjacent tie-sets, each tie-set exchanges the value of TEF denoted as $\Phi(L_i, t)$ with its adjacent tie-sets $\mathbb{L}_i^a$. TEF $\Phi(L_i, t)$ is calculated based upon the current measurement vector $y_i(t)$. TEF is subject to objective functions, constraints, characteristics of a power network, etc. TEF with random value is only used when the process priority cannot be determined by specific TEF.

The TEF is defined in each problem as described in chapters 3, 4, and 5.

### 2.4.3 Tie-set Flag

*Tie-set Flag (TF)* is used to distinguish a tie-set in process from a stand-by tie-set once the process priority is decided using TEFs. The definition of TF denoted as $\zeta(L_i)$ is as follows:

- $\zeta(L_i) = 0$: $L_i$ is Stand-By

- $\zeta(L_i) = 1$: $L_i$ is In Process

When $\zeta(L_i) = 1$, each adjacent tie-set $L_j \in \mathbb{L}_i^a$ should stand by setting its TF as $\zeta(L_i) = 0$ until $L_i$ finishes its procedure. After $L_i$ finishes its procedure, $L_i$ sends a signal to each adjacent tie-set in order to notify the completion of its procedure as well as making its TF as $\zeta(L_i) = 0$. Then, each tie-set $L_i$ calculates its value of TEF $\Phi(L_i, t)$ based on $y_i(t)$ that is periodically sent with its Tie-set Agent (TA), and compares its value of $\Phi(L_i, t)$ with those of adjacent tie-sets. Parallel optimization among tie-sets is feasible with the TF concept.

## 2.5 Model of TADiC

We describe the procedure of *Tie-set based Autonomous Distributed Control (TADiC)* for complete automation of future power networks. A system operating point (SOP) has the task of integrating growing amounts of renewable power into the power network. The SOP makes decisions to balance the supply and load of electric power at every instant. As forecasts of these variables may have errors, a real-time approach to improve their accuracy was proposed on the basis of accumulating observations when supply and load are balanced [43]. Considering a leader node $v_l^i$ of a tie-set $L_i$ as a SOP, the approach can be integrated with TADiC to accurately estimate load, renewable generation, and power flows.

### 2.5.1 Overview of TADiC

We describe the overview of *Tie-set based Autonomous Distributed Control* (*TADiC*) conducted in each $L_i$, which is described in Algorithm 1 and Figure 2.6, to accomplish complete automation of a future smart grid. The TADiC is conducted if $L_i$ gains process priority to control resources within the tie-set. In order to explain TADiC, we use the notations and

Table 2.1: Notations and Definitions for TADiC

| | |
|---|---|
| AdT $\mathbb{L}_i^a$ | Adjacent Tie-sets $\mathbb{L}_i^a = \{L_j\}$. If $V(L_i) \cap V(L_j) \neq \emptyset$, $L_j$ is an adjacent tie-set of $L_i$. |
| MV $y_i(t)$ | Measurement Vector. MV $y_i(t)$ contains various information about a node $v_i \in V(L_i)$ at time $t$ such as loads and renewables. |
| TA | Tie-set Agent. An autonomous agent that constantly navigates a tie-set to bring the current MV $y_i(t)$ with state info of $L_i$ to its leader node. |
| TEF $\Phi(L_i, t)$ | Tie-set Evaluation Function. A function that evaluates a tie-set based upon the current MV $y_i(t)$ with certain predefined criteria. |
| TEFM | Tie-set Evaluation Function Message. A message used to exchange the value of TEF $\Phi(L_i, t)$ with adjacent tie-sets $\mathbb{L}_i^a$. |
| TF $\zeta(L_i)$ | Tie-set Flag. When $\zeta(L_i) = 0$, a tie-set $L_i$ is stand-by; otherwise $L_i$ is in process ($\zeta(L_i) = 1$). |
| TFS | Tie-set Flag Signal. A signal to notify the state of TF $\zeta(L_i)$. |

definitions in Table 2.1.

Let a *Tie-set Evaluation Function Message (TEFM)* be a message that is used when exchanging the value of TEF $\Phi(L_i, t)$ with adjacent tie-sets $\mathbb{L}_i^a = \{L_1^i, L_2^i, ...\}$. TADiC exchanges TEFM with adjacent tie-sets asynchronously without any global clock.

Each tie-set $L_i \in \mathbb{L}_B$ has a leader node $v_l^i$. TADiC is conducted in the leader $v_l^i$ by communicating with leaders of Adjacent Tie-sets (AdT) $\mathbb{L}_i^a$ using a predefined routing table. Tie-set Agent (TA) constantly navigates each tie-set $L_i$ to bring Measurement Vector (MV) $y_i(t)$ to the leader node $v_l^i$ at time $t$. Based on the MV $y_i(t)$, $v_l^i$ calculates the value of TEF $\Phi(L_i, t)$ and exchanges it with AdT using Tie-set Evaluation Function (TEFM) to decide the process priority among tie-sets. If $L_i$ gains the process priority, $v_l^i$ set its Tie-set Flag (TF) as $\zeta(L_i) = 1$, otherwise $\zeta(L_i) = 0$, and then conducts optimization for each problem in the tie-set. Tie-set Flag Signal (TFS) is used to confirm that the TF of $L_i$ and its AdT is set as 0 so that TADiC is iterated.

### 2.5.2 Procedure of TADiC

The procedure of TADiC is sequentially conducted from *Initialize* to *Send*, *Receive*, *Optimize*, *Notify*, *Confirm*, and *StandBy* in a leader node $v_l^i$ of $L_i$ described as follows:

1. **Initialize**

   In **Initialize**, TEF and TF of $L_i$ are set as $\Phi(L_i, t) = 0$ and $\zeta(L_i) = 0$, respectively. Then, $L_i$ calls **Send**.

2. **Send**

   In **Send**, the value of TEF $\Phi(L_i, t)$ is calculated based upon the current measurement vector $y_i(t)$ at time $t$ provided by TA that constantly navigates $L_i$. After calculating $\Phi(L_i, t)$, $L_i$ writes its TEF value in TEFM. Then, $L_i$ sends the TEFMs to all the adjacent tie-sets $\mathbb{L}_i^a$.

3. **Receive**

   **Receive** is called when $L_i$ receives a TEFM. Until $L_i$ receives TEFMs from all the adjacent tie-sets $\mathbb{L}_i^a$, $L_i$ stands by to receive another TEFM. After receiving all the TEFMs form $\mathbb{L}_i^a$, $L_i$ compares its value of $\Phi(L_i, t)$ with those of adjacent tie-sets. If the value of $\Phi(L_i, t)$ is the largest among those of all the adjacent tie-sets, $L_i$ sets its TF as $\zeta(L_i) = 1$, otherwise $\zeta(L_i) = 0$. If the value of $\Phi(L_i, t)$ is the same as $\Phi(L_j, t)$, $L_i$ uses another TEF $\Phi(L_i, t)$ to decide the procedure priority such as Random TEF. Then, $L_i$ calls **Optimize**.

**Algorithm 1** Tie-set based Autonomous Distributed Control (TADiC)

**INITIALIZE:**
  Set TF as $\zeta(L_i) = 0$. Call SEND.
**SEND:**
  Calculate TEF $\Phi(L_i, t)$ based upon current MV $y_i(t)$.
  Write the value of $\Phi(L_i, t)$ into TEFM.
  Send TEFM to $\mathbb{L}_i^a$.
**RECEIVE:** (Called when received TEFM after SEND)
  **if** $L_i$ received TEFM from all of $\mathbb{L}_i^a$ **then**
    Call COMPARE.
  **else**
    Wait for receiving another TEFM.
  **end if**
**COMPARE:**
  Set TF as $\zeta(L_i) = 1$.
  **for** each $L_j \in \mathbb{L}_i^a$ **do**
    **if** $\Phi(L_i, t) < \Phi(L_j, t)$ **then**
      Set TF as $\zeta(L_i) = 0$.
    **else if** $\Phi(L_i, t) = \Phi(L_j, t)$ **then**
      Use TEF with random value to set TF.
    **end if**
  **end for**
  Call OPTIMIZE.
**OPTIMIZE:**
  **if** $\zeta(L_i) = 1$ **then**
    Conduct optimization for each problem.
    Set TF as $\zeta(L_i) = 0$.
  **end if**
  Call NOTIFY.
**NOTIFY:**
  Send TFS to $\mathbb{L}_i^a$ to notify that TF $\zeta(L_i) = 0$.
**CONFIRM:** (Called when received TFS after NOTIFY)
  **if** $L_i$ received TFS from all of $\mathbb{L}_i^a$ **then**
    Confirm that each TF of $L_j \in \mathbb{L}_i^a$ is $\zeta(L_j) = 0$.
    Call STAND-BY.
  **else**
    Wait for receiving another TFS.
  **end if**
**STAND-BY:**
  Stand by for $\Delta t$.
  Call SEND.

4. **Optimize**

   If $\zeta(L_i) = 1$, $L_i$ conducts optimization within a tie-set $L_i$. Optimization procedure for each problem is defined in chapters 3, 4, and 5. After the optimization, $L_i$ sets its TF as $\zeta(L_i) = 0$. Then, $L_i$ calls **Notify**.

5. **Notify**

   In **Notify**, $L_i$ sends a TFS to each adjacent tie-set $L_j \in \mathbb{L}_i^a$ to notify that $\zeta(L_i) = 0$.

6. **Confirm**

   **Confirm** is called when $L_i$ receives TFS from an adjacent tie-set $L_j \in \mathbb{L}_i^a$ after **Optimize**. In case that **Optimize** is not finished, TFS is temporarily stored in $L_i$. Until $L_i$ receives all the TFSs from all the adjacent tie-sets $\mathbb{L}_i^a$, $L_i$ waits for another TFS. After receiving all the TFSs form $\mathbb{L}_i^a$, $L_i$ confirms that each TF of $L_j \in \mathbb{L}_i^a$ is $\zeta(L_j) = 0$.

7. **Stand-By**

   Let $\Delta t$ be a communication interval among adjacent tie-sets. In **Stand-By**, $L_i$ stands by for $\Delta t$, and then calls **Send** again so that TADiC is iterated.

The flowchart of TADiC is described in Figure 2.6.

### 2.5.3 Comparison in Speed of Convergence

On the basis of TADiC, we achieve parallel optimization within several tie-sets at the same optimization span. Here, we use the data from chapter 5 about Power-flow Loss Minimization (PLM) problem to see how fast it is to conduct parallel optimization compared with non-parallel optimization. In non-parallel optimization, the distributed systems pick only one

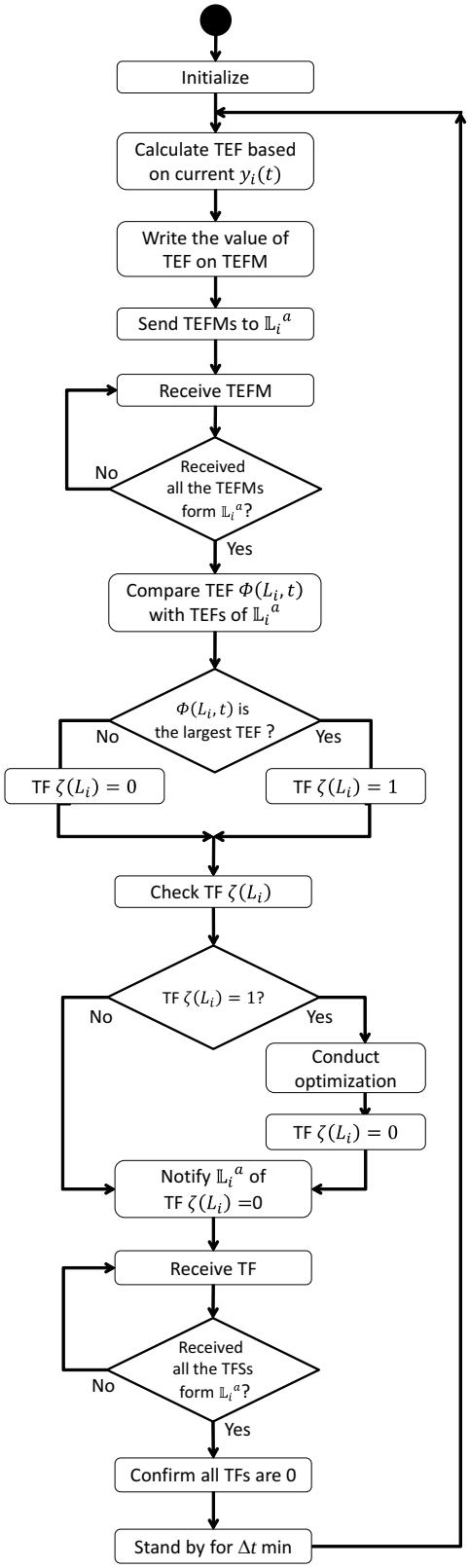



Figure 2.6: Flowchart of Tie-set based Autonomous Distributed Control (TADiC).

Table 2.2: Comparison of the speed of convergence by non-parallel and parallel computations.

| | IEEE 118-Bus System | Random 100-Node Grid |
|---|---|---|
| Non-Parallel | 500 ms | 100 ms |
| Parallel | 174 ms | 15 ms |
| Improvement Rate | 2.87 | 6.67 |

tie-set to be optimized at a time. The time interval between optimization stages, that is, $\Delta t$ of **Stand-By** is 1 ms (millisecond). Table 2.2 shows the speed of convergence by non-parallel and parallel computations in PLM problem. In the IEEE 118 bus system, 48% of power loss reduction is achieved when non-parallel computation and parallel computation spend 500 ms and 174 ms, respectively. In the 100-node grid with random 192-link connections, 51% of power loss reduction is achieved when non-parallel and parallel computations spend 100 ms and 15 ms, respectively. As seen in Table 2.2, the parallel computation model by TADiC demonstrates 2.87 and 6.67 times faster in convergence speed than the non-parallel model using IEEE 188-bus system and random 100-node grid, respectively. This result substantiates the effectiveness of parallel optimization realized by the TADiC model.

### 2.5.4 Error Recovery Method

In **Send** and **Receive** after **Initialize**, a leader node for a tie-set $L_i$ sends and receives TEFMs to decide whether it should be the one to optimize itself based on the magnitude of the TEF with respect to $L_i$ and the TEFs for all adjacent tie-sets $\mathbb{L}_i^a$. To determine whether the magnitude of the TEF for $L_i$ is the maximum among all such tie-sets, the leader node for $L_i$ must receive the TEFM from the leader nodes of all adjacent tie-sets. According to the TADiC, failure to receive the TEFM from one or more leader nodes of adjacent tie-sets means that the leader node for $L_i$ simply stops, waits, and never again recomputes the current flows for $L_i$. In order to make the TADiC sufficiently robust to recover from communications failures, we have a mechanism to cope with this type of failure. A leader

node for $L_i$ polls a leader for an adjacent tie-set, if it has not received the TEFM from the adjacent leader node within a given amount of time, and if no response is elicited from the poll, then it decides to proceed without the TEF information about that adjacent tie-set for this round.

In **Compare**, a leader node for tie-set $L_i$ determines that the magnitude of its TEF is the maximum among adjacent tie-sets in order to **Optimize** power flows and resources within the tie-set. However, one or more adjacent tie-sets also might share the same the TEF values. According to the TADiC, the leader node for $L_i$ uses random choice to decide whether it should optimize $L_i$. The leader nodes for the adjacent tie-sets sharing the same TEF values for their tie-sets also need to determine whether they should optimize their tie-sets. If the leader nodes of the tie-sets with identical TEF values do not make a consistent decision on whether to compute their respective resources, then it is possible that either leader nodes for some or all of these tie-sets recompute their resources at the same time or that none of the leader nodes for these tie-sets recomputes its resources. To specify a resolution technique that results in a consistent decision among tie-sets concerning computation of resources in this case, we incorporate the algorithm to assign unique identifiers to leader nodes and to use them in resolving tie-sets in TEF values among adjacent tie-sets. This results in a consistent resolution of such tie-sets among leader nodes.

In **Notify** and **Confirm**, in order to proceed to the **Stand-By**, the leader node for a tie-set $L_i$ must receive from the leader nodes of all adjacent tie-sets the TFS (flag value) indicating that they are ready to continue. If the leader node for $L_i$ fails to receive flags from the leader nodes of one or more adjacent tie-sets, it exploits the polling method that is the same mechanism used in **Send** and **Receive**.

## 2.6 Summary

In this chapter, we have introduced tie-set graph theory that plays an important role in establishing the distributed control model for smart grid. The tie-set graph creates independent logical loop structure in an underlying graph of the network; each loop can be seen as a tie-set representing a set of all edges comprising a loop. The set of loops (tie-sets) forms a $\mu$-dimensional linear vector space in the network, where $\mu$ is the nullity of a graph, and thus the complexity of a target network can be significantly reduced.

On the basis of tie-set graph theory, a variety of decentralized algorithms have been developed to realize global optimization of an entire network with iterative local autonomous distributed control based on individual tie-sets. The information of tie-sets is constructed within each node as state information using DATIC explained in Section 2.2. Utilizing the Tie-set Information constructed by DATIC, each tie-set selects a leader node that is in charge of optimizing the resources on the tie-set by communicating with the leaders of adjacent tie-sets.

Then, we defined the tie-set agent that navigates a tie-set constantly to (1)bring state information of the node in the tie-set to its leader node and (2)detect error in within the tie-set. The information collected by the tie-set agent is processed to calculate tie-set evaluation function that is used to decide which tie-set occupies its resources and conducts optimization. If the tie-set is in process, we set the tie-set flag as 1; 0 otherwise, which is similar approach to the Semaphore model. By focusing on those concepts, we have proposed the model of Tie-set based Autonomous Distributed Control, which we call TADiC, in order to achieve complete automation of a future smart grid. Each time step, TADiC enables tie-sets to communicate with each other, compare Tie-set Evaluation Functions, and process resources in proper tie-sets.

# Chapter 3

# ORDER: Optimal Real-Time Distribution of Renewables

An *Optimal Real-Time Distribution of Renewable Energy Resources (ORDER)* is a model by which balanced allocation of Distributed Energy Resources (DERs) is dynamically realized in order to achieve a reliable and sustainable grid as well as reduce the power production from centralized generation facilities that exploits fossil fuels, nuclear, etc. Renewable energies are constantly distributed from the premises with excessive energy to the premises that are in short of energy.

The objective of the ORDER problem is minimizing the power supply from a Centralized Generation Facility (CGF), which is a power production station that uses fossil fuels or any other resources except renewables, with balanced distribution and allocation of DERs. To minimize the power supply from CGF, the net demands (loads minus renewables) and power from CGF should be constantly balanced under the condition that the storage at each node has at least certain amount of energy within the fixed range of the capacity. If all the net demands are not balanced, excessive power supply from CGF would be needed to prevent

blackout.

## 3.1 Problem Formulation for ORDER

Here, we formulate the ORDER problem by clarifying definitions and an objective function.

### 3.1.1 Definitions

We consider a connected graph $G = (V, E)$ with a set of nodes $V = (v_i, i = 1, ..., n)$ representing the nodes/buses and a set of links $E \subseteq V \times V$ representing the power lines. The links are directed with arbitrarily defined directions, and a link from nodes $v_i$ to $v_j$ is denoted interchangeably by either $e(v_i, v_j) \in E$ or $i \to j$. Let $i : i \to j$ and $k : j \to k$ denote the set of predecessors and successors of node $v_j$ in the directed graph, respectively.

We focus on the notion of a system operating point (SOP) [43]. A node in this model has a task to integrate renewables, loads, and in-and-out power flows into a smart grid. The node is also responsible for making a decision to balance the supply and load at every moment, with the help of installed storage devices, e.g., batteries. Storage systems are significantly able to save generation cost without wasting consumers' facilities, by charging or discharging to balance the surplus or shortfall in generation compared to load. Then, we define the following variables in the power network.

- $b_j(t)$: The node energy that is the battery energy level at node $v_j \in V$ at time $t \in \mathcal{T} := \{1, 2, ...\}$.
- $p_j(t)$: The net power export that is discharging process at node $v_j \in V$ at time $t$.
- $l_j(t)$: The total load of node $v_j$ at time $t$.

- $r_j(t)$: The amount of real power injection of node $v_j$ at time $t$ that is provided by renewable energy resources.
- $g_j(t)$: The amount of real power injection of node $v_j$ at time $t$ provided from *Centralized Generation Facility (CGF)* that utilizes resources except renewable resources, such as fossil fuel (coal, gas powered) or nuclear.
- $f_{ij}(t)$: The real power flow on the link $e(v_i, v_j) \in E$.

We use variables without subscripts to denote vectors with the proper components. For example, $b = (b_j(t), v_j \in V \text{ and } t \in \mathcal{T})$ denotes the vector of node battery energy levels. First we define the net power export as

$$p_j(t) = d_j(t) - g_j(t) + F_j(t), \quad \text{for } v_j \in V, \tag{3.1}$$

where $d_j(t) := l_i(t) - r_j(t)$ is called the *net demand*, and

$$F_j(t) := \sum_{k:j\to k} f_{jk}(t) - \sum_{i:i\to j} f_{ij}(t) \tag{3.2}$$

is the net power flow from node $v_j$ to other nodes at time $t$.

Then, the battery energy levels $b_j(t)$ follow the dynamics as

$$b_j(t) = b_j(t-1) - p_j(t)\Delta\tau, \quad \text{for } v_j \in V, \tag{3.3}$$

where $\Delta\tau$ is an actual time interval (hour) between $t-1$ and $t$. Note that the battery energy levels are bounded by finite capacities as

$$0 \le b_j(t) \le \bar{b}_j, \quad \text{for } v_j \in V. \tag{3.4}$$

Also, the CGFs are subject to generation capacity constraints

$$0 \leq g_j(t) \leq \overline{g}_j, \quad \text{for } v_j \in V, \tag{3.5}$$

and, due to thermal and safety limits, the power flows on the lines are subject to capacity constraints

$$-\overline{f}_{ij} \leq f_{ij}(t) \leq \overline{f}_{ij}, \quad \text{for } e(v_i, v_j) \in E. \tag{3.6}$$

When a *Renewable Penetration Rate (RPR)* is less than 100%, $b_j(t)$ decreases to 0 unless we appropriately set a boundary condition for a node energy with controllable sources of CGF and power flows. It is risky that all the node energies converge on 0 as $b_j(t)$ can be smaller than 0 due to sudden load peaks which may eventually lead to blackout. To deal with unexpected net demands that may lead to $b_j(t) < 0$, we use a minimum energy level $b^* > 0$ in a boundary condition of a node energy as in

$$b^* \leq b_j(t) \leq \overline{b}_j, \quad j \in V. \tag{3.7}$$

We want to balance all the node energy levels $b_i(t)$ at all times to minimize the total amount of power by CGF $g_i(t)$. As we assume a power network is lossless,

$$\sum_{v_i \in V} F_i(t) = 0, \tag{3.8}$$

the total amount of all the node energies $\sum_{v_i \in V} b_i(t)$ does not change with edge flows. Therefore, the balancing problem is considered as a min-max problem about node energies defined as follows:

### 3.1.2 Objective Function

Given a minimum energy level $b^*$, initial node energies $b_i(0)$, battery capacities $\overline{b}_i$, loads $l_i(t)$, and renewable generations $r_i(t)$ for $v_i \in V$, link capacities $\overline{f}_{ij}$ for $e(v_i, v_j) \in E$, and a given time sequence $\mathcal{T} = \{t\}$, the ORDER problem is

$$\begin{aligned} \text{Minimize} \quad & \max_{v_i \in V} \{b_i(t)\} \\ \text{Over} \quad & f, F, g, p \\ \text{Subject to} \quad & (3.1), (3.2), (3.3), (3.5), (3.6), (3.7), (3.8). \end{aligned} \tag{3.9}$$

With the objective function (3.9), the total amount of power provided from CGF is minimized as analyzed in Section 3.3.5. The objective function (3.9) can be quantified using the average deviation of all the node energies of $G$. The average deviation at $t$ is defined as

$$\delta(t) = \frac{\sum_{v_i \in V} |b_i(t) - \overline{B}(t)|}{|V|}, \tag{3.10}$$

where

$$\overline{B}(t) = \frac{\sum_{v_i \in V} b_i(t)}{|V|}. \tag{3.11}$$

To solve the ORDER problem, we minimize the average deviation by adjusting $f$ and $g$, thereby $F$ and $p$.

## 3.2 Decentralized Model for ORDER

### 3.2.1 ORDER in Tie-set

*ORDER in Tie-set (ORDER-T)* is an optimization procedure in a tie-set $L_i$ that decides $g_i(t)$ in $v_i \in V(L_i)$ and $f_{ij}(t)$ on $e(v_i, v_j) \in L_i$ at time $t$ to minimize $\max_{v_i \in V(L_i)} \{b_i(t)\}$ with conditions (3.3), (3.1), (3.2), (3.5), (3.6), (3.7), and (3.8).

A decentralized algorithm for ORDER-T is described in Section 3.2.4 in detail. Here, we briefly explain ORDER-T using the examples $L_1$ and $L_4$ in Figure 3.1. In Figure 3.1, let us assume that $b^* = 40$ MWh (Megawatt Hour), $\overline{b}_i = 100$ MWh, and $\overline{f}_{ij} = 50$ MW (Megawatt).

Let $b_i^d(t)$ be a *net node energy* decided by net demands $d_i(t)$ (uncontrollable processes) without CGF power $g_i(t)$ and net power flow $F_i(t)$ as follows:

$$b_i^d(t) := b_i(t-1) - d_i(t)\Delta\tau. \tag{3.12}$$

In this example, $\Delta tau$ is 1 hour so that we do not have to distinguish power and energy. In $L_1$ in the left network of Figure 3.1, current node energies are $b_a^d(t) = 24$ MWh, $b_b^d(t) = 87$ MWh, and $b_c^d(t) = 54$ MWh at time $t$. As edge flows in $L_1$ of the right side of Figure 3.1 are $f_{ab}(t) = -31$ MW ($f_{ba}(t) = 31$ MW), $f_{bc}(t) = 1$ MW ($f_{cb}(t) = -1$ MW), and $f_{ca}(t) = f_{ac}(t) = 0$ MW, node energies are updated as $b_a(t) = 24 - ((-31) + 0) = 55$ MWh, $b_b(t) = 87 - (31 + 1)) = 55$ MWh, and $b_c(t) = 54 - ((-1) + 0)) = 55$ MWh solving the min-max function (3.25) in $L_1$. Note that in $L_1$, each $g_i(t) = 0$ MW where $v_i \in V(L_1)$, as all the nodes in $L_1$ satisfy the conditions of ORDER-T.

In $L_4$ in the left network of Figure 3.1, current node energies are $b_e^d(t) = 48$ MWh, $b_f^d(t) = 26$ MWh, $b_g^d(t) = 50$ MWh, and $b_h^d(t) = 3$ MWh at time $t$ in $L_4$. Let the power from CGF of a

node $h$ be $g_h(t) = 33$ MW to satisfy the conditions of the ORDER in $L_4$. Thereby, $b_h(t)$ is updated as $b_h(t) = 3 + 33 = 36$ MWh. As flows in $L_4$ are $f_{eg}(t) = 8$ MW, $f_{gh}(t) = 18$ MW, $f_{hf}(t) = 14$ MW, and $f_{fe}(t) = 0$ MW, node energies are updated as $b_e(t) = 48-(0+8) = 40$ MWh, $b_f(t) = 26 - (0 + (-14)) = 40$ MWh, $b_g(t) = 50 - ((-8) + 18) = 40$ MWh, and $b_h(t) = 36 - ((-18) + 14) = 40$ MWh. With $g_h(t)$ and $f$ in $L_4$, the ORDER-T is completed.

### 3.2.2 Nature of Iteration of ORDER in Tie-set

Let $\overline{B}_l(t)$ be the average value of node energies in $L_i$ as follows:

$$\overline{B}_l(t) = \frac{\sum_{v_i \in V(L_i)} b_i(t)}{|V(L_i)|} \tag{3.13}$$

Here, we consider $b_i(\tau), v_i \in V$ that is a scalar derived from a snapshot of a fluctuating $b_i(t)$ in a power network at a given time $t = \tau$. With the following additional condition

$$\overline{B}_l(\tau) \leq \min_{v_i \in V} \{\bar{b}_i\} \tag{3.14}$$

the ORDER-T can be considered as *Circuit Equalization* defined in [27], where $b_i(\tau)$ at $v_i \in V(L_i)$ is averaged to $\overline{B}_l(\tau)$ in a circuit formed by a tie-set $L_i$. Then, the following theorem holds true, which is theoretically verified in [27, 28].

*Theorem 1:* For a given bi-connected graph $G = (V, E)$ with a scalar value $b_i(\tau)$ assigned to each node $v_i \in V$ and an arbitrary small positive real number $\epsilon$, iteration of ORDER-T make the average deviation of $G$ less than $\epsilon$ as follows:

$$\delta(\tau) < \epsilon \tag{3.15}$$

Theorem 1 is also verified by simulation experiments in [29]. We apply this theorem to

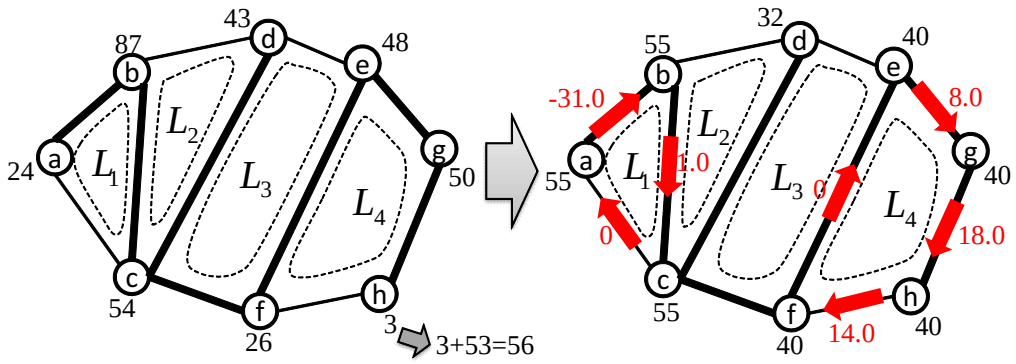



Figure 3.1: Parallel optimization of ORDER-T in $L_1$ and $L_4$.

designing an autonomous distributed control model to solve the ORDER problem where $b_i(t)$ constantly fluctuates due to the random processes of loads and renewables.

### 3.2.3 TADiC for ORDER

In the ORDER problem, we use the average deviation of node energies $b_i(t)$ in a tie-set as TEF, namely,

$$\Phi(L_i, t) = \frac{\sum_{v_i \in V(L_i)} |b_i(t) - \overline{B}_l(t)|}{|V(L_i)|}, \tag{3.16}$$

where $\overline{B}_l(t)$ is defined in (3.13).

At time $t$, TA of each tie-set brings the information of $b_i(t)$ at the nodes in the tie-set, and TEF is calculated. Suppose the leader node of $L_1, L_2, L_3$, and $L_4$ in Figure 3.1 is are $a, b, d$, and $e$, respectively. Those leader nodes have received information of (net) node energies at time $t$. For example, the leader $a$ of $L_1$ receives $b_a(t) = 24$, $b_b(t) = 87$, and $b_c(t) = 54$ at $t$.

The average node energy at each tie-set $\overline{B}_l(t)$ in Figure 3.1 respectively is

$$
\begin{aligned}
L_1: &\quad \overline{B}_l(t) = (24 + 87 + 54)/3 = 55, \\
L_2: &\quad \overline{B}_l(t) = (87 + 54 + 43)/3 = 61.333..., \\
L_3: &\quad \overline{B}_l(t) = (54 + 43 + 48 + 26)/4 = 42.75, \\
L_4: &\quad \overline{B}_l(t) = (48 + 26 + 50 + 3)/4 = 31.75.
\end{aligned}
$$

Then, TEF values of tie-sets are calculated as follows:

$$
\begin{aligned}
\Phi(L_1, t) &= (|24 - 55| + |87 - 55| + |54 - 55|)/3 = 21.33, \\
\Phi(L_2, t) &= (|87 - 61.33...| + |54 - 61.33...| + |43 - 61.33...|)/3 = 17.11..., \\
\Phi(L_3, t) &= (|54 - 42.75| + |43 - 42.75| + |48 - 42.75| + |26 - 42.75|)/4 = 10.75, \\
\Phi(L_4, t) &= (|48 - 31.75| + |26 - 31.75| + |50 - 31.75| + |3 - 31.75|)/4 = 17.25.
\end{aligned}
$$

Each tie-set exchange the TEF above with adjacent tie-sets to decide which tie-set(s) conduct optimization. As the TEFs of $L_1$ and $L_2$ are larger than their adjacent tie-sets, the TFs of $\{\zeta(L_1), \zeta(L_2), \zeta(L_3), \zeta(L_4)\}$ are $\{1, 0, 0, 1\}$ so that $L_1$ and $L_4$ are in process in Figure 3.1.

### 3.2.4 Decentralized Algorithm for ORDER in Tie-set

*Decentralized Algorithm for ORDER-T (DA-ORDER)* is described in Algorithm 2 and called by the step "OPTIMIZE" of TADiC. DA-ORDER calculates the optimal edge flows $f$ and the amount of CGF power $g$ in the leader $v_l^i$ of $L_i$ to realize the balanced distribution of renewables.

**Algorithm 2** Decentralized Algorithm for ORDER-T

**STEP 0:** Initialzation
  $b_i^d(t) := b_i(t-1) - d_i(t)\Delta\tau$.
  **for** each $v_i \in V(L_i)$ **do**
    $b_i(t-1)$ is preserved in $v_l^i$ of $L_i$ at $t$.
    $l_i(t)$ and $r_i(t)$ are provided by TA.
    Set each $g_i(t) = 0$.
  **end for**
  For each $e(v_i, v_j) \in L_i$, set $f_{ij}(t) = 0$.
**STEP 1:** Deciding Power Supply from CGF
  Calculate $\overline{B}_l^d(t) = \frac{\sum_{v_i \in V(L_i)} b_i^d(t)}{|V(L_i)|}$.
  **if** $\overline{B}_l^d(t) < b^*$ **then**
    $\mathcal{D} = |V(L_i)| \times (b^* - \overline{B}_l^d(t))$.
    **while** $\mathcal{D} \neq 0$ **do**
      Select $v_i \in V(L_i)$, where $b_i^d(t) < \bar{b}_i$.
      **if** $\bar{b}_i - b_i^d(t) \geq \mathcal{D}$ **then**
        $g_i(t) = \mathcal{D}$.
        $\mathcal{D} = 0$.
      **else** $\{\bar{b}_i - b_i^d(t) < \mathcal{D}\}$
        $g_i(t) = \bar{b}_i - b_i^d(t)$.
        $\mathcal{D} \leftarrow \mathcal{D} - \bar{b}_i + b_i^d(t)$.
      **end if**
    **end while**
  **end if**
**STEP 2:** Deciding Edge Flows
  Calculate $\overline{B}_l(t) = \frac{\sum_{v_i \in V(L_i)} b_i(t)}{|V(L_i)|}$.
  **for** each $v_i \in V(L_i)$ **do**
    **if** $b_i(t) \geq \overline{B}_l(t)$ **then**
      $X \leftarrow X \cup \{v_i\}$.
    **else** $\{b_i(t) < \overline{B}_l(t)\}$
      $Y \leftarrow Y \cup \{v_i\}$.
    **end if**
  **end for**
  **while** $X \cup Y \neq \emptyset$ **do**
    Select $x \in X$ and $y \in Y$ arbitrarily.
    **if** $b_x(t) - \overline{B}_l(t) > \overline{B}_l(t) - b_y(t)$ **then**
      $f_{P(x,y)}(t) = \overline{B}_l(t) - b_y(t)$.
      $Y \leftarrow Y - \{y\}$.
    **else if** $b_x(t) - \overline{B}_l(t) < \overline{B}_l(t) - b_y(t)$ **then**
      $f_{P(x,y)}(t) = b_x(t) - \overline{B}_l(t)$.
      $X \leftarrow X - \{x\}$.
    **else** $\{b_x(t) - \overline{B}_l(t) = \overline{B}_l(t) - b_y(t)\}$
      $f_{P(x,y)}(t) = b_x(t) - \overline{B}_l(t)$.
      $X \leftarrow X - \{x\}$, $Y \leftarrow Y - \{y\}$.
    **end if**
    Edge flows $f$ of $f_{P(x,y)}(t)$ are calculated according to (3.17), (3.18).
    Node energies are updated with $f$ according to (3.3) and (3.1).
  **end while**
**STEP 3:** Distributing Edge Flows
  With a set of path flows $\{f_{P(x,y)}(t)\}$, calculate $f_{ij}(t)$ on $e(v_i, v_j) \in L_i$ that satisfies (3.6).
  Distribute edge flows $f$ in $L_i$.

*STEP 0:* Step 0 is an initialization procedure. $b_i^d(t)$ is defined in (3.12). For each node $v_i$ in a tie-set $L_i$, a node energy from a previous time step $b_i(t-1)$ has been preserved in a leader node $v_l^i$ of $L_i$ at time $t$. In addition, $l_i(t)$ and $r_i(t)$ are provided by TA, and $g_i(t)$ is set as 0. For each $e(v_i, v_j) \in L_i$, the edge flow is initialized as 0.

*STEP 1:* In Step 1, CGF powers $g_i(t)$ at nodes in $L_i$ are decided. If the average value $\overline{B}_l^d(t)$ is smaller than $b^*$, the CGF provides the tie-set $L_i$ with power in the amount of $\mathcal{D} = |V(L_i)| \times (b^* - \overline{B}_l^d(t))$. When each $g_i(t)$ in $L_i$ is decided with $\mathcal{D}$, $b_i(t) \leq \overline{b}_i$ is satisfied.

*STEP 2:* With the average node energy $\overline{B}_l(t)$ in a tie-set defined in (3.13), let $X$ be a set of nodes where $b_i(t) \geq \overline{B}_l(t)$, and $Y$ be a set of nodes where $b_i(t) < \overline{B}_l(t)$. Then, each node of $V(L_i)$ is classified into $X$ or $Y$.

In the Algorithm 2, let $f_{P(x,y)}(t)$ be a path flow on a path $P(x, y)$ from node $x$ to $y$ at time $t$. Then, a path flow $f_{P(x,y)}(t)$ is defined as follows:

$$f_{P(x,y)}(t) = \{f_{x0}(t), f_{01}(t), ..., f_{hy}(t)\} \tag{3.17}$$

If the value of $f_{P(x,y)}(t)$ on the path $P(x, y)$ is decided as $\mathcal{F}$, each edge flow of $f_{P(x,y)}(t)$ is updated as follows:

$$f_{P(x,y)}(t) = \mathcal{F} \Longrightarrow \begin{cases} f_{x0}(t) \leftarrow f_{x0}(t) + \mathcal{F} \\ f_{01}(t) \leftarrow f_{01}(t) + \mathcal{F} \\ \qquad\qquad \vdots \\ f_{hy}(t) \leftarrow f_{hy}(t) + \mathcal{F} \end{cases} \tag{3.18}$$

In Step 2, a set of optimal path flows $\{f_{P(x,y)}(t)\}$ is calculated so that all the node energies of $V(L_i)$ are balanced where $b_i(t)$ becomes the same as $\overline{B}_l(t)$. The node energies $b_i(t)$ in $L_i$ are updated according to $f_{ij}(t)$ on $e(v_i, v_j) \in L_i$.

*STEP 3:* In Step 3, each edge flow is calculated based on the set of path flows $\{f_{P(x,y)}(t)\}$ decided in Step 2 to satisfy the condition (3.6). Then, those flows are distributed in $L_i$.

## 3.3 Simulation and Experiments for ORDER

A simulator is written in Java to realize the distributed algorithms to solve the ORDER problem and to conduct experiments. In a bi-connected power network, power flows can pass bi-directionally even if links are set to be directed. Each node employs common buffering method and polling method.

Every measurement vector $y_i(t)$ is sent to the leader node of each tie-set constantly. Accumulated data of $l_i(t)$ and $r_i(t)$ from smart meters are sent to a leader node of a tie-set with a 15-minute interval as in [44, 45]. The load $l_i(t)$ and renewable generation $r_i(t)$ are respectively bounded by $\overline{l}_i$ and $\overline{r}_i$ as

$$0 \leq l_i(t) \leq \overline{l}_i, \quad 0 \leq r_i(t) \leq \overline{r}_i.$$

They are randomly assigned as real numbers for a given time sequence ($\mathcal{T} = \{0, 15, 30, ...\}$). We uniformly set each maximum load at node $v_i$ as $\overline{l}_i = 20$ MW. The communication interval $\Delta t$ in STAND-BY of TADiC is defined as 1 min (minute). In this experiment, $b^*$ and $\overline{b}_i$ are set as $b^* = 40$ MWh and $\overline{b}_i = 100$ MWh, respectively. The initial power $b_i(0)$ of each node is randomly assigned between $b^*$ and $\overline{b}_i$ as a real number. The maximum flow (capacity) on each edge is $\overline{f}_{ij} = 50$ MW. Each tie-set exchanges $\Phi(L_i, t)$ (Deviation) as TEF.

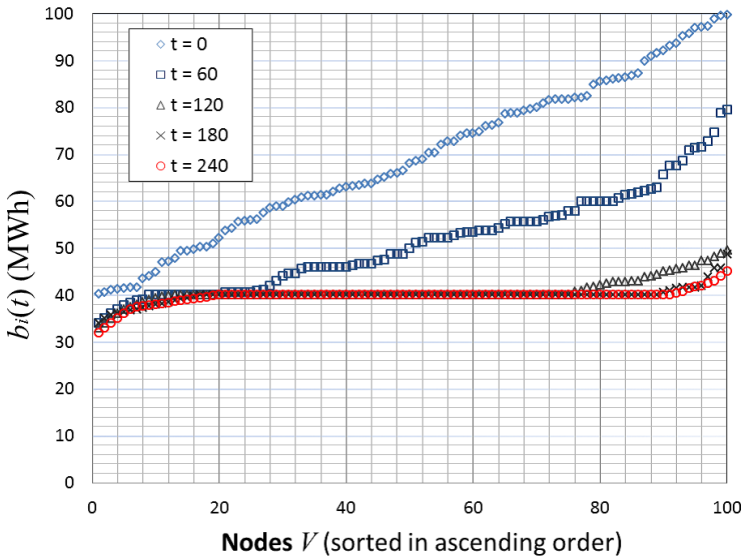



Figure 3.2: Convergence of $b_i(t)$ in a 100-node network at $t = 0, 60, 120, 180$, and 240 (min).

### 3.3.1 Behavior of Overall Convergence

In this experiment, a simulation network $G = (V, E)$ is created with 100 nodes and 190 links with random connections. The number of tie-sets is 91 and the height of a tree is 6. The maximum renewable generation is $\overline{r}_i = 10$ MW, where a penetration rate of renewables is intended to be 50% at each node as $\overline{l}_i = 20$ MW.

Figure 3.2 shows the behavior of overall convergence of all the node energies in a 100-node network by executing the proposed algorithms where simulation data of $b_i(t)$ at each node $v_i \in V$ is taken at $t = 0, 60, 120, 180, 240$ (min). Every $b_i(t)$ is sorted in ascending order in Figure 3.2. As indicated in Figure 3.2, each node energy $b_i(t)$ gradually converges on $b^*$ by iterative optimizations in each tie-set. When $t = 240$ min, the average value is $\overline{B}(t) = 39.69$ MWh; thereby all of the energies $b_i(240)$ almost converge on $b^*$. The average deviation at $t = 120$ min and 240 min is $\delta(120) = 1.842$ and $\delta(240) = 0.911$, respectively. This result demonstrates that most of the energies quickly converge on a near-optimal solution within a short time, and then all of them become close to the optimal value as $t$ increases.

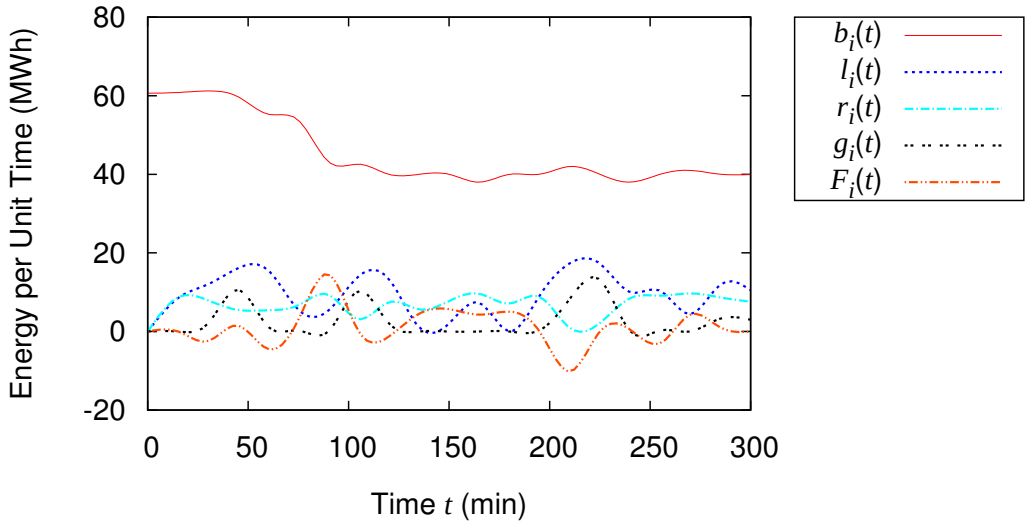



Figure 3.3: Changing process of $b_i(t)$, $l_i(t)$, $r_i(t)$, $g_i(t)$, and $F_i(t)$ from $t = 0$ to 300.

### 3.3.2 Convergence Behavior of Node Energy

#### Analysis of Behavior at Node

In this experiment, we look at the simulated behavior at a typical node $v_i$. A simulation network $G = (V, E)$ is created with 100 nodes and 190 links with random connections. The number of tie-sets is 91 and the height of a tree is 6. The maximum renewable generation is $\overline{r}_i = 10$ MW, where a penetration rate of renewables is intended to be 50% at each node as $\overline{l}_i = 20$ MW.

Figure 3.3 shows the changing process of $b_i(t)$, $l_i(t)$, $r_i(t)$, $g_i(t)$, and $F_i(t)$ from $t = 0, 15, ..., 300$ (min). As the proposed algorithms based on tie-sets are applied to the control of $b_i(t)$ with modest generation of $g_i(t)$, the value of $b_i(t)$ keeps to be around 40 MWh ($= b^*$) from $t \approx 100$. The total generation amount by CGF at $v_i$ is $\sum_{0 \le t \le 300} g_i(t) = 54.855$ MW. $v_i$ also distributes power flows to adjacent nodes as it has an excess amount of node energy. This result substantiates the effectiveness of the proposed autonomous control method in terms of sustainability as local optimizations by tie-sets to which each node belongs make

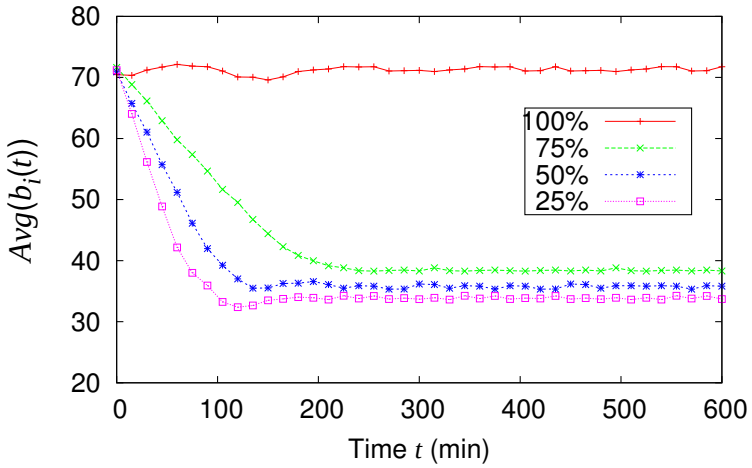



Figure 3.4: Convergence behavior of the average node energy $Avg(b_i(t)) = \overline{B}(t)$ with different RPRs (100%, 75%, 50%, 25%) in a 100-node network.

the demand response at the node quite stable.

Although only the simulation result at a node is introduced here, the results of other nodes are similar with a minor difference of the initial power $b_i(0)$ as it is assigned at random from 40 MWh to 100 MWh.

As to overall convergence of node energies, node energies $b_i(240)$ at $t = 240$ min almost converge on $b^*$ as the average value is $\overline{B}(t) = 39.69$ MWh. The average deviation at $t = 120$ min and 240 min is $\delta(120) = 1.842$ and $\delta(240) = 0.911$, respectively. This result demonstrates that most of the node energies quickly converge on a near-optimal solution within a short time, and then they keep close to the optimal value as $t$ increases.

**Convergence with Different Renewable Penetration Rates (RPRs)**

We conducted a simulation with different Renewable Penetration Rates (RPRs) as it is an important issue to be considered when creating future power networks with renewable integration. The RPR can be adjusted by changing the value of $\overline{r}_i$ in this simulation (load is fixed as $\overline{l}_i = 20$ MW). Simulation data of $b_i(t)$ of each node is taken at $t = 0, 15, 30, ..., 600$

(min). $Avg(b_i(t)) = \overline{B}(t)$ stands for the average value of all the node energies at time $t$ as in (3.10). Experiments are conducted 20 times for each case of RPR, and the average is calculated.

Figure 3.4 shows the convergence process in a 100-node network with expected different RPRs $\mathbb{E}\{r_i(t)/l_i(t)\}$ intended to be 100% ($\bar{l}_i = 20$ MW, $\overline{r}_i = 20$ MW), 75% ($\bar{l}_i = 20$ MW, $\overline{r}_i = 15$ MW), 50% ($\bar{l}_i = 20$ MW, $\overline{r}_i = 10$ MW), and 25% ($\bar{l}_i = 20$ MW, $\overline{r}_i = 5$ MW), respectively. As seen in Figure 3.4, the higher a RPR is, the closer to $b^*$ the convergence line becomes except $\mathbb{E}\{r_i(t)/l_i(t)\} = 100\%$. However, even though a RPR gets lower, the convergence becomes still stable from $t \approx 120$ (min) due to the convergence nature by iteration of ORDER-T. When the RPR is 100%, $b_i(t)$ does not converge on $b^*$; $b_i(t)$ converges on the average value of initial node energies $\overline{B}(0)$ instead of $b^*$ once the average deviation $\delta(t)$ becomes small. The total power provided from CGF also becomes stable after the node energies converge on $b^*$ or $\overline{B}(0)$.

We verified that the behavior of convergence in 200-node to 500-node networks with the different RPRs from 25% to 100% is quite similar to the one in the 100-node network above. In addition, we confirmed that the convergence process does not depend on the initial power at each node.

### 3.3.3 Centralized Generation with Different Renewable Penetration Rates

Next, we look at a changing process of the total power supplied from CGF $\sum_{v_i \in V} g_i(t)$ with time $t$ in a 100-node network. Simulation data of $\sum_{v_i \in V} g_i(t)$ is taken at $t = 0, 15, 30, ..., 600$ (min). Experiments are conducted 20 times for each case of RPR, and the average is calculated.

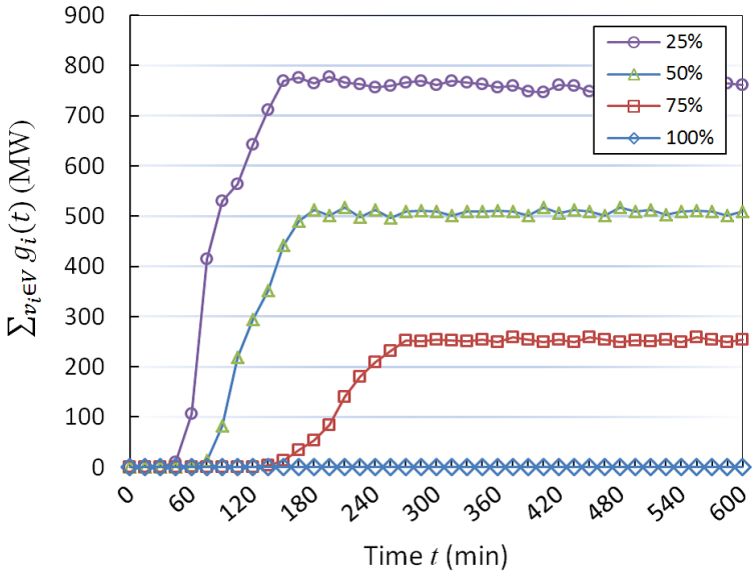



Figure 3.5: Changing process of $\sum_{v_i \in V} g_i(t)$ with different RPRs (25%, 50%, 75%, 100%) in a 100-node network.

Figure 3.5 shows the process of change of $\sum_{v_i \in V} g_i(t)$ with the different RPRs $\mathbb{E}\{r_i(t)/l_i(t)\}$ intended to be $25\%, 50\%, 75\%$, and $100\%$, respectively. As seen in Figure 3.5, the power provided from CGF becomes stable after the node energies converge on $b^*$ or $\overline{B}(0)$. For instance, in Figure 3.4, the average of all the node energies converge on $b^*$ from $t \approx 180$ when the RPRs are 25%, 50%, and 75%. Then, in Figure 3.5, the amount of power by CGF becomes stable from $t \approx 180$ when the RPRs are 25%, 50%, and 75% as well. This result demonstrates that the power supply from CGF becomes stable soon after the node energies converge by our proposed method. When the RPR is 100%, no power from CGF is provided. This is because the proposed method constantly distributes renewable energy resources in a balanced manner at all times. Thus, the proposed method is effective in the future power network where renewables are highly penetrated.

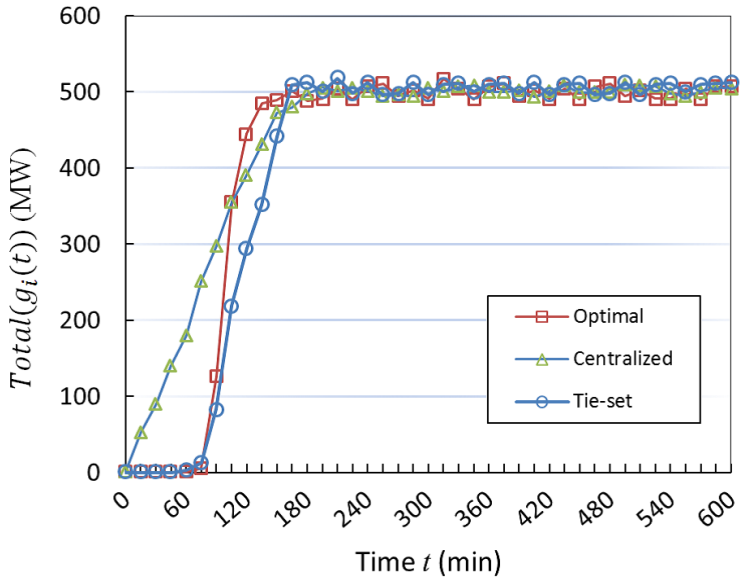



Figure 3.6: Comparison of total CGF power provision $Total(g_i(t)) = \sum_{v_i \in V} g_i(t)$ with the optimal line of the solution and the centralized provision when the RPR is 50%.

Table 3.1: Comparison of total CGF power provision $\sum_{t=0}^{600} \sum_{v_i \in V} g_i(t)$ (MW) with different RPRs

| RPR | 25% | 50% | 75% | 100% |
|---|---|---|---|---|
| Optimal | 26426.13 | 16714.97 | 6821.70 | 0 |
| Centralized | 27254.82 | 17688.10 | 7806.13 | 1994.07 |
| Tie-set | 26361.56 | 16583.44 | 6763.27 | 0 |

### 3.3.4 Comparison Experiments against Centralized Model

In this experiment, we assess the optimality of the proposed method by comparing its changing process of power supply from CGF ($\sum_{v_i \in V} g_i(t)$) with the theoretical solution of the ORDER problem and Centralized Provision without distribution of renewables $\sum_{e(v_i,v_j) \in E} f_{ij}(t)$ at node $v_i$ in (3.1), although the centralized control is intended to be replaced by distributed operations. In centralized provision, if a node energy $b_i(t)$ at time $t$ is smaller than $b^*$, CGF power with the amount of $g_i(t) = b^* - b_i(t)$ is supplied to the node $v_i$. The solution of ORDER is $\sum_{v_i \in V} g_i(t) = 0$ if $\overline{B}^d(t) \geq b^*$, and $\sum_{v_i \in V} g_i(t) = |V| \times (b^* - \overline{B}^d(t))$ if $\overline{B}^d(t) < b^*$ as analyzed in Section 3.3.5. Naturally, any optimization technique cannot excel in the solution. Therefore, it is important to analyze how the proposed technique is close to the theoretical limitation of the ORDER problem in terms of the power supply from CGF. The simulation network is the same as a 100-node network above with 50% of the RPRs ($\bar{l}_i = 20$ MW, $\overline{r}_i = 10$ MW). Simulation data of $\sum_{v_i \in V} g_i(t)$ is taken at $t = 0, 15, 30, ..., 600$ (min). Experiments are conducted 20 times and the average is calculated.

Figure 3.6 shows the changing process of $\sum_{v_i \in V} g_i(t)$ by the theoretical solution (Optimal), centralized provision (Centralized), and the proposed method (Tie-set). As indicated in Figure 3.6, the optimal line shows a faster convergence than the proposed method. However, the proposed technique also becomes stable just after the optimal solution became stable. On the other hand, the result of the centralized provision (Centralized) demonstrates the excessive power supply from the beginning of the operation as well as a little slower convergence compared with the one by the proposed method. Table 3.1 shows comparison of the total CGF power provision from $t = 0$ to 600 at all the nodes in $V$ ($\sum_{t=0}^{600} \sum_{v_i \in V} g_i(t)$) with different RPRs. The result indicates that the total CGF supply by Tie-set based optimization is almost the same as the theoretical solution. However, the amount of power supply by the centralized provision method (Centralized) is larger than the optimal amount. When the RPR is 100%, both of the theoretical solution and the proposed method demonstrated an

efficient distribution of DERs without any CGF power; the centralized method still provides power using fossil fuels to satisfy the conditions of the ORDER problem. This substantiates the effectiveness of the optimization in a system of $\mu$ independent tie-sets that quickly leads to global optimization.

### 3.3.5 Analysis on Solution of ORDER

We briefly analyze the optimal solution of $\sum_{v_i \in V} g_i(t)$ at time $t$ to minimize the power supply from CGF. According to the objective function (3.9) of ORDER problem, the optimal state of balanced $b_i(t)$ is that the average deviation (3.10) is minimized. Here, let $\overline{B}^d(t)$ be the average value of net node energies as follows:

$$\overline{B}^d(t) = \frac{\sum_{v_i \in V}(b_i(t-1) - d_i(t)\Delta\tau)}{|V|} \tag{3.19}$$

Then, the optimal solution of $\sum_{v_i \in V} g_i(t)$ at time $t$ is as follows:

$$\sum_{v_i \in V} g_i(t) = \begin{cases} 0, & \overline{B}^d(t) \geq b^* \\ |V| \times (b^* - \overline{B}^d(t)), & \overline{B}^d(t) < b^* \end{cases} \tag{3.20}$$

If all the node energies are not balanced; if the objective function 3.9 is not satisfied, there would be excess provision of power from CGF to ensure that every node $v_i$ has its power $b_i(t)$ at least $b^*$. Therefore, the optimal amount of power from CGF in 3.20 can be provided if each node energy is balanced with the objective function 3.9, which leads to minimizing the power from CGF.

## 3.4 ORDER with Real Power Flow

In this section, we explore more realistic power flow model where we consider real power flows focusing on voltages, admittances, phase angles, and so forth. By this change, power flows on different lines become dependent with Kirchhoff's laws. We explain the parallel optimization procedure with the redefined ORDER problem, and introduce simulation results that capture all the features of the reformulated power flow model.

### 3.4.1 Real Power Flow (RPF)

For any two vertices $v_i, v_j$ ($v_i \neq v_j$), we redefine $f_{ij}(t)$ as a real power flow over an edge $e(v_i, v_j)$ at time $t$. The real power flow from node $v_i$ to $v_j$ in a power network at time $t$ is

$$f_{ij}(t) = V_i V_j Y_{ij}(\theta_i(t) - \theta_j(t)), \quad e(v_i, v_j) \in E, \tag{3.21}$$

where $V_i$ is the voltage at node $v_i$, $\theta_i(t)$ is the phase angle at node $v_i$ at time $t$, and $Y_{ij} = Y_{ji}$ is the admittance on the link $e(v_i, v_j)$[1]. As voltages must be constantly maintained close to 1 (usually between 0.95 and 1.05), we consider $V_i = 1$ at each node $v_i$. As a stability constraint, the angle difference $\theta_i(t) - \theta_j(t)$ between two neighboring nodes satisfies the following condition:

$$-\frac{\pi}{2} \leq \theta_i(t) - \theta_j(t) \leq \frac{\pi}{2} \tag{3.22}$$

The power flow in this case is also bounded by $-\overline{f}_{ij} \leq f_{ij}(t) \leq \overline{f}_{ij}$ as in (3.6) and the CGF generation has its limitation as in (3.5).

Let $\mathcal{A}(v_i)$ denote the set of adjacent nodes of node $v_i$. From Kirchhoff's laws, the *net power*

[1]Here, we assume that $|\theta_i(t) - \theta_j(t)|$ is small enough to approximate $\sin(\theta_i(t) - \theta_j(t))$ by $|\theta_i(t) - \theta_j(t)|$.

*flow* from node $v_i$ to adjacent nodes at time $t$ is given by

$$\begin{aligned} F_i(t) &= \sum_{v_j \in \mathcal{A}(v_i)} f_{ij}(t) \\ &= \sum_{v_j \in \mathcal{A}(v_i)} V_i V_j Y_{ij}(\theta_i(t) - \theta_j(t)). \end{aligned} \tag{3.23}$$

If the net power flow at node $v_i$ is positive, $v_i$ supplies power at time $t$; otherwise $v_i$ demands power.

As discussed in (3.3) and (3.1), the node energy $b_i(t)$ at $v_i$ at time $t$ follows the dynamics as $b_i(t) = b_i(t-1) - (d_i(t) - g_i(t) + F_i(t))\Delta\tau$.

### 3.4.2 Lower Energy Limit (LEL)

A node energy $b_i(t)$ is bounded by a battery storage capacity where $0 \leq b_i(t) \leq \overline{b}_i$. To deal with unexpected load peaks, we define a *Lower Energy Limit (LEL)* $LB_i(t)$ that satisfies $0 < LB_i(t) < \overline{b}_i$. Then, the boundary condition of a node energy is

$$LB_i(t) \leq b_i(t) \leq \overline{b}_i, \quad v_i \in V, \tag{3.24}$$

with a given initial LEL $LB_i(0)$. Let $\overline{l}_i(t)$ be $\max_{t \in \mathcal{T}}\{l_i(t)\}$ among the load profile $\{l_i(0), l_i(1), ..., l_i(t)\}$ until time $t$ at node $v_i$. The value of $LB_i(t)$ at each time $t$ dynamically changes if the load peak at time $t$ is larger than the LEL form previous time, where $LB_i(t-1) < \overline{l}_i(t)$. In this case, the LEL becomes $LB_i(t) := \overline{l}_i(t)$. $\Delta B_i(t) = \overline{b}_i - LB_i(t)$ is preserved for unexpected renewable generation peak when $r_i(t)$ is larger than the load process $l_i(t)$. In this case, the net demand becomes negative as $d_i(t) < 0$, so that $|d_i(t)|$ is stored in the battery at node $v_i$.

### 3.4.3 Objective Function of ORDER with Real Power Flow

Given initial node energies $b_i(0)$, battery capacities $\overline{b}_i$, initial LELs $LB_i(0)$, loads $l_i(t)$ and renewable generations $r_i(t)$ at each time step $t$ for $v_i \in V$, edge capacities $\overline{f}_{ij}$ for $e(v_i, v_j) \in E$, and a given time sequence $\mathcal{T} = \{t\}$, the ORDER with Real Power Flow *(ORDER-RPF)* problem is

$$\begin{aligned} \text{Minimize} \quad & \sum_{t \in \mathcal{T}} \sum_{v_i \in V} g_i(t), \\ \text{Over} \quad & g, f, F, p, \theta, \\ \text{Subject to} \quad & (3.1), (3.3), (3.5), (3.6), (3.21), (3.22), (3.23), (3.24). \end{aligned} \tag{3.25}$$

In order to minimize the total amount power generated at CGFs at every time step, we consider the balanced allocation of DERs stored at batteries of nodes.

### 3.4.4 ORDER with Real Power Flow in Tie-set

*ORDER with Real Power Flow in Tie-set (ORDER-RT)* is a procedure in a tie-set $L_i$ that decides $g_i(t)$ and $\theta_i(t)$ in $v_i \in V(L_i)$ at time $t$ to minimize $\sum_{v_i \in V(L_i)} g_i(t)$ with conditions above.

As a decentralized algorithm for ORDER-RT is described in Section 3.4.5 in detail, we briefly explain ORDER-RT using the example of parallel optimization of $L_1$ and $L_4$ in Figure 3.7 here.

In the power network in Figure 3.7, we assume that each voltage is $V_i = 1$ (MV), each admittance is $Y_{ij} = 100$, and each battery capacity is $\overline{b}_i = 100$ MWh. Every minimum battery level is $LB_i(t) = 40$ MWh and every link capacity is $\overline{f}_{ij} = 50$ MW at time $t$. Angles

and CGFs are initialized as $\theta_i(t) = 0$ and $g_i(t) = 0$ MW.

In the upper network of Figure 3.7, the left values and the right values in brackets represent $b_i^d(t)$ (defined in (3.12)) and $\theta_i(t)$, respectively. In the network below of Figure 3.7, the left values in brackets stand for $b_i(t)$.

Given net demands at time $t$, current net node energies are $b_a^d(t) = 24.0$ MWh, $b_b^d(t) = 85.0$ MWh, and $b_c^d(t) = 31.0$ MWh in $L_1$ of the upper network of Figure 3.7. $\Delta\tau$ is 1 h at this time. Next, the angels of nodes $V(L_1)$ are calculated based on ORDER-RT in $L_1$, where $\theta_a(t) = 0.0$, $\theta_b(t) = 0.15$, and $\theta_c(t) = 0.015$. With the angels in $L_1$, power flows are decided as $f_{ab}(t) = -15.0$ MW ($f_{ba}(t) = 15.0$ MW), $f_{bc}(t) = 13.5$ MW ($f_{cb}(t) = -13.5$ MW), $f_{bd}(t) = 15.0$ MW ($f_{db}(t) = -15.0$ MW), $f_{ca}(t) = 1.5$ MW ($f_{ac}(t) = -1.5$ MW), $f_{cd}(t) = 1.5$ MW ($f_{dc}(t) = -1.5$ MW), and $f_{cf}(t) = 1.5$ MW ($f_{fc}(t) = -1.5$ MW) as in the network below of Figure 3.7. Then, node energies in $L_1$ are updated as $b_a(t) = 24.0 - ((-15.0) + (-1.5)) = 40.5$ MWh, $b_b(t) = 85.0 - (15.0 + 13.5 + 15.0) = 41.5$ MWh, and $b_c(t) = 31.0 - ((-13.5) + 1.5 + 1.5 + 1.5) = 40.0$ MWh. Therefore, in $L_1$, $g_i(t) = 0$ MW at each node $v_i \in V(L_1)$, since all the nodes in $L_1$ satisfy the conditions of ORDER-RT.

In $L_4$ in the upper network of Figure 3.7, current net node energies are $b_e^d(t) = 70.0$ MWh, $b_f^d(t) = 28.0$ MWh, $b_g^d(t) = 34.0$ MWh, and $b_h^d(t) = 30.0$ MWh at time $t$. The angels of nodes $V(L_4)$ are calculated by ORDER-RT in $L_4$, where $\theta_e(t) = 0.108$, $\theta_f(t) = 0.0$, $\theta_g(t) = 0.024$, and $\theta_h(t) = 0.0$. Thereby, power flows are decided as $f_{ed}(t) = 10.8$ MW, $f_{ef}(t) = 10.8$ MW, $f_{eg}(t) = 8.4$ MW, and $f_{gh}(t) = 2.4$ MW. Then, node energies in $L_4$ are updated as $b_e(t) = 70.0 - (10.8 + 10.8 + 8.4) = 40.0$ MWh, $b_f(t) = 28.0 - ((-1.5) + (-10.8)) = 40.3$ MWh, $b_g(t) = 34.0 - ((-8.4) + 2.4) = 40.0$ MWh, and $b_h(t) = 30.0 - (-2.4) = 32.4$ MWh. Next, the CGF supplies power to the node $h$ where $g_h(t) = 7.6$ MW to satisfy the condition that $b_h(t) \geq LB_h(t)(= 40)$. Therefore, $b_h(t)$ is updated as $b_h(t) = 32.4 + 7.6 = 40.0$ MWh. With $g(t)$ and $f(t)$ in $L_4$, the ORDER-RT is completed.

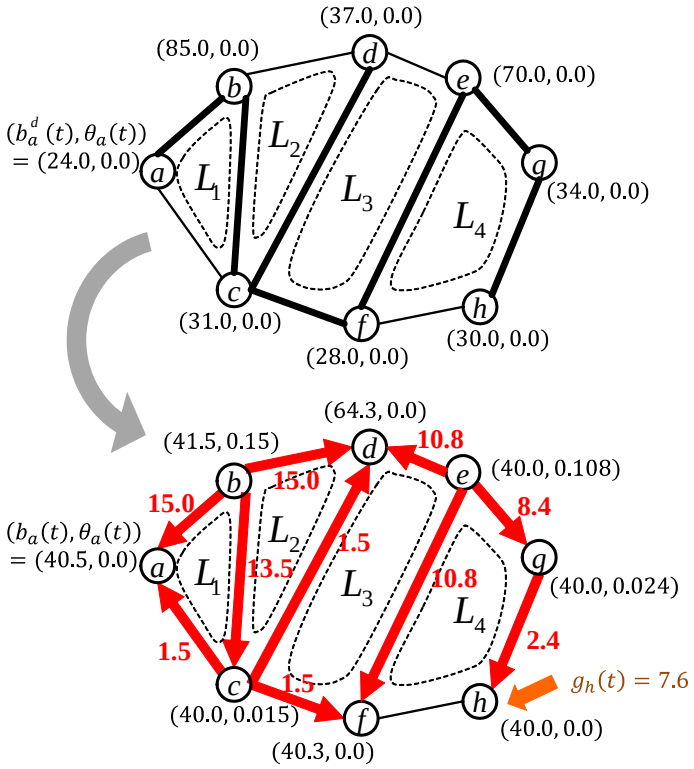



Figure 3.7: Parallel optimization of ORDER-RT in $L_1$ and $L_4$.

Note that the node energy $b_d(t)$ is also updated as $b_d(t) = 37.0 - ((-15.0) + (-1.5) + (-10.8)) = 64.3$ MWh.

### 3.4.5 Decentralized Algorithm for ORDER with RPF in Tie-set

Here, for each node $v_i \in V$, we define a *Battery Penalty Cost (BPC)* $H_i(t)$ as

$$H_i(t) := \begin{cases} 0, & \text{if } b_i(t) \geq LB_i(t) \\ \alpha_i(LB_i(t) - b_i(t)), & \text{if } b_i(t) < LB_i(t) \end{cases} \tag{3.26}$$

with some $\alpha_i > 0$. Then, TEF with BPC is defined as follows:

$$\Phi(L_i, t) = \sum_{v_i \in V(L_i)} H_i(t). \tag{3.27}$$

For instance, if BPC is used as TEF, the TEF values of tie-sets in the upper network in Figure 3.7 are

$$\begin{aligned}
\Phi(L_1, t) &= H_a(t) + H_b(t) + H_c(t) = 25.0, \\
\Phi(L_2, t) &= H_b(t) + H_c(t) + H_d(t) = 12.0, \\
\Phi(L_3, t) &= H_c(t) + H_d(t) + H_e(t) + H_f(t) = 24.0, \\
\Phi(L_4, t) &= H_e(t) + H_f(t) + H_g(t) + H_h(t) = 28.0,
\end{aligned}$$

as LELs of all nodes in the example network is set as $LB_i(t) = 40.0$. Therefore, the TFs of $\{\zeta(L_1), \zeta(L_2), \zeta(L_3), \zeta(L_4)\}$ are set as $\{1, 0, 0, 1\}$ so that $L_1$ and $L_4$ are in process in Figure 3.7.

*Decentralized Algorithm for ORDER-RT (DA-ORDER-RT)* is described in Algorithm 3 and called by OPTIMIZE of TADiC. DA-ORDER-RT calculates the optimal power flows $f$ and the amount of CGF power $g$ in the leader $v_l^i$ of $L_i$ to realize the balanced distribution of renewables.

*STEP 0:* Step 0 is an initialization procedure. For each node $v_i$ in a tie-set $L_i$, its angle $\theta_i(t)$ and CGF generation $g_i(t)$ are set as 0. A node energy from a previous time step $b_i(t-1)$ has been preserved in a leader node $v_l^i$ of $L_i$ at time $t$. As information of loads $l_i(t)$ and renewables $r_i(t)$ in $L_i$ are provided by TA, the net demand is assigned as $d_i(t) := l_i(t) - r_i(t)$. Then, the net node energy is calculated as $b_i^d(t) := b_i(t-1) - d_i(t)$. In addition, LEL $LB_i(t)$ is updated based on the load profile at $v_i$ where if $LB_i(t) < \bar{l}_i(t)$, $LB_i(t) := \bar{l}_i(t)$, otherwise LEL remains the same.

*STEP 1:* In Step 1, the angles of nodes in $L_i$ are calculated, and thus power flows on the edges are decided. We define a $flag = 0, 1$ to decide whether or not $L_i$ conducts the **while**

**Algorithm 3** Decentralized Algorithm for ORDER-RT

**STEP 0:** Initialzation

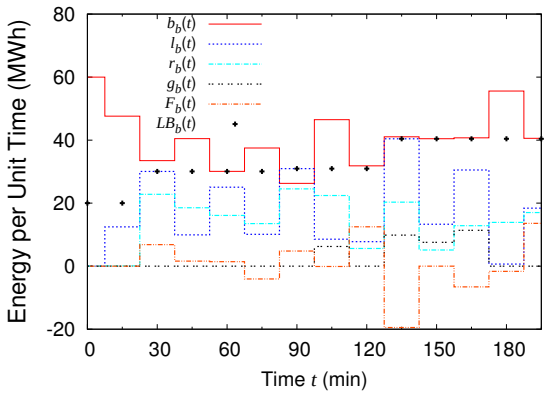
**for** each $v_i \in V(L_i)$ **do**
  $\theta_i(t) = 0$, $g_i(t) = 0$.
  $d_i(t) := l_i(t) - r_i(t)$.
  $b_i^d(t) := b_i(t-1) - d_i(t)\Delta\tau$.
  **if** $LB_i(t) < \bar{l}_i(t)$ **then**
    $LB_i(t) := \bar{l}_i(t)$.
  **end if**
**end for**

**STEP 1:** Deciding Angles for power flows

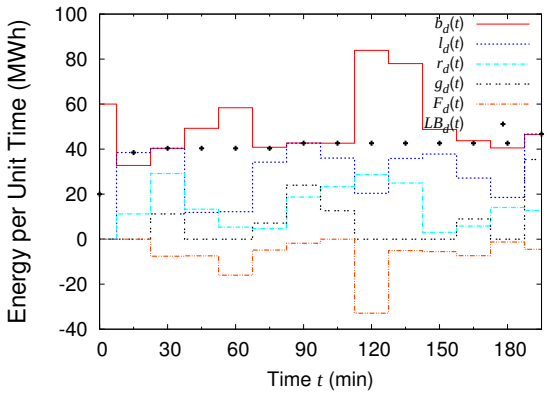
Set $falg = 1$.
**for** each $v_i \in V(L_i)$ **do**
  **if** $b_i^d(t) > LB_i(t)$ **then**
    $flag = 0$.
  **end if**
**end for**
**while** $flag \neq 1$ **do**
  Select $v_i \in V(L_i)$, where $b_i^d(t)$ is maximum.
  Calculate $\theta_i'(t) = \frac{b_i^d(t) - LB_i(t)}{\sum_{v_j \in \mathcal{A}(v_i)} V_i V_j Y_{ij}}$.
  Update $\theta_i(t) := \theta_i(t) + \theta_i'(t)$.
  **for** each $v_j \in \mathcal{A}(v_i)$ **do**
    Update $b_i^d(t) := b_i^d(t) - V_i V_j Y_{ij} \theta_i'(t)$.
    Update $b_j^d(t) := b_j^d(t) + V_i V_j Y_{ij} \theta_i'(t)$.
  **end for**
  Set $flag = 1$.
  **for** each $v_i \in V(L_i)$ **do**
    **if** $b_i^d(t) < LB_i(t)$ **then**
      Set $flag = 0$.
    **end if**
  **end for**
  $B_{max} := \max_{v_i \in V(L_i)} \{b_i^d(t)\}$.
  **if** $B_{max} - LB_i(t) \leq \epsilon$ **then**
    $flag = 1$.
  **end if**
**end while**
Set angles $\theta_i(t)$ for all nodes $v_i \in V(L_i)$ with (3.22), (3.6).
Distribute power flows in $L_i$ according to (3.21).

**STEP 2:** Deciding Power Supply from CGF

**for** each $v_i \in V(L_i)$ **do**
  **if** $b_i(t) < LB_i(t)$ **then**
    $g_i(t) = LB_i(t) - b_i(t)$.
    Supply CGF power to $v_i$ with the amount of $g_i(t)$.
  **end if**
**end for**




(a) Performance at node $b$ in Figure 3.7.




(b) Performance at node $d$ in Figure 3.7.

Figure 3.8: Case studies for performance over time in facing with different levels of uncertainty.

loop, which is from line 15 to 33 of Algorithm 3. If there is a node in $L_i$ that has power that can be distributed to other nodes, that is, $b_i^d(t) > LB_i(t)$, the $flag = 0$. Otherwise, $L_i$ goes to Step 2 without distributing node energies. In the **while** loop, phase angles are calculated in order to balance the node energies. The node with maximum power that meets $b_i^d(t) > LB_i(t)$ distributes its power to other nodes until it reaches LEL $LB_i(t)$. The **while** procedure stops if all the nodes in $L_i$ store power greater than or equal to LEL where $b_i^d(t) \geq LB_i(t)$ is satisfied. The **while** procedure also stops if energiesof the nodes that meet $b_i^d(t) > LB_i(t)$ converge on its LEL $LB_i(t)$, where a certain small value $\epsilon$ is used to check the convergence.

*STEP 2:* Step 2 decides the amount of power generated at CGF $g_i(t)$ at nodes in $L_i$. With the LEL $LB_i(t)$ at each node in a tie-set, if node $v_i$ does not have minimum level of power as in $b_i(t) < LB_i(t)$, then CGF supplies power to the node $v_i$ with the amount of $g_i(t) = LB_i(t) - b_i(t)$.

### 3.4.6 Case Study for Decentralized Model

In order to show the performance of the decentralized model in facing with different levels of uncertainty that are seen in load processes and renewable generation, we discuss some case studies on the performance of node $b$ and $d$ of the network in Figure 3.7, which are shown in Figure 3.8(a) and 3.8(b). At time $t = 0$, node energy and LELs on nodes $b$ and $d$ are given with $b_b(0) = b_d(0) = 60$ and $LB_b(0) = LB_d(0) = 20$, respectively.

As in the case studies, nodes $b$ and $d$ update LELs $LB_b(t)$ and $LB_d(t)$ according to the sudden load peaks. For instance, at $t = 30$ and 135, LELs $LB_b(30)$ and $LB_b(135)$ at node $b$ are updated as there are sudden load peaks. This procedure demonstrates that those nodes deal with the uncertainty of uncontrollable resources by changing the LELs in order to prevent running out of power.

Power flows between the nodes $b$ and $d$ on the case studies also demonstrate the framework that each node distributes power to the other peripheral nodes when a node has enough power, that is, $b_i(t) > LB_i(t)$. At the same time, the node that is in short of power ($b_i(t) < LB_i(t)$) receives incoming power flows with negative values ($F_i(t) < 0$) from the nodes that have the excess amount of power. For example, at $t = 30$, node $b$ had enough power from the previous time step $t = 15$, $b$ distributes its power to the adjacent nodes $a$, $c$, and $d$. The portion of the power from $b$ is sent to node $d$ as seen in the incoming power flow $F_d(30)$, which satisfies the net demand $d_d(30) = l_d(30) - r_d(30)$ together with $g_d(30)$. Therefore, the node $d$ makes up for the first load peak that happened from $t = 15$ to 30.

The process of CGF constantly compensates the net demand that has been partially satisfied by incoming power flows. As in the case studies, although node $b$ obtains incoming power flows at $t = 135$, $b$ still needs more power to meet $b_b(135) \geq LB_b(135)$. In order to satisfy the demand, CGF supplies power to $b$ with the amount of $g_b(135)$. There also are sudden renewable peak as indicated in the Figures 3.8(a) and 3.8(b). For example, when $t = 45$, 75,

105, and 180 at node $b$, the renewable generation is higher than the load process.

The amount of the sudden renewable generation is also stored in the node, and then distributed to the adjacent nodes at the next time step. This also substantiates the effectiveness of the model as the excess generation by renewables can be temporarily stored at the node, and then delivered to the other nodes that require power to satisfy their net demands.

## 3.5 Experiments for ORDER with RPF

Now we verify the distributed algorithms for ORDER with RPF.

A simulation network $G = (V, E)$ is created with 100 nodes and 190 links with random connections. The number of tie-sets is 91 and the height of a tree is 6. Every measurement vector $y_i(t)$ is sent to the leader node of each tie-set constantly. Accumulated data of $l_i(t)$ and $r_i(t)$ from smart meters are sent to a leader node of a tie-set with a 15-minute interval as in [44, 45]. Therefore, at each node $v_i$, $l_i(t)$ and $r_i(t)$ are respectively given for a given time sequence ($\mathcal{T} = \{0, 15, 30, ...\}$). The communication interval $\Delta t$ in TADiC is defined as 1 min (minute). In this experiment, every voltage is kept to be $V_i = 1$ MV, and each admittance is set to be $Y_{ij} = 100$. Initial LELs $LB_i(0)$ and $\overline{b}_i$ are set as $LB_i(0) = 40$ MWh and $\overline{b}_i = 100$ MWh, respectively. The initial power $b_i(0)$ of each node is assigned as $b_i(0) = 60$ MWh. The maximum flow (capacity) on each edge is $\overline{f}_{ij} \equiv 50$ MW. $\alpha_i$ of BPC defined in (3.26) is set as 1.

### 3.5.1 Power Stimulus Response at Node

In this experiment, we look at the simulated behavior of power stimulus response by picking up several nodes with different load and renewable profiles.

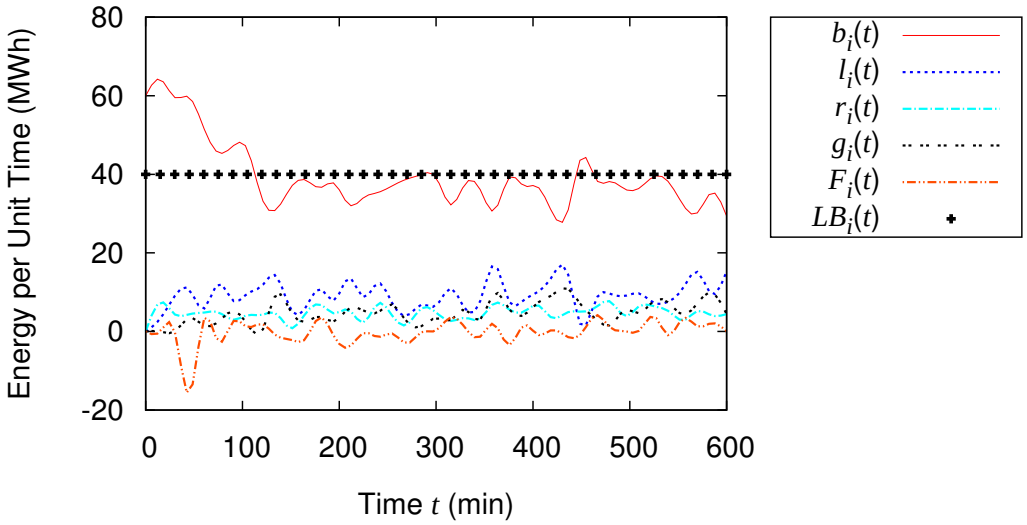



Figure 3.9: Changing process of $b_i(t)$, $l_i(t)$, $r_i(t)$, $g_i(t), F_i(t)$, and $LB_i(t)$ from $t = 0$ to 600 where load process and renewable generation is given with $0 \leq l_i(t) \leq 20$ and $0 \leq r_i(t) \leq 10$, respectively.

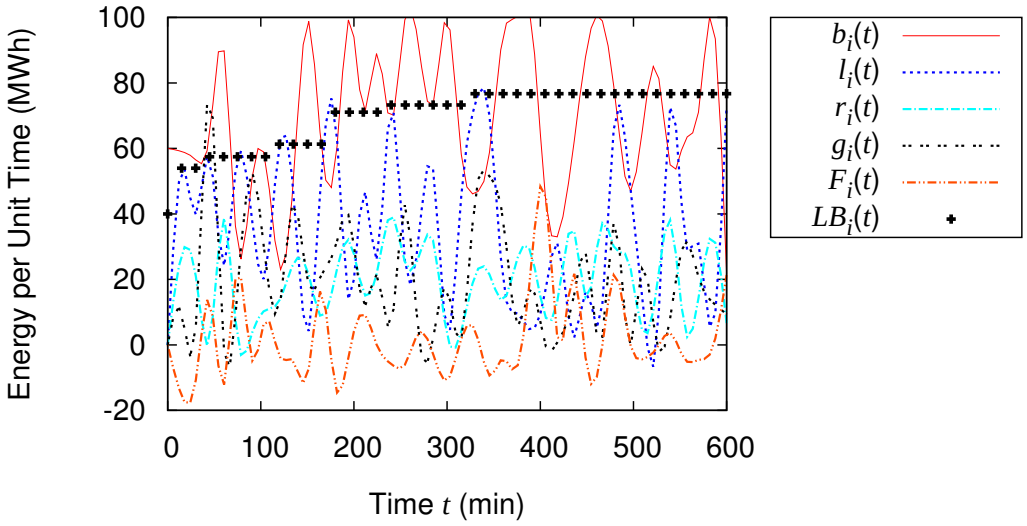



Figure 3.10: Changing process of $b_i(t)$, $l_i(t)$, $r_i(t)$, $g_i(t), F_i(t)$, and $LB_i(t)$ from $t = 0$ to 600 where load process and renewable generation is given with $0 \leq l_i(t) \leq 80$ and $0 \leq r_i(t) \leq 60$, respectively.

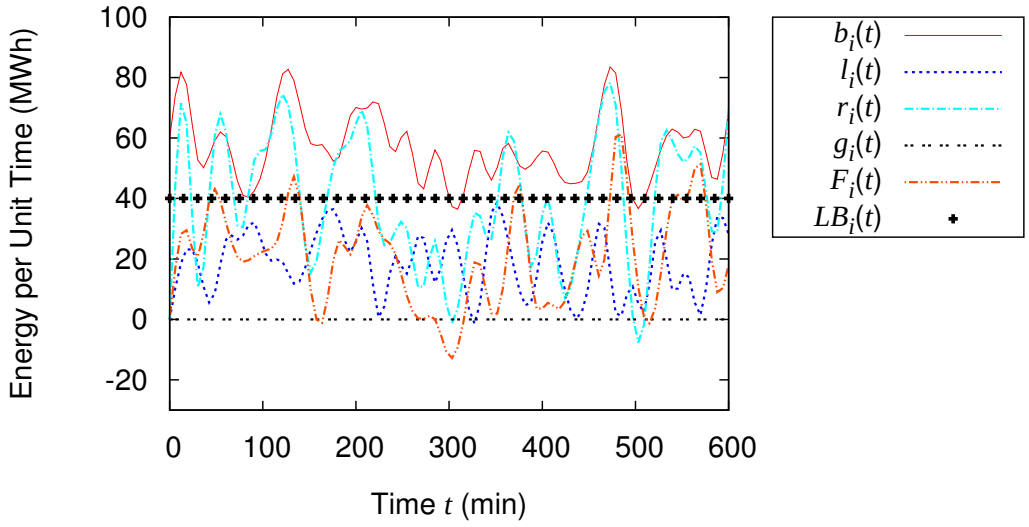



Figure 3.11: Changing process of $b_i(t)$, $l_i(t)$, $r_i(t)$, $g_i(t)$, $F_i(t)$, and $LB_i(t)$ from $t = 0$ to 600 where load process and renewable generation is given with $0 \leq l_i(t) \leq 40$ and $0 \leq r_i(t) \leq 80$, respectively.

Figure 3.9 shows the changing process of $b_i(t)$, $l_i(t)$, $r_i(t)$, $g_i(t)$, and $F_i(t)$ from $t = 0, 15, ..., 600$ (min) at node $v_i$. At this node, the load input and renewable generation are randomly given between 0 to 20 MW and 0 MW to 10 MW, respectively. As the proposed method based on tie-sets is applied to the control of $g_i(t)$ and $F_i(t)$, the value of $b_i(t)$ keeps to be around 40 MWh from $t \approx 100$ min. The value of $LB_i(t)$ keeps to be 40 MWh. $v_i$ also distributes power flows to adjacent nodes when it has an excess amount of node energy. This result substantiates the effectiveness of the proposed autonomous control method in terms of sustainability as local optimizations by tie-sets to which each node belongs make the demand response at the node quite stable.

Figure 3.10 is the power stimulus response at another node $v_i$ where the load input is assigned with the range of 0 MW to 80 MW, and the renewable generation is given between 0 MW to 60 MW. As the there are many sudden peak load processes that makes convergence of node energy at $v_i$ unstable. However, the iteration of ORDER-RT constantly tries to cope with the uncertainty of load processes so that the power at $v_i$ never runs out. The behavior of LEL at this node dynamically changes according to the load peaks, which is also the key

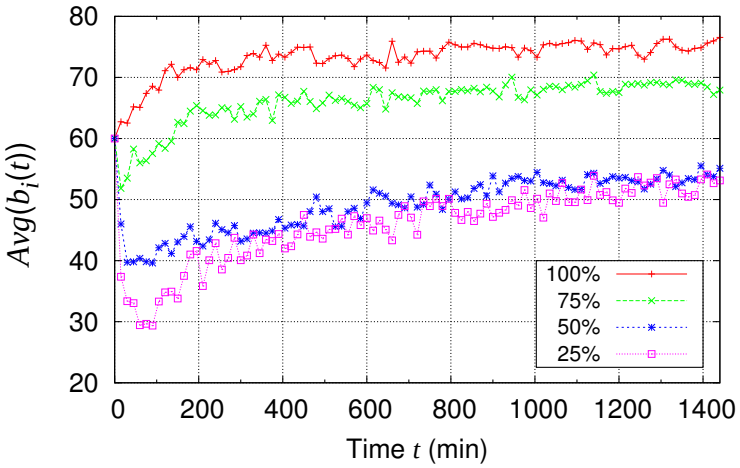



Figure 3.12: Convergence behavior of the average node energy $Avg(b_i(t)) = \overline{B}(t)$ with different Renewable Penetration Rates (RPRs) (100%, 75%, 50%, 25%) in a 100-node power network.

control point to make the node energy stable.

At another node where the maximum load and renewable generation are respectively bounded by 40 MW and 80 MW as shown in Figure 3.11, the energy level is saturated with sudden renewable peaks. As $\Delta B_i(t)$ is preserved for unexpected renewable peaks, this demonstrates the effective use of renewables. When the node cannot store the excess amount of renewable, it distributes the power to the other nodes.

The results as in Figures 3.9 to 3.11 demonstrate the effectiveness of the Tie-set Based Optimization since it is able to cope with unexpected load peaks and renewable generation by dynamically controlling power flows and CGF supplies within the system of fundamental tie-sets. In addition, the results indicate that most of the node energies quickly converge on LELs within a short time, and then they keep close to LELs according to the changes of LELs as $t$ progresses.

### 3.5.2 Convergence with Different Renewable Penetration Rates

We conducted simulation with different RPRs as it is an important issue to be considered when creating future power networks with renewable integration. The RPR can be adjusted by changing the amount of renewable generation. In this simulation, load input is randomly given between 0 MW and 60 MW at each node $v_i$ at each time $t$. When expecting RPRs $\mathbb{E}\{r_i(t)/l_i(t)\}$ to be 100%, 75%, 50%, and 25% as in Figure 3.12, we randomly input renewable generation within the range of 0 MW to 60 MW, 0 MW to 45 MW, 0 MW to 30 MW, and 0 MW to 15 MW at each node $v_i$ at each time $t$, respectively. Initial LELs $LB_i(0)$ is set as $LB_i(0) = 20$ MWh.

Simulation data of $b_i(t)$ of each node is taken at $t = 0, 15, 30, ..., 600$ (min). $Avg(b_i(t)) = \overline{B}(t)$ shown in Figure 3.12 stands for the average node energy at time $t$ as in (3.10). Experiments are conducted 20 times for each case of RPR, and the average is calculated.

Figure 3.12 shows the simulated behavior of the average node energy $\overline{B}$ in a 100-node power network with different RPRs. When RPR is 100%, the average node energy $\overline{B}(t)$ converges around 75 MWh since each node stores the renewable energy in case the amount of renewable generation is larger than the load process. After the average node energy draws to 75 MWh, power flows among nodes become stable by distributing stored DERs to the nodes in short of power with minimum amount power from CGF. The similar behavior can be seen when RPR is 75% where $\overline{B}(t)$ converges around 70 MWh. sIn case of 50% and 25% RPRs, the average energy $\overline{B}(t)$ first drops since the initial LELs are $LB_i(0) = 20$ MWh at all the nodes. The values of $LB_i(t)$ are dynamically updated change according to the peak load profile at each node over the time range. Thus, the average energies gradually converge to 60 MWh with the updated LELs $LB_i(t)$.

We verified that the behavior of convergence in 200-node to 500-node networks with the different RPRs from 25% to 100% is similar to the behavior in the 100-node network. In

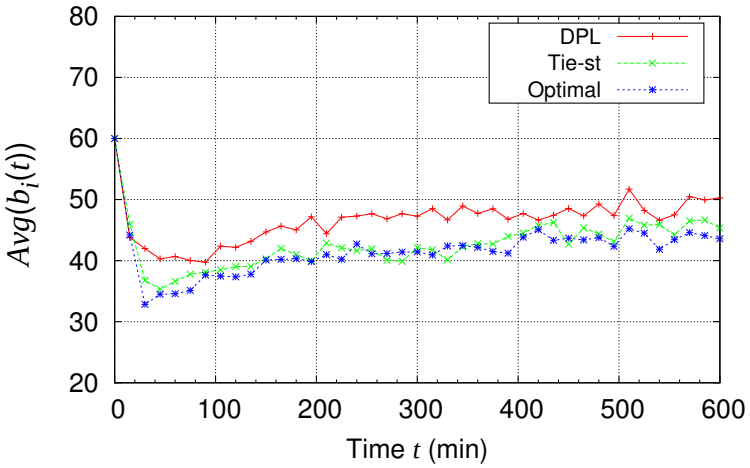



Figure 3.13: Comparison of the average node energy $\overline{B}(t) = Agv(b_i(t))$ with the optimal $\overline{B}^*(t)$ and the Distributed Linear Programing (DLP) when RPR is 50%.

Table 3.2: Comparison of total CGF power provision $\sum_{t=0}^{600} \sum_{v_i \in V} g_i(t)$ (MW) with different RPRs

| RPR | 25% | 50% | 75% | 100% |
|---|---|---|---|---|
| DLP | 86708.70 | 62315.36 | 38134.25 | 28874.94 |
| Tie-set | 85637.09 | 57945.28 | 36460.37 | 25686.16 |
| Optimal | 85007.26 | 57535.15 | 34474.77 | 25302.25 |

addition, we confirmed that the convergence process does not depend on the initial power at each node.

### 3.5.3 Comparison Experiment with Distributed Linear Programing

In this experiment, we assess the optimality of the proposed method and compare it with Distributed Linear Programing (DLP) conducted at each node. The mechanism of DLP for ORDER-RPF problem is described in Appendix 3.5.5 We analyze the following factors:

- Changing process of the average of node energies $Avg(b_i(t)) = \overline{B}(t)$

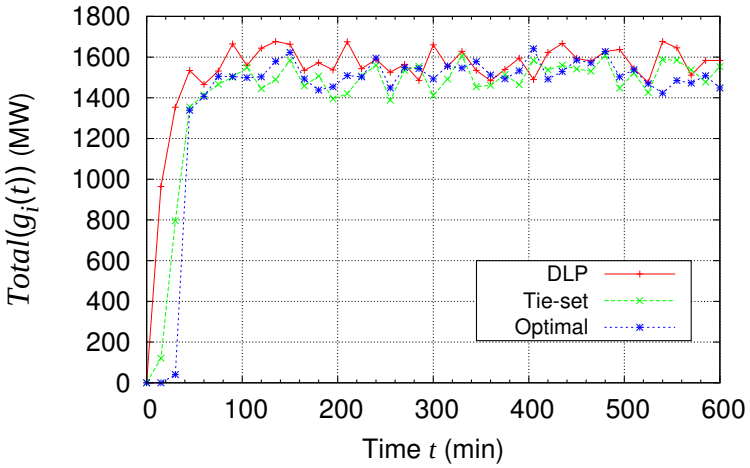



Figure 3.14: Comparison of total CGF power provisions $Total(g_i(t)) = \sum_{v_i \in V} g_i(t)$ with the optimal line of the solution and DLP when RPR is 50%.

- Amount of total power generated at CGF $Total(g_i(t)) = \sum_{v_i \in V} g_i(t)$

In order to assess the optimality of Tie-set Based Optimization (TBO), the theoretical solution of ORDER-RPF is analyzed in Section 3.5.4. Naturally, any optimization technique cannot excel in the solution. Therefore, it is important to analyze how the proposed technique is close to the theoretical limitation of the ORDER-RPF problem in terms of the power supply from CGF. The simulation network is the same as the 100-node network in the previous section. The RPR is intended to be 50% by giving load and Renewable generation with the range of 0 MW to 60 and 0 MW to 30 MW, respectively. Simulation data of $\sum_{v_i \in V} g_i(t)$ is taken at $t = 0, 15, 30, ..., 600$ (min). Experiments are conducted 20 times and the average is calculated.

Figure 3.13 demonstrates the average node energy $\overline{B}(t) = Agv(b_i(t))$ within the time from $t = 0$ to 1440 min (= 24 h) to compare TBO with DLP and the optimal $\overline{B}^*(t)$. The average node energy by TBO is closer to the theoretical solution than DLP since TBO can calculate power flows that balance the node energies within the optimization units defined by tie-sets by using the nature that local optimization conducted within a system of tie-sets leads to the global optimization [20, 27]. DLP requires more power than TBO almost at each

time step since the optimization is based on individual nodes and power is exchanged only with adjacent nodes, which restricts the scale of optimization, even though the process of DLP is completely decentralized by communicating with other peripheral nodes with limited state information. Therefore, CGF supply at each node is not minimized as it must satisfy $b_i(t) \geq LB_i(t)$ to compensate the demand that has not been met by its net power flow $F_i(t)$.

Figure 3.14 shows the changing process of $\sum_{v_i \in V} g_i(t)$ by DLP and TBO (Tie-set) as well as the theoretical solution (Optimal). As indicated in Figure 3.14, the optimal line shows minimum allocations of CGF at each time. The the result by TBO is similar to the theoretical solution as both TBO and Optimal line converge between 1400 MW and 1600 MW after $t = 60$ min. DLP makes up for the demand not sufficiently satisfied by power flows at each node, which leads to excessive power supply from CGF. That is why the simulated behavior of the average node energy by DLP seen in Figure 3.13 is constantly higher than that of TBO and theoretical solution.

Table 3.2 shows comparison of the total CGF power provision from $t = 0$ to 600 at all the nodes in $V$ ($\sum_{t=0}^{600} \sum_{v_i \in V} g_i(t)$) with different RPRs. The result indicates that the total CGF supply by TBO is close to the theoretical solution with modest increase, whereas the amount of power supply by DLP is larger than TBO and the optimal amount. Therefore, we could reduce the cost of using fossil fuels and nuclear plants by fully taking advantage of renewable generation on the basis of the nature of iterative ORDER-RT where the optimization in a system of $\mu$ independent tie-sets quickly leads to global optimization.

### 3.5.4 Solution of Total CGF Power Supply in ORDER-RPF Problem

We analyze the theoretical solution for the total amount of power generated at CGFs $\sum_{v_i \in V} g_i(t)$ at time $t$. Let $X = \{x\}$ and $Y = \{y\}$ be the sets of nodes that satisfy

$b_x^d(t) \geq LB_x(t)$ and $b_y^d(t) < LB_y(t)$, respectively. Then, we define $\Delta b^+$ and $\Delta b^-$ as

$$\Delta b^+ = \sum_{x \in X} (b_x^d(t) - LB_x(t)), \tag{3.28}$$

$$\Delta b^- = \sum_{y \in Y} (LB_y(t) - b_y^d(t)). \tag{3.29}$$

With $\Delta b^+$ and $\Delta b^-$, the solution of $\sum_{v_i \in V} g_i(t)$ at time $t$ in ORDER-RPF problem is as follows:

$$\sum_{v_i \in V} g_i(t) = \begin{cases} 0, & \Delta b^+ \geq \Delta b^- \\ \Delta b^- - \Delta b^+, & \Delta b^+ < \Delta b^- \end{cases} \tag{3.30}$$

If all the node energies are not balanced, there would be excess power supply from CGF to ensure that every $b_i(t)$ has power at least $LB_i(t)$. Therefore, the optimal amount of CGF power (3.30) can be provided if node energies are balanced by optimal power flows, which leads to minimizing the power supply from CGF.

### 3.5.5 Approach based on Distributed Linear Programing

The most common way to solve the ORDER-RPF problem would be exploiting the notion of Linear Programing that has been applied to a lot of linear-optimization problems. As we are solving the ORDER-RPF problem in a distributed manner, a Distributed Linear Programing (DLP) approach is formulated in Algorithm 4 where each node $v_i$ tries to minimize the amount of power generated at its CGF $g_i(t)$ by communicating with adjacent nodes $\mathcal{A}(v_i)$.

Step 0 is an initialization procedure where the angle and CGF are set as 0, and its net node energy is calculated with the net demand. Then, $v_i$ shares $b_i^d(t)$, $LB_i(t)$, and $H_i(t)$ with adjacent nodes $\mathcal{A}(v_i)$.

In Step 1, as each node receives BPC of its adjacent nodes, $v_i$ set $flag_1 = 1$ if the value of BPC $H_i(t)$ is the greater than those of adjacent nodes, otherwise $flag_1 = 0$.

Only if $flag_1 = 1$, Step 2 is conducted at node $v_i$. If none of the node energy at every adjacent node $b_j^d(t)$ is less than its LEL $LB_j(t)$, $flag_2 = 1$, otherwise $flag_2 = 0$ and the procedure from line 18 to 31 is skipped. If $flag_2 = 1$, the angle of the node is calculated to satisfy all the ORDER-RPF conditions. Then, the net power flow and CGF generation is calculated.

After deciding $F_i(t)$ and $g_i(t)$ at each node, in Step 3, $v_i$ notifies its adjacent nodes that optimization has finished. Then, $v_i$ stands by for $\Delta t$ min and iterates the procedure above.

## 3.6 Summary

In this chapter, we introduced an optimal real-time distribution of renewables (ORDER) and described how to solve it in a decentralized manner. There mainly are two parts of this chapter, which are ORDER with independent power flows and ORDER with real power flows. In the ORDER problem with independent power flows, we employ a simplified model where the power flows are flexible enough to consider them as routing commodities. By simplifying the power flows, this model can be integrated into various kinds of power networks even though the flows are wireless power packets.

In another ORDER model with dependent real power flows, we focused on DC power flows using power systems' properties such as voltages, admittances, and phase angles. By considering those properties, we can directly apply the model to transmission networks where they also integrates storage systems in the near future. The nature of ORDER does not change even though the power flows are dependent or independent, and the optimization nature of tie-set graph theory is verified where for each case the globally balanced state of DERs is

obtained by iterative local optimization based on tie-sets.

The uncontrollability of loads and renewables is accounted for by random power output level determined at each time step in the simulation. The simulation results show that balanced distributions of renewables constantly conducted within a fundamental system of tie-sets realize a sustainable power network complimented by optimal power supplies from a centralized generation facility even if the load process and renewable generation is highly variable and unpredictable.

This work deals with essential aspects of prospective power networks so that optimization theories and algorithms being studied in the boundary domain between power systems and distributed network systems can be integrated into the proposed method. With the characteristics of prospective power systems, we are extending our algorithmic solution and simulation environment to incorporate them.

**Algorithm 4** Decentralized Linear Programing for ORDER-RPF

**STEP 0:** Initialzation

$\theta_i(t) = 0$, $g_i(t) = 0$.
$d_i(t) := l_i(t) - r_i(t)$.
$b_i^g(t) := b_i(t-1) - d_i(t)\Delta\tau$.
Exchange $b_i^g(t)$, $LB_i(t)$, and $H_i(t)$ with adjacent nodes $\mathcal{A}(v_i)$.

**STEP 1:** BPC Comparison

Set $flag_1 = 1$.
**for** each $v_j \in \mathcal{A}(v_i)$ **do**
  **if** $H_i(t) < H_j(t)$ **then**
    Set $flag_1 = 0$.
  **end if**
**end for**

**STEP 2:** Deciding Net Power Flow and CGF

**if** $flag_1 = 1$ **then**
  Set $flag_2 = 1$.
  **for** each $v_j \in \mathcal{A}(v_i)$ **do**
    **if** $b_j^d(t) < LB_j(t)$ **then**
      Set $flag_2 = 0$.
    **end if**
  **end for**
  **if** $flag_2 = 1$ **then**
    Set $\theta_{min} := 100$.
    **for** each $v_j \in \mathcal{A}(v_i)$ **do**
      $\theta_{tmp} := -\frac{b_j^d(t) - LB_j(t)}{V_i V_j Y_{ij}}$.
      **if** $|\theta_{tmp}| < |\theta_{min}|$ **then**
        Set $\theta_{min} = \theta_{tmp}$.
      **end if**
    **end for**
    **if** $|F_i(t)| > LB_i(t) - b_i(t)$ **then**
      $\theta_i(t) = -\frac{LB_i(t) - b_i(t)}{\sum_{v_j \in \mathcal{A}(v_i)} V_i V_j Y_{ij}}$.
    **else**
      Set $\theta_i(t) = \theta_{min}$.
      Set $g_i(t) = LB_i(t) - b_i(t)$.
    **end if**
  **else**
    Set $g_i(t) = LB_i(t) - b_i(t)$.
  **end if**
**end if**

**STEP 3:** Stand-By and Iteration

Notify $\mathcal{A}(v_i)$ that optimization has finished.
Stand by for $\Delta t$.
Go to STEP 0.

# Chapter 4

# ORPF: Optimal Real-Time Power Flow

In ORDER problem, we try to minimize the amount of power generated at Centralized Generation Facility (CGF) with the model that integrates renewable generation, energy storage, bi-directional power-flow switches, and real-time end-use devices such as smart meters. An Optimal Real-Time Power Flow (ORPF) with energy storage discussed in this chapter extends the model above to minimize the costs of energy storage and of power generation from fossil fuel that are required to balance the loads and generation from renewable sources across multiple time periods.

## 4.1 Problem Formulation for ORPF

### 4.1.1 Constraints

We again consider a connected graph $G = (V, E)$ with a set of nodes $V = (v_i, i = 1, ..., n)$ representing the nodes/buses and a set of links $E = \{e(v_i, v_j)\} \subseteq V \times V$ representing the power lines. The constraints of the ORPF problem are the same as those of ORDER problem listed as

$$b_j(t) = b_j(t-1) - p_j(t)\Delta\tau, \quad \text{for } v_j \in V, \tag{4.1}$$

where $\Delta\tau$ is an actual time interval between $t-1$ and $t$, and

$$p_j(t) = d_j(t) - g_j(t) + F_j(t), \tag{4.2}$$

$$d_j(t) := l_j(t) - r_j(t), \tag{4.3}$$

$$F_j(t) := \sum_{k:j\to k} f_{jk}(t) - \sum_{i:i\to j} f_{ij}(t), \tag{4.4}$$

$$0 \le b_j(t) \le \overline{b}_j, \quad \text{for } v_j \in V, \tag{4.5}$$

$$0 \le g_j(t) \le \overline{g}_j, \quad \text{for } v_j \in V, \tag{4.6}$$

$$-\overline{f}_{ij} \le f_{ij}(t) \le \overline{f}_{ij}, \quad \text{for } e(v_i, v_j) \in E, \tag{4.7}$$

where $b_j(t)$ is the node energy, $d_j(t)$ is the net demand, $l_j(t)$ is the load, $r_j(t)$ is the renewable generation, $g_j(t)$ is the generation at CGF, $F_j(t)$ is the net power flow, $f_{ij}(t)$ is power flow, $\overline{b}_j$ is the battery capacity, and $\overline{g}_j$ is the capacity of CGF generation.

### 4.1.2 Objective Function

The generated power injections $g_i(t)$ by CGF incur some cost. We denoted the cost of generation $g_i(t)$ at time $t$ by $c_i(g_i(t), t)$. We also consider a cost $h_i(b_i(t))$ for the battery with energy level $b_i(t)$ at time $t$. Then, we formulate the ORPF problem to minimize the cost of CGF generations and efficiently distribute DERs for reliable use of batteries.

*ORPF:* Given initial battery levels $b(0) \geq 0$, net demand $d$, battery capacities $\overline{b}$, CGF capacities $\overline{g}$, link capacities $\overline{f}$, and a given time sequence $\mathcal{T} = \{t\}$, the ORPF is

$$\begin{aligned} \min \quad & \sum_{t \in \mathcal{T}} \sum_{v_i \in V} \big( c_i(g_i(t), t) + h_i(b_i(t)) \big) \\ \text{over} \quad & b, f, F, g, p \\ \text{s.t.} \quad & (4.1), (4.2), (4.3), (4.4), (4.5), (4.6), (4.7). \end{aligned} \tag{4.8}$$

Here, we consider quadratic cost functions for CGFs:

$$c_i(g_i(t), t) := \frac{1}{2} \gamma_i(t) g_i^2(t) \tag{4.9}$$

where $\gamma_i(t)$ is a time-varying coefficient. The battery cost is assumed to be dependent on the battery level $b_i$. As we want to maintain the useful life of a battery, we consider the battery cost function that imposes a penalty proportional to the deviation from its capacity as

$$h_i(b_i(t)) := \alpha_i(\overline{b}_i - b_i(t)) \tag{4.10}$$

for some $\alpha_i > 0$.

## 4.2 Optimal Real-Time Power Flow based on Tie-sets

### 4.2.1 TADiC for ORPF

In the OPPF problem, TEF is defined as the average value of battery cost functions in a tie-set as follows:

$$\Phi(L_\lambda, t) = \frac{\sum_{v_i \in V(L_\lambda)} h_i(b_i(t))}{|V(L_\lambda)|}. \tag{4.11}$$

The tie-set(s) with larger value of $\Phi(L_\lambda, t)$ gains the process priority to conduct Algorithm 5 in the next section or Algorithm 6 in the section 4.4.2.

For instance, in Figure 4.1, let us suppose that $b_a(t) = 40$ MWh, $b_b(t) = 50$ MWh, $b_c(t) = 60$ MWh, and $b_d(t) = 70$ MWh at time $t$. For each node of Figure 4.1, $\alpha_i$ of the battery cost (4.10) is $\alpha_i = 1$ and the capacity is $\overline{b}_i = 100$. Therefore, the battery costs are $h_a(b_a(t)) = 60$, $h_b(b_b(t)) = 50$, $h_c(b_c(t)) = 40$, and $h_d(b_d(t)) = 30$. According to (4.11), the TEF value of $L_1$ and $L_2$ is $\Phi(L_1, t) = 50$ and $\Phi(L_2, t) = 40$, respectively. On the basis of TADiC written in Chapter 2, $L_1$ will conduct the optimization (Algorithm 5 or 6) in that case.

### 4.2.2 Decentralized Algorithm for Optimal Real-Time Power Flow

We optimize the objective function (4.8) in a tie-set $L_\lambda$ at every time step $t$. A node energy $b_i(t)$ has two types of variables; the CGF generation $g_i(t)$ and the net power injection $F_i(t)$. Our algorithmic solution is designed to first optimize the cost (4.8) focusing on CGF generations, and then optimize flows in a tie-set as described in Algorithm 5. Necessary

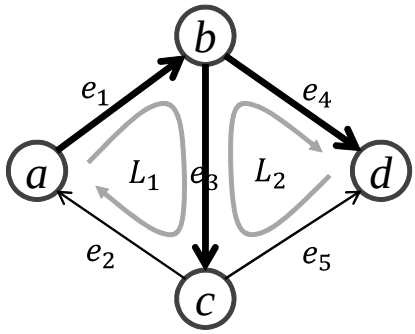



Figure 4.1: Example of tie-sets $L_1$ and $L_2$.

information in a tie-set $L_\lambda$ is provided by MV $y_\lambda(t)$ included in TA and stored in its leader node $v_l^\lambda$. After Initialization, $v_l^\lambda$ conducts the following procedures. For simplicity, $\Delta\tau = 1$ hour.

**CGF Optimization**

In Step 1, the optimum CGF power $g_i^*(t)$ at $v_i \in V(L_\lambda)$ is first calculated without dealing with the net power injection $F_i(t)$. Let us define $b_i(g_i(t), t)$ as

$$b_i(g_i(t), t) := b_i(t-1) - d_i(t) + g_i(t), \quad \text{for } v_i \in V(L_\lambda) \tag{4.12}$$

where $V(L_\lambda)$ is a set of all the nodes in a tie-set $L_\lambda$. Solving for $g_i(t)$, we get: $g_i(t) = b_i(g_i(t), t) - b_i(t-1) + d_i(t)$.

Now we have some constraints on $g_i(t)$ and $b_i(g_i(t), t)$ as in (4.5) and (4.6). On the one hand, since $b_i(g_i(t), t) \le \overline{b}_i$ and $g_i(t) \le \overline{g}_i$, we have

$$g_i(t) \le \min\{\overline{b}_i - b_i(t-1) + d_i(t),\ \overline{g}_i\}. \tag{4.13}$$

On the other hand, we know that $g_i(t) \geq 0$ and $b_i(g_i(t), t) \geq 0$. Therefore, we have

$$g_i(t) \geq \max\ \{d_i(t) - b_i(t-1),\ 0\}. \tag{4.14}$$

Let us define $\psi(g_i(t), t)$ on the basis of (4.8) as

$$\begin{aligned} \psi(g_i(t), t) &= c_i(g_i(t), t) + h_i(b_i(g_i(t), t)) \\ &= \frac{1}{2}\gamma_i(t) g_i^2(t) + \alpha_i \left(\overline{b}_i - b_i(g_i(t), t)\right). \end{aligned} \tag{4.15}$$

The CGF optimization problem is now to find the value of $g_i^*(t)$ that minimizes $\psi(g_i(t), t)$ at each node $v_i \in V(L_\lambda)$ subject to the above constraints (4.13) and (4.14). We take a derivative of $\psi(g_i(t), t)$ at $v_i \in V(L_\lambda)$ at time $t$.

$$\frac{\partial \psi}{\partial g_i(t)}(g_i(t), t) = 0 \tag{4.16}$$

i.e.,

$$g_i(t) = \frac{\alpha_i}{\gamma_i(t)} \tag{4.17}$$

Let us define $g_{min} = \max\ \{d_i(t) - b_i(t-1),\ 0\}$ and $g_{max} = \min\ \{\overline{b}_i - b_i(t-1) + d_i(t),\ \overline{g}_i\}$, and then the optimal CGF $g_i^*(t)$ for each $v_i \in V(L_\lambda)$ at $t$ is:

$$g_i^*(t) = \begin{cases} \frac{\alpha_i}{\gamma_i(t)} & \text{if } g_{min} \leq \frac{\alpha_i}{\gamma_i(t)} \leq g_{max} \\ g_{min} & \text{if } \frac{\alpha_i}{\gamma_i(t)} < g_{min} \\ g_{max} & \text{if } \frac{\alpha_i}{\gamma_i(t)} > g_{max} \end{cases} \tag{4.18}$$

**Flow Optimization**

After deciding CGF variables $g_j(t)$ in $V(L_\lambda)$, the flow values $f$ on $L_\lambda$ are optimized in Step 2 considering the battery cost function with $g_j^*(t)$ decided above. Let $\phi_\lambda$ be

$$\begin{aligned}\phi_\lambda &= \sum_{v_j \in V(L_\lambda)} \alpha_j \left(\overline{b}_j - (b_j(g_j^*(t), t) - F_j(t))\right) \\ &= \sum_{v_j \in V(L_\lambda)} \alpha_j F_j(t) + \sum_{v_j \in V(L_\lambda)} \alpha_j \left(\overline{b}_j - b_j(g_j^*(t), t)\right) \end{aligned} \tag{4.19}$$

According to (4.4), $F_j(t)$ at node $v_j$ in a tie-set $L_\lambda$ is

$$F_j(t) = f_{jk}(t) - f_{ij}(t), \tag{4.20}$$

where $\{e(v_j, v_k), e(v_i, v_j)\} \subset L_\lambda$. On the basis of (4.20), $\sum_{v_j \in V(L_\lambda)} \alpha_j F_j(t)$ of $\phi_\lambda$ in (4.19) is

$$\begin{aligned}\sum_{v_j \in V(L_\lambda)} \alpha_j F_j(t) &= \alpha_1(f_{12}(t) - f_{\Gamma 1}(t)) + \alpha_2(f_{23}(t) - f_{12}(t)) \\ &\quad + \cdots + \alpha_\Gamma(f_{\Gamma 1}(t) - f_{\Gamma,\Gamma-1}(t)) \\ &= (\alpha_1 - \alpha_2) f_{12}(t) + (\alpha_2 - \alpha_3) f_{23}(t) \\ &\quad + \cdots + (\alpha_\Gamma - \alpha_1) f_{\Gamma 1}(t) \end{aligned} \tag{4.21}$$

where $V(L_\lambda) = \{v_1, v_2, ..., v_\Gamma\}$ and $L_\lambda = \{e(v_1, v_2), e(v_2, v_3), ..., e(v_\Gamma, v_1)\}$.

Therefore, $\phi_\lambda$ in (4.19) is transformed into a function of a *tie-set flow vector* $\mathbf{f}$:

$$\phi_\lambda(\mathbf{f}) = A^{\mathrm{T}}\mathbf{f} + B^{\mathrm{T}}(C - D) \tag{4.22}$$

where $\mathbf{f} = \begin{bmatrix} f_{12}(t) \\ f_{23}(t) \\ \vdots \\ f_{\Gamma 1}(t) \end{bmatrix}$, $A = \begin{bmatrix} \alpha_1 - \alpha_2 \\ \alpha_2 - \alpha_3 \\ \vdots \\ \alpha_\Gamma - \alpha_1 \end{bmatrix}$, $B = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_\Gamma \end{bmatrix}$, $C = \begin{bmatrix} \overline{b}_1 \\ \overline{b}_2 \\ \vdots \\ \overline{b}_\Gamma \end{bmatrix}$, and $D = \begin{bmatrix} b_1(g_1^*(t), t) \\ b_2(g_2^*(t), t) \\ \vdots \\ b_\Gamma(g_\Gamma^*(t), t) \end{bmatrix}$.

Each element of the tie-set flow vector satisfies (4.7). In addition, $b_j(t)$ satisfies (4.5). Therefore, we solve the following problem:

$$\min \quad A^{\mathrm{T}}\mathbf{f} + B^{\mathrm{T}}(C - D) \tag{4.23}$$

$$\text{s.t.} \quad D - C \leq \mathbf{f}' \leq D \tag{4.24}$$

$$-\overline{f} \leq \mathbf{f} \leq \overline{f} \tag{4.25}$$

where $\mathbf{f}' = \begin{bmatrix} f_{12}(t) - f_{\Gamma 1}(t) \\ f_{23}(t) - f_{12}(t) \\ \vdots \\ f_{\Gamma 1}(t) - f_{\Gamma-1,\Gamma}(t) \end{bmatrix}$ and $\overline{f} = \begin{bmatrix} \overline{f}_{12} \\ \overline{f}_{23} \\ \vdots \\ \overline{f}_{\Gamma 1} \end{bmatrix}$.

As the edges in a tie-set form a linear structure, we can solve (4.23) by linear programing.

## 4.3 Simulation and Analysis for ORPF

Simulation and experiments were conducted to verify the proposed method for solving the ORPF problem and to analyze the solution and its behavior. The results described below confirm that at every node the useful battery life is maintained by the procedure described for calculating CGF generation and in-and-out power flows. As the network becomes larger, the stability of behavior of overall battery levels does not change, which demonstrates the

**Algorithm 5** Decentralized Algorithm for ORPF

**STEP 0:** Initialization
  **for** each $v_i \in V(L_\lambda)$ **do**
    $b_i(t-1)$ is preserved in $v_l^\lambda$ of $L_\lambda$ at $t$.
    $d_i(t) := l_i(t) - r_i(t)$ is provided by TA.
    Set $g_i(t) = 0$.
  **end for**
  For each $e(v_i, v_j) \in L_\lambda$, set $f_{ij}(t) = 0$.
**STEP 1:** CGF Optimization
  **for** each $v_i \in V(L_\lambda)$ **do**
    Calculate $g_i^*(t)$ according to (4.18).
  **end for**
**STEP 2:** Flow Optimization
  Calculate $\mathbf{f}$ according to (4.23) - (4.25).
  Distribute flows in $L_\lambda$.

effectiveness of the proposed method in terms of scalability.

In each run of our simulation, the network is configured to be a biconnected network in which each node has at least 2 link connections. Although links are directed, power flow can pass bidirectionally. Each node has a function that produces its load and renewable power generation over time. In addition, each node $v_i$ has a battery whose capacity is set as $\overline{b}_i = 20$ MWh (Megawatt). The generation capacity is $\overline{g}_i = 100$ MW, and the link capacity is $\overline{f}_{ij} = 100$ MW. The initial energies are uniformly set as $b_i(0) = 5$. $\alpha_i$ of the storage cost $h_i(b_i(t)) = \alpha_i(\overline{b}_i - b_i(t))$ is assigned between 1 to 2 at random; $\gamma_i(t) \equiv 1$. The loads $l_i(t)$ and renewable injections $r_i(t)$ are randomly generated with a time interval of 15 min, where $\mathcal{T} = \{0, 15, ...\}$. The communications interval $\Delta t$ in TADiC is set as $\Delta t = 1$ min. MV $y_\lambda(t)$ is constantly reported to the leader of each tie-set by TA.

### 4.3.1 Power Injections at Node

We first analyze the simulated behavior of various power injections at a node. The simulation network $G = (V, E)$ has 100 nodes and 200 links. Since $G$ has 200 links, the number of tie-

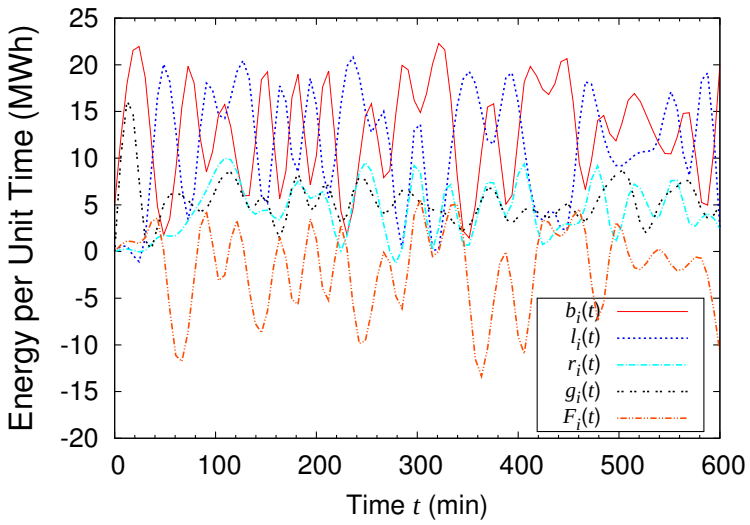



Figure 4.2: The behavior of $b_i(t)$, $l_i(t)$, $r_i(t)$, $g_i(t)$, and $F_i(t)$ at a typical node.

sets is $\mu(G) = |E| - |V| + 1 = 101$. The height of the tree is 7. The loads and renewable injections are bounded by $0 \leq l_i(t) \leq 20$ (MW) and $0 \leq r_i(t) \leq 10$ (MW), respectively, so that the expected RPR (i.e., the expected value of $r_i(t)/l_i(t)$) is intended to be 50%. We show the results at two nodes: an arbitrary node and the root node of the spanning tree in $G$ used to generate the tie-sets.

Figure 4.2 shows the behavior of battery level $b_i(t)$, load $l_i(t)$, renewable generation $r_i(t)$, CGF power generation $g_i(t)$, and net power injection $F_i(t)$ at a typical node $v_i \in V$. The values are shown at 15 minute intervals from $t = 0$ to 600. As shown in Figure 4.2, the battery level $b_i(t)$ varies in the range $0 < b_{10}(t) \leq 20$ (MWh). Shortly after the sudden load peaks occur, node $v_i$ makes up for those peaks with increased CGF generation and increased power inflows. At node $v_i$, the net power injection $F_i(t)$ can be either positive or negative depending on the power levels and the value of $\alpha_j$ at other nodes $V(L_\lambda)$ within a tie-set. The value of $\alpha_i$ is 1.601.

Figure 4.3 shows the behavior at the root node. At this node, $\alpha_i = 1.120$. Since this node is the root of the tree, it belongs to many tie-sets, so it is involved in many optimization procedures conducted in those tie-sets. As can be seen in of Figure 4.3, the root node

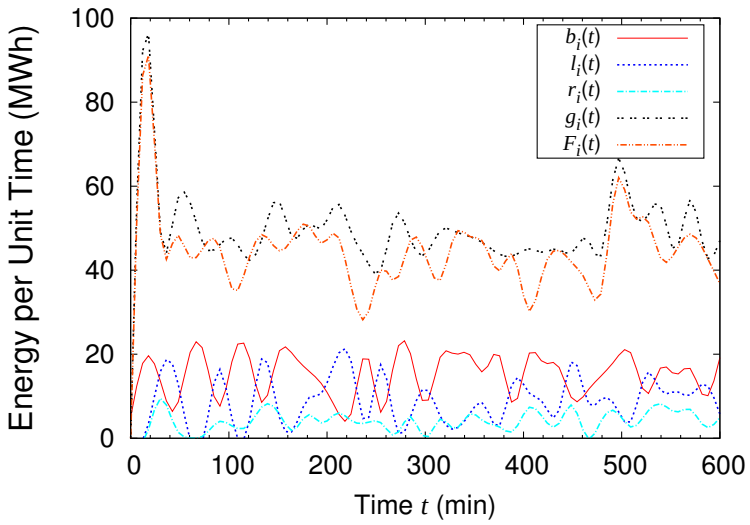



Figure 4.3: The behavior of $b_i(t)$, $l_i(t)$, $r_i(t)$, $g_i(t)$, and $F_i(t)$ at a root node.

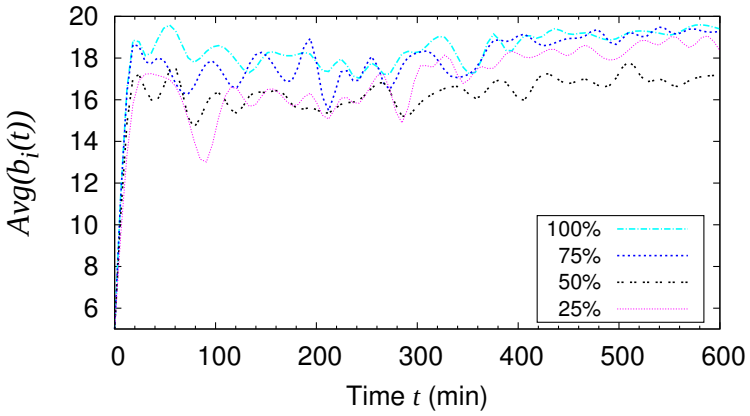



Figure 4.4: The convergence behavior of the average node energy $Avg(b_i(t)) = \overline{B}(t)$ with different renewable penetrations (25% to 100%) in a 100-node grid.

produces more CGF power than the node in Figure 4.2, with the most of the produced CGF power distributed to other nodes. This suggests that the nodes in a grid can be divided into two categories: nodes that obtain power from other nodes, or the nodes that distribute power to other nodes. This category to which a node $v_i$ belongs depends in part on on the number of tie-sets to which it belongs and the value of $\alpha_i$.

### 4.3.2 Convergence of the Overall Power Levels

We analyze the convergence behavior of the overall power levels, taking their average value to see if the proposed scheme is capable of coping with different values of the renewable penetration rate. We calculate $\overline{B}(t)$, the average value of the power levels at time $t$, as follows:

$$\overline{B}(t) = \frac{\sum_{v_i \in V} b_i(t)}{|V|}. \tag{4.26}$$

Convergence can be measured by looking at the changing values of $\overline{B}(t)$ over time. Convergence is achieved from time $t_s$ to $t_e$ provided $0.9 \leq \overline{B}(t+1)/\overline{B}(t) \leq 1.1$ is satisfied during the time interval $t_s \leq t \leq t_e$. We define the maximum load $\bar{l}_i$ and the maximum renewable generation $\bar{r}_i$ by

$$0 \leq l_i(t) \leq \bar{l}_i, 0 \leq r_i(t) \leq \bar{r}_i. \tag{4.27}$$

The renewable penetration rate can be adjusted by fixing the load and changing the maximum renewable generation as follows:

- 100%: $\bar{l}_i = 20$ MW, $\bar{r}_i = 20$ MW
- 75%: $\bar{l}_i = 20$ MW, $\bar{r}_i = 15$ MW
- 50%: $\bar{l}_i = 20$ MW, $\bar{r}_i = 10$ MW
- 25%: $\bar{l}_i = 20$ MW, $\bar{r}_i = 5$ MW

Experimental data was obtained for each value of the renewable penetration rate by running the simulation 10 times and averaging the results.

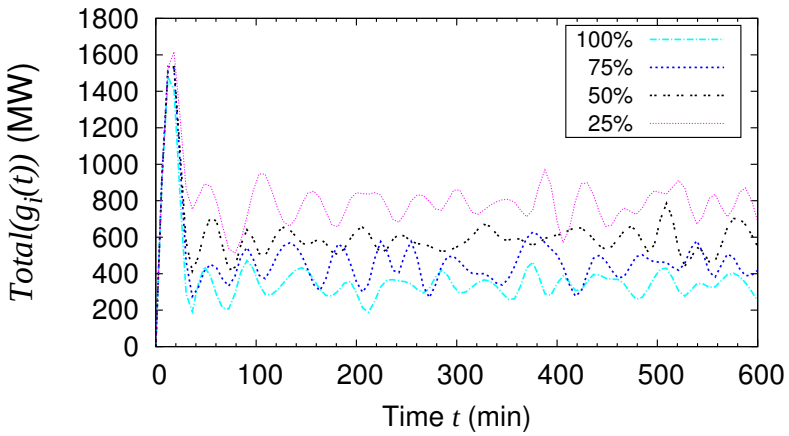



Figure 4.5: The behavior of the total CGF power $Total(g_i(t)) = \sum_{v_i \in V} g_i(t)$ with different renewable penetrations (25% to 100%) in a 100-node grid.

Figure 4.4 shows the result of the simulations in a 100-node grid. From the figure, we see that convergence occurs for all values of the renewable penetration rate, and that as the renewable penetration rate increases the convergence line moves closer to the capacity $\overline{b}_i$. For all renewable penetration rates, after 100 minutes the average battery levels are converging in the range between 15 and 20, and they remain in this range once they have reached it.

### 4.3.3 Analysis of CGF Power Supply

We analyze the behavior of the total CGFs $\sum_{v_i \in V} g_i(t)$ in a 100-node grid from over a 600-minute period with the same simulation conditions as previously. Figure 4.5 shows the result of these simulations.

There is an initial spike in the CGF power injection, as power is injected to match the load. After about 30 minutes, the total CGF power becomes stable as $\overline{B}(t)$ starts to converge. As the renewable penetration rate is lowered, the amount of CGF power injection increases.

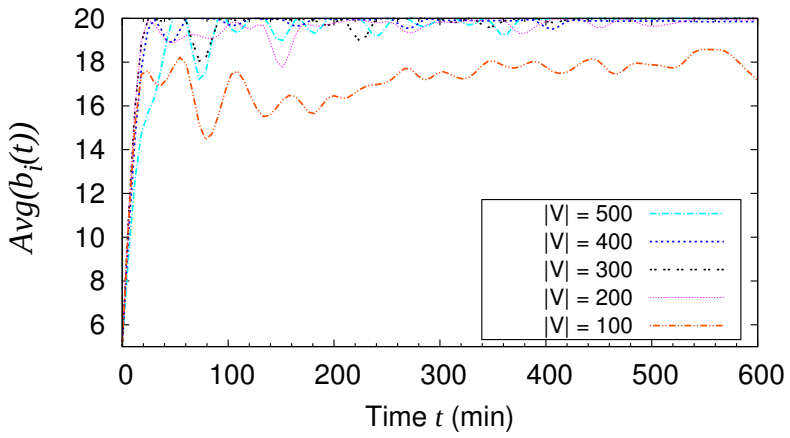



Figure 4.6: The convergence behavior of the average node energy $Avg(b_i(t)) = \overline{B}(t)$ with the number of nodes varying from 100 to 500. The renewable penetration is fixed at 50%.

### 4.3.4 Scalability

To assess scalability, we conducted simulation on networks where the number of nodes varied from 100 to 500 in increments of 100. In these simulations, we held the renewable penetration rate constant at 50%. The results are shown in Figure 4.6. The convergence results are similar to those seen in Figure 4.4 above, as in all cases convergence occurred after no more than 100 minutes. As can be seen in Figure 4.6, as the number of nodes becomes large the final average battery levels approach battery capacity. Although the reason is not clear, this could be because the size of a tie-set (the number of nodes in a tie-set) becomes large and many nodes conduct optimizations in parallel. This behavior suggests that the proposed method will be well suited to future large-scale power networks.

## 4.4 Extended Model for ORPF (ORPF-E)

### 4.4.1 Problem Reformulation

The model we exploited so far assumes that each node has a storage system, the process of load, generation using renewable energies, a CGF generation function, and the power flow export or injection. In this section, we extend the model by classifying the nodes into two types; the generation nodes $\mathcal{G} \subset V$ and the demand nodes $\mathcal{D} \subset V$.

According to (4.1), at the generation node $v_i \in \mathcal{G}$, we have

$$b_i(t) = b_i(t-1) + (r_i(t) + g_i(t) - F_i(t))\Delta\tau, \quad \text{for } v_i \in \mathcal{G}, \tag{4.28}$$

since there is no load process. On the other hand, at the demand node $v_j \in \mathcal{D}$, we define

$$l_j(t) = -F_j(t), \quad \text{for } v_j \in \mathcal{D}, \tag{4.29}$$

as the demand node does not have a storage device, renewable generation function, and CGF generation function.

Then, the extended ORPF (ORPF-E) problem is as follows:

*ORPF-E:* Given initial battery levels $b(0) \geq 0$, the load processes $l$, reneable generation $r$, battery capacities $\overline{b}$, CGF capacities $\overline{g}$, link capacities $\overline{f}$, and a given time sequence $\mathcal{T} = \{t\}$,

the ORPF-E is

$$\begin{aligned} \min \quad & \sum_{t\in\mathcal{T}}\sum_{v_i\in\mathcal{G}}\big(c_i(g_i(t),t)+h_i(b_i(t))\big) \\ \text{over} \quad & b, f, F, g, p \\ \text{s.t.} \quad & (4.4), (4.5), (4.6), (4.7), (4.28), (4.29). \end{aligned} \tag{4.30}$$

### 4.4.2 Decentralized Algorithm for ORPF-E

Decentralized algorithm for ORPF-E is called in the procedure of TADiC and conducted asynchronously by the leader node $v_l^\lambda$ of a tie-set $L_\lambda$. We have the set of generation nodes $\mathcal{G}(L_\lambda)$ and the set of demand nodes $\mathcal{D}(L_\lambda)$ within the tie-set $L_\lambda$, where $\mathcal{G}(L_\lambda)+\mathcal{D}(L_\lambda) = V(L_\lambda)$. Algorithm 6 describes the algorithmic solution for ORPF-E.

#### Initialization

STEP 0 is the process of initialization.

The information of the previous battery power level $b_i(t-1)$ and $r_i(t)$ at each generation node $v_i \in \mathcal{G}(L_\lambda)$ is reported to the leader node of the tie-set $L_\lambda$ by using TA. Let $w_i = \{0,1\}$ be a binary function at a generation node $v_i \in \mathcal{G}(L_\lambda)$ where $w_i = 1$ if node $v_i$ runs out of power, that is, $b_i(t) = 0$, otherwise $w_i = 0$. The CGF generation $g_i(t)$ and the binary function $w_i$ are initialized as 0 at the generation node.

At demand node $v_j \in \mathcal{D}(L_\lambda)$, the information of the load $l_j(t)$ is reported to its leader node using TA. The binary function $w_j$ of a demand node $v_j \in \mathcal{D}(L_\lambda)$ is defined that if the demand $l_j(t)$ is met according to (4.29), $w_j = 1$, otherwise $w_j = 0$. It is also initialized as 0.

The power flows in the tie-set $L_\lambda$ are set as 0.

**CGF Optimization**

CGF optimization is conducted in STEP 1. Let $b_i(g_i(t), t)$ at node $v_i \in \mathcal{G}(L_\lambda)$ be

$$b_i(g_i(t), t) := b_i(t-1) + (r_i(t) + g_i(t))\Delta\tau, \quad \text{for } v_i \in \mathcal{G}(L_\lambda). \tag{4.31}$$

$\Delta\tau = 1$ hour here. According to the constraints (4.5) and (4.6), we have

$$g_i(t) \leq \min\ \{\overline{b}_i - b_i(t-1) - r_i(t),\ \overline{g}_i\}, \tag{4.32}$$

where $0 \leq r_i(t) \leq \overline{b}_i - b_i(t-1)$. Suppose $g_{max} = \min\ \{\overline{b}_i - b_i(t-1) - r_i(t),\ \overline{g}_i\}$. On the basis of the discussion in Section 4.2.2, the optimal CGF generation $g_i^*(t)$ for each node $v_i \in \mathcal{G}(L_\lambda)$ at time $t$ is

$$g_i^*(t) = \begin{cases} \frac{\alpha_i}{\gamma_i(t)} & \text{if } 0 \leq \frac{\alpha_i}{\gamma_i(t)} \leq g_{max}, \\ g_{max} & \text{if } \frac{\alpha_i}{\gamma_i(t)} > g_{max}. \end{cases} \tag{4.33}$$

**Flow Optimization**

Flow Optimization has two phases; one is satisfying the demand $l_j(t)$ at each demand node $v_j \in \mathcal{D}(L_\lambda)$ (STEP2) and another is minimizing the battery cost function $h_i(b_i(t))$ at each generation node $v_i \in \mathcal{G}(L_\lambda)$ (STEP 3) by deciding the power flow on each link of the tie-set $L_\lambda$.

In the Algorithm 6, let $P(i,j)$ and $f_{P(i,j)}(t)$ be a path and its path flow from node $v_i$ to $v_j$

at time $t$, respectively. Then, a path flow $f_{P(i,j)}(t)$ is defined as follows:

$$f_{P(i,j)}(t) = \{f_{i1}(t), f_{12}(t), ..., f_{j-1,j}(t)\} \tag{4.34}$$

If the value of $f_{P(i,j)}(t)$ is decided as $F$, each edge flow of $f_{P(i,j)}(t)$ is updated as follows:

$$f_{P(i,j)}(t) = F \Longrightarrow \begin{cases} f_{i1}(t) \leftarrow f_{i1}(t) + F \\ f_{12}(t) \leftarrow f_{12}(t) + F \\ \qquad\qquad \vdots \\ f_{j-1,j}(t) \leftarrow f_{j-1,j}(t) + F \end{cases} \tag{4.35}$$

In STEP 2, the $flag$ is used to check if every demand $l_j(t)$ at $v_j \in \mathcal{D}(L_\lambda)$ is satisfied or not by using the binary function $w_j$ defined above. After all the demands at $v_j \in \mathcal{D}(L_\lambda)$ are satisfied, $flag$ becomes 1 and it goes on to the next step.

In STEP 3, the optimal flows that minimize the battery cost function $h_i(b_i(t))$ at all the generation nodes $\mathcal{G}(L_\lambda)$ are calculated. We define the set of generation nodes as $\mathcal{G}(L_\lambda) = \{v_1, v_2, v_3, ..., v_\Xi\}$ and the set of paths among generation nodes on the tie-set as $L_\lambda^P = \{P(1,2), P(2,3), ..., P(\Xi, 1)\}$.

Then, we solve the following problem:

$$\min \quad A_b^{\mathrm{T}} \mathbf{f}^P + B_b^{\mathrm{T}}(C_P - D_P) \tag{4.36}$$

$$\text{s.t.} \quad D_P - C_P \leq \mathbf{f}^{P'} \leq D_P \tag{4.37}$$

$$-\overline{f}^P \leq \mathbf{f}^P \leq \overline{f}^P \tag{4.38}$$

where $\mathbf{f}^P = \begin{bmatrix} f_{P(1,2)}(t) \\ f_{P(2,3)}(t) \\ \vdots \\ f_{P(\Xi,1)}(t) \end{bmatrix}$, $A_P = \begin{bmatrix} \alpha_1 - \alpha_2 \\ \alpha_2 - \alpha_3 \\ \vdots \\ \alpha_\Xi - \alpha_1 \end{bmatrix}$, $B_P = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_\Xi \end{bmatrix}$, $C_P = \begin{bmatrix} \overline{b}_1 \\ \overline{b}_2 \\ \vdots \\ \overline{b}_\Xi \end{bmatrix}$, $D_P = \begin{bmatrix} b_1(g_1^*(t),t) \\ b_2(g_2^*(t),t) \\ \vdots \\ b_\Xi(g_\Xi^*(t),t) \end{bmatrix}$,

$\mathbf{f}^{P'} = \begin{bmatrix} f_{P(1,2)}(t) - f_{P(\Xi,1)}(t) \\ f_{P(2,3)}(t) - f_{P(1,2)}(t) \\ \vdots \\ f_{P(\Xi,1)}(t) - f_{P(\Xi-1,\Xi)}(t) \end{bmatrix}$, and $\overline{f}^P = \begin{bmatrix} \overline{f}_{P(1,2)} \\ \overline{f}_{P(2,3)} \\ \vdots \\ \overline{f}_{P(\Xi,1)} \end{bmatrix}$. Note that $\overline{f}_{P(i,j)} = \min\{\overline{f}_{i1}, \overline{f}_{12}, ..., \overline{f}_{j-1,j}\}$.

## 4.5 Simulation and Analysis for ORPF-E

The simulation conditions are as follows:

- The simulation network $G = (V, E)$ has 100 nodes and 200 links.
- The network is biconnected where each node has at least 2 link connections.
- The number of generation nodes $\mathcal{G}$ is 50 (50% of all the nodes).
- The number of demand nodes $\mathcal{D}$ is 50.
- Each generation node has energy storage, the energy harvesting device for renewable energies, and CGF generation function.
- The battery capacity is set as $\overline{b}_i = 20$ MWh.
- The generation capacity is $\overline{g}_i = 100$ MW.
- The link capacity is $\overline{f}_{ij} = 100$ MW.
- The initial battery energies are set as $b_i(0) = 5$.
- $\alpha_i$ is randomly assigned between 1 and 2; $\gamma_i(t) \equiv 1$.

**Algorithm 6** Decentralized Algorithm for ORPF-E

**STEP 0:** Initialization
  **for** each $v_i \in \mathcal{G}(L_\lambda)$ **do**
    $b_i(t-1)$ and $r_i(t)$ is reported to $v_l^\lambda$ of $L_\lambda$ at time $t$ using TA.
    Set $g_i(t) = 0$ and $w_i = 0$.
  **end for**
  **for** each $v_j \in \mathcal{D}(L_\lambda)$ **do**
    $l_j(t)$ is reported to $v_l^\lambda$ of $L_\lambda$ at time $t$ using TA.
    Set $w_j = 0$.
  **end for**
  **for** each each $e(v_i, v_j) \in L_\lambda$ **do**
    Set $f_{ij}(t) = 0$.
  **end for**
**STEP 1:** CGF Optimization
  **for** each $v_i \in \mathcal{G}(L_\lambda)$ **do**
    Calculate $g_i^*(t)$ according to (4.33).
  **end for**
**STEP 2:** Flow Optimization 1
  Set $flag = 0$.
  **while** $flag = 0$ **do**
    Select $v_i \in \mathcal{G}(L_\lambda)$ whose $w_i = 0$.
    Select $v_j \in \mathcal{D}(L_\lambda)$ whose $w_j = 0$.
    **if** $l_j(t) > b_i(t)$ **then**
      $f_{P(i,j)}(t) = b_i(t)$.
      $w_i = 1$.
    **else if** $l_j(t) < b_i(t)$ **then**
      $f_{P(i,j)}(t) = l_j(t)$.
      $w_j = 1$.
    **else**
      $f_{P(i,j)}(t) = l_j(t)$.
      $w_i = 1$, $w_j = 1$.
    **end if**
    For all $v_j \in \mathcal{D}(L_\lambda)$, if $w_j = 1$, set $flag = 1$.
  **end while**
**STEP 3:** Flow Optimization 2
  Calculate $\mathbf{f}^P$ according to (4.36) - (4.38).
  Distribute flows in $L_\lambda$.

- The loads $l_j(t)$ and renewable injections $r_i(t)$ are randomly generated with a time interval of 15 min.
- The communications interval $\Delta t$ in TADiC is set as $\Delta t = 1$ min.
- MV $y_\lambda(t)$ is constantly sent to the leader of each tie-set by TA.

### 4.5.1 Power Stimulus Response at Generation Node

The simulated behavior of battery level, renewable injection, CGF generation, and net power flow at a typical generation node is analyzed here. The process of loads and renewables is bounded by $0 \leq l_j(t) \leq 20$ (MW) and $0 \leq r_i(t) \leq 10$ (MW), respectively, so that the expected renewable penetration rate is meant to be 50%.

Figure 4.7 shows the behavior of $b_i(t)$, $r_i(t)$, $g_i(t)$, and $F_i(t)$ at a typical generation node $v_i \in V$ where the values are shown at 15 minute intervals from $t = 0$ to 600. As indicated in Figure 4.7, the battery energy level $b_i(t)$ also demonstrates the useful life of storage in the range $0 < b_i(t) \leq 20$ (MWh). Generally, the amount of renewable and CGF generation $r_i(t) + g_i(t)$ is almost equivalent to the amount of $F_i(t)$ since the optimization procedure constantly aims at satisfying loads at demand nodes by sustaining the battery life. In addition, $F_i(t)$ is always positive (discharging) as node $v_i$ is a generation node.

### 4.5.2 Load Satisfaction at Demand Node

We show the behavior of load $l_j(t)$ and net power injection $F_j(t)$ at a typical demand node in this section. As in Figure 4.8, the load process exactly matches the net power injection at all times where $l_j(t) = -F_j(t)$ is always satisfied. $F_j(t)$ is always negative because the demand node $v_j$ obtains power from generation nodes. At every demand node, the load

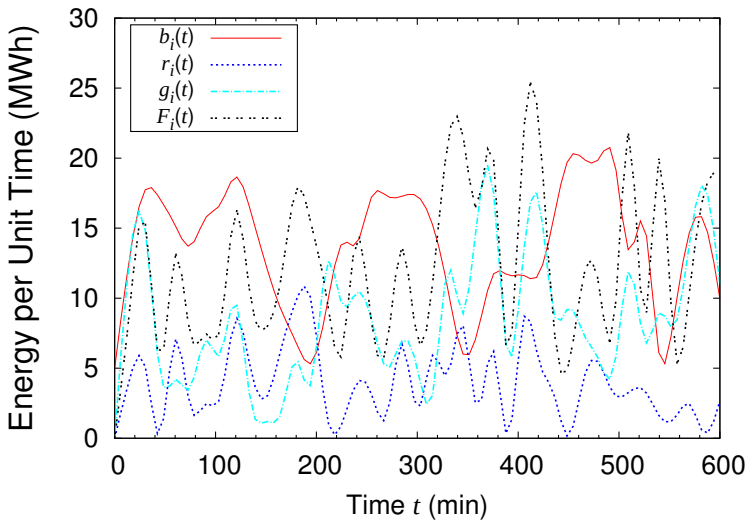



Figure 4.7: The behavior of $b_i(t)$, $r_i(t)$, $g_i(t)$, and $F_i(t)$ at a generation node from $t = 0$ to 600 min with 50% of the renewable penetration rate in a 100-node grid.

process is met at all times with optimized power flows decided in STEP 2 in Algorithm 6 in each tie-set.

### 4.5.3 Average Node Energy with Different Renewable Penetrations

We also analyze the overall behavior of battery power levels by looking at their average value. In Figure 4.9, we calculate $\overline{B}(t)$ according to (4.26) and the experimental results are obtained for each renewable penetration rate (25% to 100%) by running the simulation 10 times.

From the result in Figure 4.9, we figure out that the average energy level $Avg(b_i(t))$ keeps to be between 5 MWh to 14 MWh. As the renewable penetration rate increases, the result behavior moves closer to the capacity $\overline{b}_i$ in a modest manner. This result also substantiates the reliable and sustainable use of battery as they do not deplete power even though the process of load and renewable is intermittent injected by randomizing the process. The

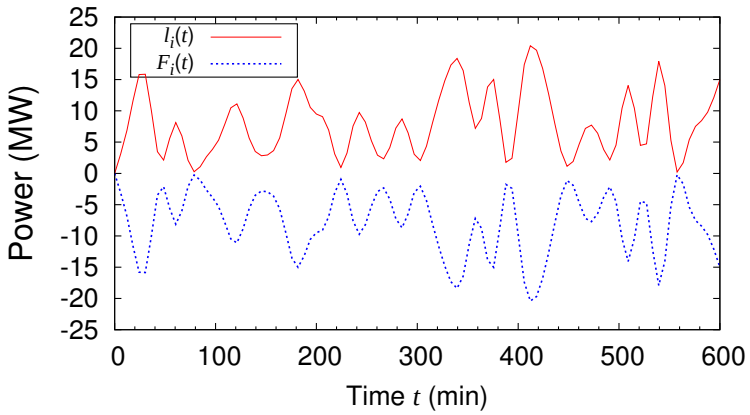



Figure 4.8: The behavior of $l_i(t)$ and $F_i(t)$ at a demand node from $t = 0$ to 600 min with 50% of the renewable penetration rate in a 100-node grid.

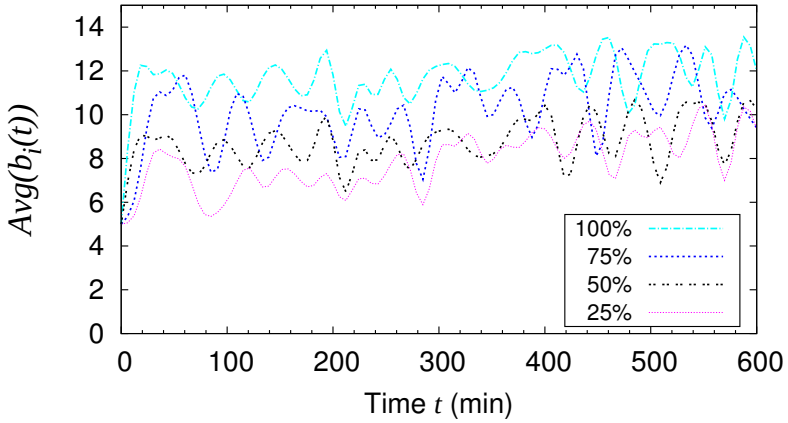



Figure 4.9: The behavior of the average node energy $Avg(b_i(t)) = \overline{B}(t)$ with different renewable penetrations (25% to 100%) in a 100-node grid.

results in Figure 4.9 are more fluctuating than those of Figure 4.4 because generation nodes should always satisfy the demands exactly. Therefore, the more the storage systems are integrated, the less fluctuating the process of battery level becomes as batteries can relax the constraints of the ORPF problems.

## 4.6 Summary

In this Chapter, we have formulated an optimal real-time power flow problem to integrate renewable energy generation and energy storage systems. In the first part of this chapter, we assume all the nodes have energy storage, the process of load, renewable generation, CGF generation, and net power flow function. In the other part of the chapter, we classify nodes into two types; generation nodes that have energy storage, renewable generation, CGF generation, and net power flow and demand nodes with loads that should be satisfied by net power injections.

We have proposed a decentralized algorithm to solve the problem, using a fundamental system of tie-sets that divides a grid into a set of independent loops. Our algorithm minimizes the costs of fossil fuel generation and battery storage within each tie-set.

Simulation results in both two cases above suggest that reliable operation of batteries may be possible with optimal power productions and flow controls based on the system of fundamental tie-sets. The experiments in the 100-node grid, in which 50% of nodes have energy storage and renewable generation, substantiate that demands are exactly satisfied at all the times with instant power dispatches from the generation nodes. The proposed method shows promise as a cost-effective and sustainable grid management strategy that compensates for fluctuations in load and renewable generation in a complex large-scale future power grid.

We are investigating the testbeds that are under development around the world as well as

the Irvine Smart Grid Demonstrating Project. Our future work will employ more efficient topologies and smart grid environment to conduct realistic simulations.

# Chapter 5

# PLM: Power Flow Loss Minimization Problem

A Power Loss Minimization (PLM) problem aims at minimizing the total power flow loss of an entire grid when distributing renewable energy resources in a decentralized manner. Though it is hard to give universal values for power flow loss (path loss) since many factors influence it (overhead or underground cables, type of cables, loading, weather, etc.), we focus on the nature of power loss that is proportionate to the product of the square of current and resistance on a link (the distance and thickness). As the power loss at a link has a quadratic flow cost, the problem can be formulated as a convex optimization problem.

## 5.1 Problem Formulation for PLM

We consider a power network in which the line losses and loads are all resistive. Consequently, all voltages and currents are in phase, and are represented by their root-mean-square values only. The network and its operating conditions are specified by a group $S = (V, E, I, C, R)$,

where $V = (v_i, i = 1, ..., n)$ is the set of vertices, $E = (e_k, k = 1, ..., m)$ is the set of links (lines), $I = (i_k, k = 1, ..., m)$ is the vector of link currents, $C = (c_k, k = 1, ..., m)$ is the vector of link current capacities, and $R = (r_k, k = 1, ..., m)$ is the vector of link resistances. Suppose the links are directed, with arbitrarily defined directions. When the current $i_k$ flows along the direction of link $e_k$ then $i_k > 0$; otherwise $i_k < 0$.

### 5.1.1 Objective Function

The power loss $p_k$ on a link $e_k$ is

$$p_k = r_k {i_k}^2 \quad (-c_k \leq i_k \leq c_k), \tag{5.1}$$

and the power loss $P_G$ of an entire graph is given by the sum of the power losses on all the links:

$$P_G = \sum_{e_k \in E} p_k = I^T R I. \tag{5.2}$$

The power loss minimization problem can be considered as an optimization problem over the current vector $I$. Note that power systems are usually controlled by changing the power injections at the vertices instead of directly controlling the currents. However, to convey the idea of power loss minimization, we select the link currents as the variable. In implementation, currents can be controlled via, e.g., controlling the voltages at the vertices.

We assume constant current injections $J \in \mathbb{R}^n$ at the vertices $v_i$, $i = 1, ..., n$. The link currents $I$ should satisfy the current injection constraints imposed by $J$. Let $A$ be the $n \times m$ incidence matrix, with $A_{ik} = 1$ if vertex $v_i$ is the source of link $e_k$, $-1$ if vertex $v_i$ is the sink

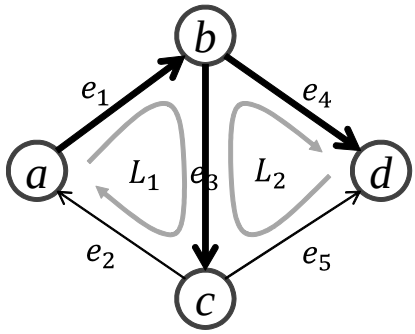



Figure 5.1: Example of tie-set flows.

of link $e_k$, and 0 otherwise. Then, we have

$$AI = J. \tag{5.3}$$

By [46], if the graph is a mesh (not a tree), given $J$, there are a subspace of solutions $I$ to (5.3). Our objective is to find the solution $I$ that minimizes the power loss $P_G$ in (5.2).

Given $C$ and $R$, the *Power Loss Minimization (PLM)* problem is

$$\min_{e_k} \quad P_G = \sum_{e_k \in E} p_k = I^T R I. \tag{5.4}$$

$$\text{s. t.} \quad -C \leq I \leq C. \tag{5.5}$$

To develop a decentralized algorithm to solve this problem using tie-set graph theory, we convert $P_G$ as a function of tie-set currents as shown below.

### 5.1.2 Problem Formulation with Tie-set Currents

Let $x = (x_1, x_2, ..., x_\mu) \in \mathbb{R}^\mu$ denote the vector of tie-set currents which circulate along the tie-sets $\mathbb{L}_B = \{L_1, ..., L_\mu\}$. Note that the tie-sets have arbitrarily defined directions where

$x_\lambda > 0$ if it is flowing in the same direction as the direction defined for $L_\lambda$, and $x_\lambda < 0$ otherwise. Let $B = [b_{k\lambda}] \in \mathbb{R}^{m\times\mu}$ be the *tie-set matrix* of the graph $(V, E)$ with respect to $\mathbb{L}_B$, where

$$b_{k\lambda} = \begin{cases} 0 & e_k \notin L_\lambda \\ 1 & e_k \in L_\lambda \text{ and in the same direction with } L_\lambda \\ -1 & e_k \in L_\lambda \text{ and in the opposite direction with } L_\lambda. \end{cases}$$

For example, $B = [b_{k\lambda}]$ in Figure 5.1 is

$$B = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & -1 \\ 0 & 1 \\ 0 & -1 \end{bmatrix}.$$

As stated above, in a mesh network, there are a subspace of solutions to (5.3) for a given $J$. Select an arbitrary solution of them, denoted by $\xi$, which is also called an *initial current vector*. Then, the current vector $I$ can be expressed in terms of the tie-set currents $x$ and the initial currents $\xi$ as

$$I = Bx + \xi,$$

since one can easily show that $AB = 0$. Then, $P_G$ in (5.2) can be transformed into a function of tie-set currents:

$$\begin{aligned} P_G(x) &= (Bx+\xi)^T R^d (Bx+\xi) \\ &= x^T B^T R^d B x^{\mathrm{T}} + 2\xi^T R^d B x + \xi^T R^d \xi \\ &= x^T M x + 2Nx + e, \end{aligned}$$

where $M := B^T R^d B$, $N := \xi^T R^d B$, and $e := \xi^T R^d \xi$.

Given $B, C, R$, and $\xi$, the PLM problem, with respect to tie-set currents, is

$$\min_x \quad P_G(x) = xMx^{\mathrm{T}} + 2xN + e \tag{5.6}$$

$$\text{s. t.} \quad -C \le Bx + \xi \le C. \tag{5.7}$$

As $P_G(x)$ is differentiable on $\mathbb{R}^\mu$, the necessary and sufficient condition for $x^* \in \mathbb{R}^\mu$ to be the optimal point of the PLM problem in (5.6) and (5.7) is that

$$\nabla P_G(x^*) = \left( \frac{\partial P_G(x^*)}{\partial x_1}, ..., \frac{\partial P_G(x^*)}{\partial x_\mu} \right) = 0. \tag{5.8}$$

Now, we define a power loss function $P_\lambda(x)$ with respect to a tie-set $L_\lambda$ as

$$P_\lambda(x) = \sum_{e_k \in L_\lambda} p(i_k) \tag{5.9}$$

For each element of the vector, the following equation holds:

$$\frac{\partial P_G(x^*)}{\partial x_\lambda} = \sum_{e_k \in E} \frac{\partial p(i_k^*)}{\partial x_\lambda} = \sum_{e_k \in L_\lambda} \frac{\partial p(i_k^*)}{\partial x_\lambda} = \frac{\partial P_\lambda(x^*)}{\partial x_\lambda} = 0. \tag{5.10}$$

According to (5.8), each power loss function $P_\lambda(x)$ with respect to a tie-set $L_\lambda$ is minimum at $x = x^*$, i.e.

$$\left( \frac{\partial P_0(x^*)}{\partial x_0}, \frac{\partial P_1(x^*)}{\partial x_1}, ..., \frac{\partial P_\mu(x^*)}{\partial x_\mu} \right) = 0. \tag{5.11}$$

The equation (5.11) represents that if each power loss function $P_\lambda(x)$ with respect to $L_\lambda$ is minimized, the overall power loss $P_G$ becomes minimum as well. Therefore, local optimization with respect to $\mu$-independent tie-sets must lead to the globally optimal solution.

## 5.2 Autonomous Distributed Control Model for PLM

### 5.2.1 TADiC for PLM

As described in Chapter 2, TADiC is conducted in a leader node in each tie-set asynchronously. A Tie-set Agent (TA) constantly brings the information of currents on the links of a tie-set $L_\lambda$ to its leader node. We define Tie-set Evaluation Function (TEF) $\Phi(L_\lambda, t)$, which decides the process priority for overlapping resources (power flows) shared by adjacent tie-sets, as

$$\Phi(L_\lambda, t) = \left| \frac{d}{dx_\lambda} p_\lambda(x_\lambda) \right| \tag{5.12}$$

where $p_\lambda(x_\lambda)$ is defined in (5.19) in Section 5.2.2. Only when the process priority cannot be determined by the TEF in (5.12), we use $\Phi(L_\lambda, t) =$ Random. TEF $\Phi(L_\lambda, t)$ is calculated

based upon the values of initial flows on edges in $L_\lambda$ at time $t$ from Measurement Vector (MV) $y_\lambda(t)$ obtained by TA. Tie-set Flags (TF) are decided by the values of TEFs.

### 5.2.2 Decentralized Algorithm for PLM

*Decentralized Algorithm for Power Loss Minimization (DAPLM)* is described in Algorithm 7 and called by the OPTIMIZE step of TADiC.

Let $\gamma$ be the number of edges in a tie-set $L_\lambda = \{e_1, e_2, ..., e_\gamma\}$ where $\gamma = |L_\lambda|$. Then, we define the followings:

$$B_\lambda = (b_{\lambda 1}, b_{\lambda 2}, ..., b_{\lambda\gamma}) \tag{5.13}$$

$$I_\lambda = (i_1, i_2, ..., i_\gamma) \tag{5.14}$$

$$R_\lambda = (r_1, r_2, ..., r_\gamma) \tag{5.15}$$

$$\xi^\lambda = (\xi_1, \xi_2, ..., \xi_\gamma) \tag{5.16}$$

With respect to a tie-set flow $x_\lambda$ of $L_\lambda$, the electric current vector $I_\lambda$ of $L_\lambda$ is defined as follows:

$$I_\lambda = x_\lambda B_\lambda + \xi^\lambda \tag{5.17}$$

Let $p_\lambda(x_\lambda)$ be a *Tie-set Power Loss (TPL)* function of a tie-set $L_\lambda$ as

$$p_\lambda(x_\lambda) = I_\lambda^{\mathrm{T}} R_\lambda^d I_\lambda \tag{5.18}$$

where $R_\lambda^d = \mathrm{diag}(r_1, r_2, ..., r_\gamma)$ is a diagonal matrix whose elements are $R_\lambda = (r_1, r_2, ..., r_\gamma)$.

On the basis of the definitions (5.13) - (5.18), the TPL function $p_\lambda(x_\lambda)$ is:

$$p_\lambda(x_\lambda) = \sum_{e_k \in L_\lambda} r_k (b_{\lambda k} x_\lambda + \xi_k)^2 \tag{5.19}$$

In STEP 0, DAPLM initializes the value of a current tie-set flow $x_\lambda$ of $L_\lambda$ as 0. The information of current edge flows $\xi^\lambda$ on $L_\lambda$ is provided by MV $y_\lambda(t)$ on Tie-set Agent (TA).

In STEP 1, DAPLM calculates the optimal tie-set flow $x_\lambda$ to satisfy

$$\begin{aligned} \frac{d}{dx_\lambda} p_\lambda(x_\lambda) &= \frac{d}{dx_\lambda} \sum_{e_k \in L_\lambda} r_k (b_{\lambda k} x_\lambda + \xi_k)^2 \\ &= \sum_{e_k \in L_\lambda} \frac{d}{dx_\lambda} r_k (b_{\lambda k} x_\lambda + \xi_k)^2 \\ &= 0 \end{aligned} \tag{5.20}$$

i.e.,

$$x_\lambda = -\frac{\sum_{e_k \in L_\lambda} r_k b_{\lambda k} \xi_k}{\sum_{e_k \in L_\lambda} r_k}, \tag{5.21}$$

since $b_{\lambda k}^2 = 1$.

In STEP 2, each edge flow $i_k \in I_\lambda$ is updated to $b_{\lambda k} x_\lambda + \xi_k$ where edge flows on $L_\lambda$ are optimized to minimize the power loss in $L_\lambda$.

An example of DAPLM is shown in Figure 5.2. In the tie-set $L_\lambda = \{e_1(b,a), e_2(a,c), e_3(c,b)\}$

**Algorithm 7** Decentralized Algorithm for Power Loss Minimization (DAPLM)

STEP 0:
  Initialize a current tie-set flow $x_\lambda$ of $L_\lambda$ as 0.
  Obtain $\xi^\lambda$ provided by MV $y_\lambda(t)$ on TA.
STEP 1:
  Calculate $x_\lambda$ according to (5.21).
STEP 2:
  **for** each $i_k$ on $e_k \in L_\lambda$ **do**
    Update an electric current $i_k$ to $b_{\lambda k}x_\lambda + \xi_k$.
    Inject the minimized lost power $r_k {i_k}^2$ at the destination node of the current $i_k$.
  **end for**

in Figure 5.2,

$$B_\lambda = (b_{\lambda 1}, b_{\lambda 2}, b_{\lambda 3}) = (1, 1, 1),$$
$$\xi^\lambda = (\xi_1, \xi_2, \xi_3) = (9.0, 9.0, 2.0),$$
$$R_\lambda = (r_1, r_2, r_3) = (0.2, 0.1, 0.05).$$

According to (5.19), the TPF function in $L_\lambda$ is

$$\begin{aligned} p_\lambda(x_\lambda) &= r_1(b_{\lambda 1}x_\lambda + \xi_1)^2 + r_2(b_{\lambda 2}x_\lambda + \xi_2)^2 + r_3(b_{\lambda 3}x_\lambda + \xi_3)^2 \\ &= 0.2 \times (1 \cdot x_\lambda + 9.0)^2 + 0.1 \times (1 \cdot x_\lambda + 9.0)^2 + 0.05 \times (1 \cdot x_\lambda + 2.0)^2 \\ &= 0.35x_\lambda^2 + 5.6x_\lambda + 24.5 \end{aligned}$$

In STEP 0, tie-set flow $x_\lambda$ is initialized 0; the value of TPL function is $p_\lambda(0) = 24.5$. In

STEP 1, the optimal $x_\lambda$ is calculated according to (5.21) as follows:

$$\begin{aligned} x_\lambda &= -\frac{r_1 b_{\lambda 1}\xi_1 + r_2 b_{\lambda 2}\xi_2 + r_3 b_{\lambda 3}\xi_3}{r_1 + r_2 + r_3} \\ &= -\frac{(0.2 \cdot 1 \cdot 9.0) + (0.1 \cdot 1 \cdot 9.0) + (0.05 \cdot 1 \cdot 2.0)}{0.2 + 0.1 + 0.05} \\ &= -8.0 \end{aligned}$$

Then, the currents are updated in STEP 2 as

$$\begin{aligned} i_1 &= b_{\lambda 1} x_\lambda + \xi_1 = 1 \cdot (-8.0) + 9.0 = 1.0 \\ i_2 &= b_{\lambda 2} x_\lambda + \xi_2 = 1 \cdot (-8.0) + 9.0 = 1.0 \\ i_3 &= b_{\lambda 3} x_\lambda + \xi_3 = 1 \cdot (-8.0) + 2.0 = -6.0 \end{aligned}$$

so that the current vector in $L_\lambda$ is

$$I_\lambda = (i_1, i_2, i_3) = (1.0, 1.0, -6.0)$$

In summary, the power flow loss in $L_\lambda$ of Figure 5.2 before conducting DAPLM is $p_\lambda(0) = 24.5$, and the loss after DAPLM is $p_\lambda(-8.0) = 2.1$ where the lost power $0.2 \times 1.0^2 = 0.2$ (W) on the link $e(b, a)$ is injected at node $a$, and the lost power $0.1 \times 1.0^2 + 0.05 \times (-6.0)^2 = 1.9$ (W) on the links $e(a, c)$ and $e(c, b)$ is injected at node $c$.

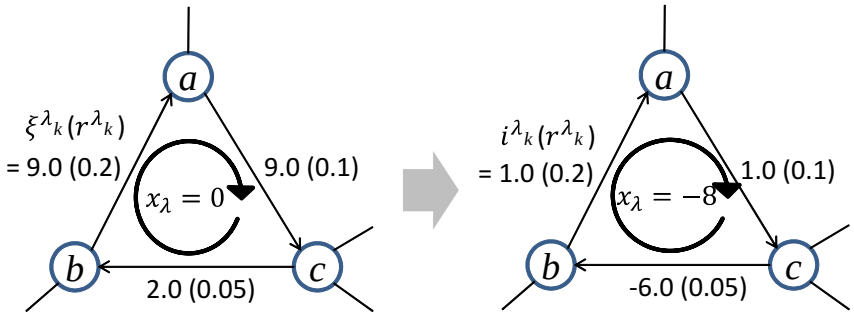



Figure 5.2: Example for DAPLM.

## 5.3 Simulation and Experiments for PLM

The PLM problem can be solved through local optimizations in a system of $\mu$-independent tie-sets that are equivalent to $\mu$-dimensional linear vector space as discussed in Section 5.1.2. By iterating the decentralized algorithm described in Section 5.2.2, optimal flows that minimize the power line loss of a graph can be obtained. To verify this property and conduct experiments as well as compare it with a centralized approach, we made a simulator in Java that realizes the distributed algorithms to solve the PLM problem. Common buffering method and polling method are employed in a simulation node.

In this experiment, each link capacity is $c_k = 100$ (A) $(k = 1, 2, ..., m)$. Tie-sets exchange $\Phi(L_\lambda, t) = \left|\frac{d}{dx_\lambda} p_\lambda(x_\lambda)\right|$ as TEF, and every MV $y_\lambda(t)$ is constantly sent to a leader node of each tie-set. Communication Interval in StandBy of TADiC is $\Delta t = 1$ ms in this simulation. When all the tie-sets satisfy the following termination condition, simulation procedure stops.

$$-0.1 < \frac{d}{dx_\lambda} p_\lambda(x_\lambda) < 0.1 \tag{5.22}$$

### 5.3.1 Experiments Using IEEE Bus Test Systems

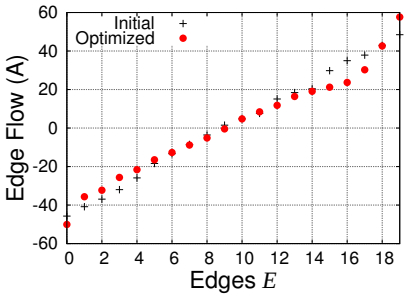
In this experiment, the simulation network $G = (V, E)$ and its properties are given on the basis of IEEE bus test systems with 14, 30, 57, 118 buses and 20, 41, 78, 179 lines, respectively. The sets of link resistances, generators, and loads are provided by the IEEE bus systems. We create fundamental systems of tie-sets on the bus systems in a distributed manner. The initial flow $\xi_k$ of each edge is randomly assigned with $-50 \leq \xi_k \leq 50$ (A). Experimental data are taken when every tie-set satisfies (5.22), and conducted 10 times from the section below so that the average value is calculated.

#### Optimized Edge Flows

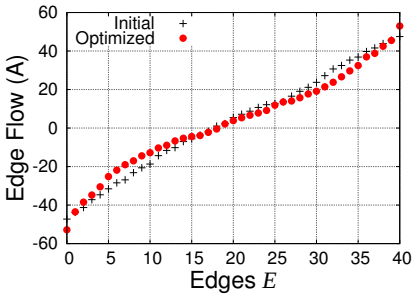
Figure 5.3 shows the values of edge flows (electric currents (A)) on $e_k \in E$ before and after optimization in the 14-bus, 30-bus, 57-bus, and 118-bus grid. In Figure 5.3, edge flows are sorted in ascending order. All the edge flows are balanced to minimize the power loss of $G$ by iteratively updating the tie-set flow $x_\lambda$ in each tie-set $L_\lambda$. From Figure 5.3, edge flows in the 118-bus system are more balanced as the grid has the larger number of links compared with other bus systems.

#### Current Distribution based on Resistance

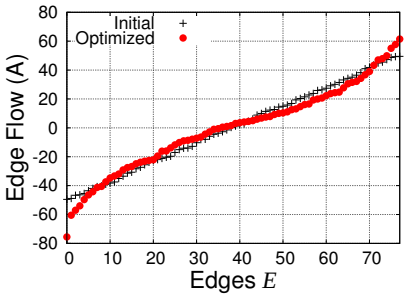
Figure 5.4 is another way of showing edge flows on resistances $r_k \in R$ sorted in ascending order. As indicated in Figure 5.4, the variance of optimized edge flows is larger on smaller resistances and smaller on lager resistances, which means the large amount of edge flows pass through links whose resistances are small.



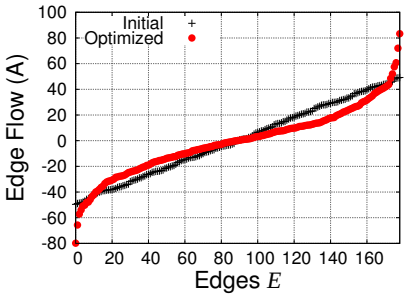

(a) 14-bus system.




(b) 30-bus system.

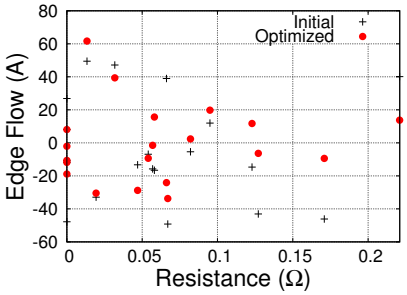



(c) 57-bus system.

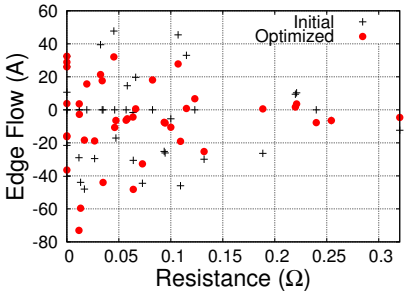



(d) 118-bus system.

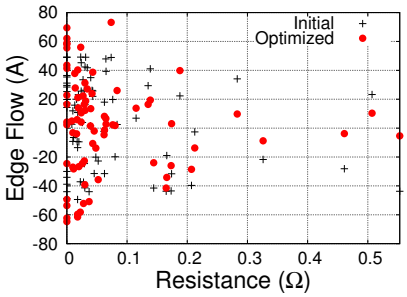
Figure 5.3: Edge flows $i_k$ on edges $e_k \in E$ before and after optimization in the IEEE bus test systems.



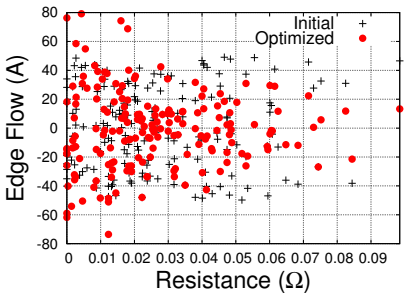

(a) 14-bus system.




(b) 30-bus system.

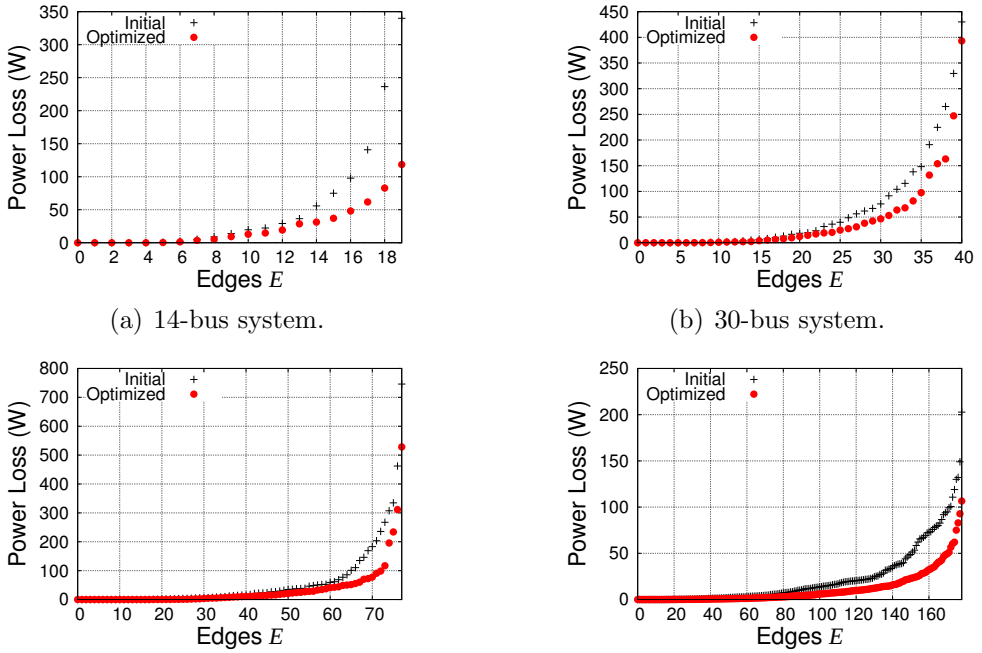



(c) 57-bus system.




(d) 118-bus system.

Figure 5.4: Edge flows $i_k$ sorted by resistances $r_k$ on edges $e_k \in E$ before and after optimization in the IEEE bus test systems.




(a) 14-bus system.

(b) 30-bus system.

(c) 57-bus system.

(d) 118-bus system.

Figure 5.5: Power losses $p_k$ on edges $e_k \in E$ before and after optimization in the IEEE bus test systems.

### Optimized Power Flow Losses

Figure 5.5 shows the power loss (W) on each edge sorted in ascending order as well. With the optimized edge flows, the power loss of every link is minimized. This result demonstrates that individual power losses on edges by optimized flows are almost half of the losses by initial flows.

### Total Power Loss by Initial and Optimized Flows

Table 5.1 shows the results of the total power loss by initial and optimal power flows with IEEE 14-bus, 30-bus, 57-bus, and 118-bus systems. Those data are taken 10 times and the averaged value has been calculated. In Table 5.1, "Initial $P_G$" represents the total power loss by not-optimized power flows on all the edges of a graph, "Optimal $P_G$" represents the total power loss by optimized power flows.

Table 5.1: Comparison of the total power loss by initial and optimized flows with IEEE bus test systems

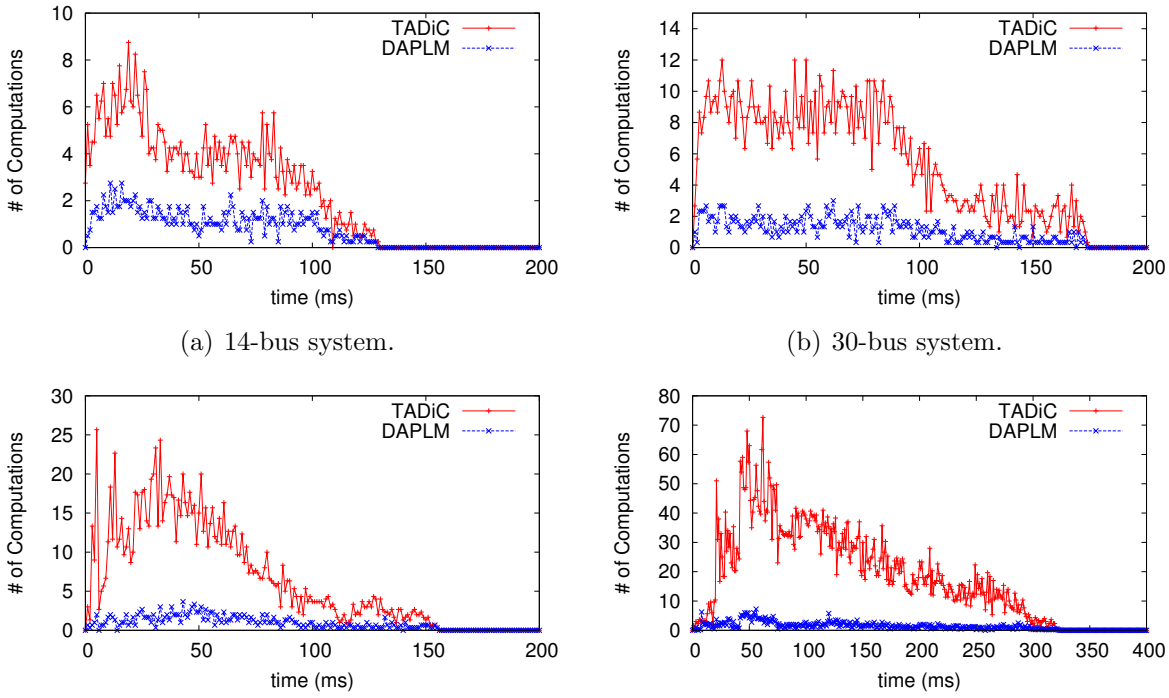
| | 14 buses | 30 buses | 57 buses | 118 buses |
|---|---|---|---|---|
| Initial $P_G$ (W) | 888.055 | 2586.321 | 4708.256 | 3980.594 |
| Optimal $P_G$ (W) | 413.982 | 1786.854 | 2938.289 | 2023.729 |

As suggested in Table 5.1, the optimal flows in 14-bus and 118-bus systems achieve about a 50% reduction of power loss. As to 30-bus and 57-bus systems, more than a 30% reduction of power loss have been realized.

**Analysis on Property Performance**

Now we analyze the property performance of the proposed decentralized algorithm. In Figure 5.6, the number of computations for TADiC and DAPLM at each time step is provided in the IEEE bus systems. As indicated in Figure 5.6, all the tie-sets of 17-bus, 30-bus, 57-bus, and 118-bus system stop their procedures (TADiC and DAPLM) when $t$ = 130, 175, 156, and 323 ms, respectively, as they satisfy (5.22). The behaviors of those results show that the proposed method conducts many times of both TADiC and DAPLM until about 100 ms, and then reduces the computations after 100 ms. This result shows that distributed optimization technique based on tie-sets tries to realize overall optimization quickly, and then continues its procedures until it converges to satisfy (5.22).

As to Communications Complexity $\Gamma(t)$ at time $t$, which is the number of messages used in conducting TADiC, we can calculate it with $\Gamma(t) = 2 \times \alpha(t) \times \sum_{L_\lambda \in \mathbb{L}_B} |\mathbb{L}_\lambda^a|$ where $\alpha(t)$ is the total number of computations by TADiC at time $t$. For example, in the IEEE 30-bus system used in this simulations, the Communications Complexity is $\Gamma(10) = 2 \times 8 \times 84 = 1344$ where $\alpha(10) = 8$ and $\sum_{L_\lambda \in \mathbb{L}_B} |\mathbb{L}_\lambda^a| = 84$.




(a) 14-bus system.

(b) 30-bus system.

(c) 57-bus system.

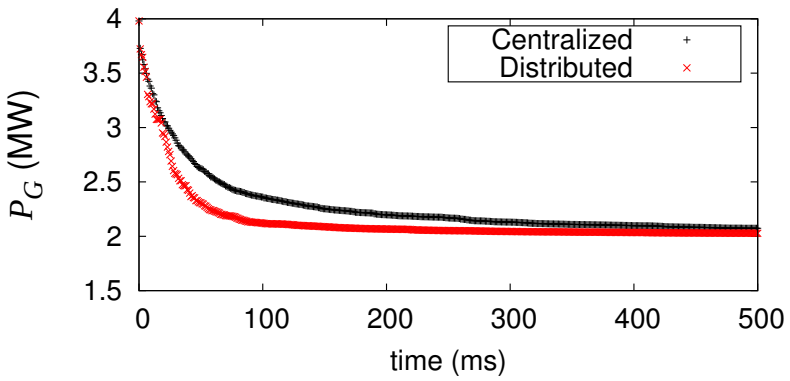
(d) 118-bus system.

Figure 5.6: The number of computations of TADiC and DAPLM at every time step in the IEEE bus systems.

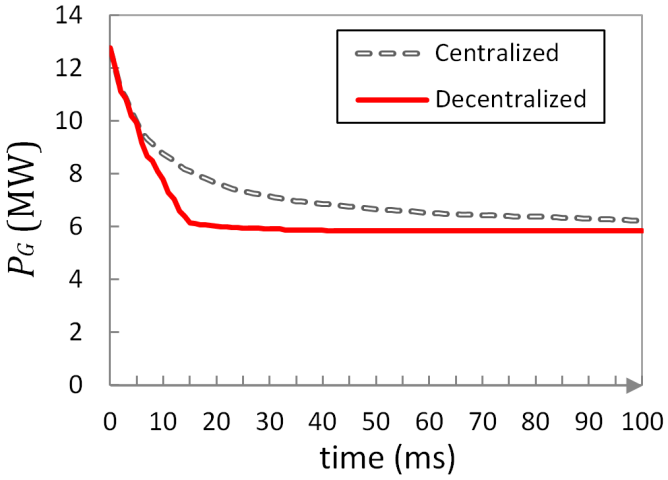
### 5.3.2 Comparison with Centralized Optimization

In this experiment, we show the effectiveness of the decentralized algorithm based on tie-sets by comparing convergence speed with the centralized optimization approach. The centralized approach sequentially optimizes flows by picking a tie-set whose TEF is the largest among all the tie-sets. Initial flows are randomly given to all the edges $E$ of $G$ with $-50 \leq \xi_k \leq 50$ $(k = 1, 2, ..., m)$.

Figure 5.7 shows the result of the comparison experiment about the convergence speed in the IEEE 118-bus grid and a 100-bus grid with random link connections. In the 100-bus grid, each link resistance $r_k$ is randomly given between 0 to 0.5 ($\Omega$). As clearly shown in Figure 5.7, the decentralized algorithm based on tie-sets demonstrated fast convergence than the centralized scheme. This is because the decentralized method realizes parallel optimizations




(a) IEEE 118-bus system.




(b) 100-bus grid with random link connections.

Figure 5.7: Comparison of convergence in power loss $P_G$ with centralized optimization in IEEE 118-bus and 100-bus grids.

by communicating with adjacent tie-sets whereas the centralized method collects all the data of current flows from each node to conduct optimization. Because the distributed algorithm shows its fast convergence nature, the result also suggests that centralized systems should be replaced by autonomous distributed systems to realize efficient parallel computation in a future power network where power flows are constantly updated with stochastic process of loads and renewables.

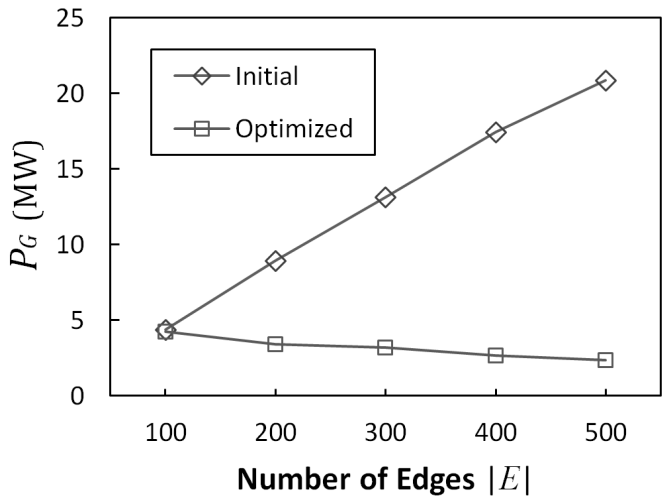



Figure 5.8: Power loss $P_G$ with the different number of edges (100 to 500) with a 100-bus grid.

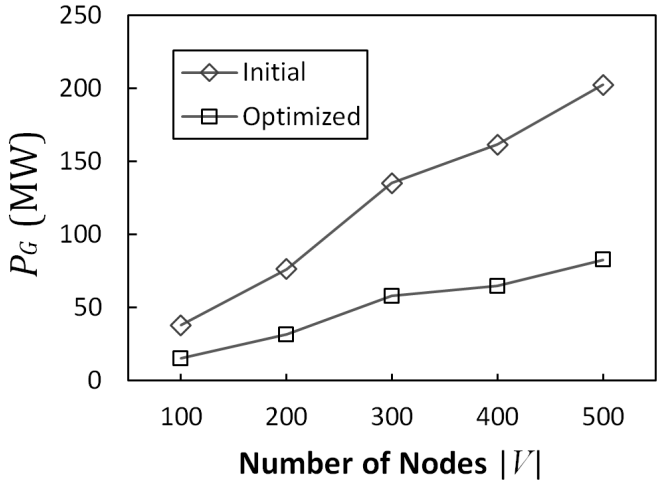
### 5.3.3 Performance in Different Topologies

We have analyzed the specific grids using IEEE bus test systems. Therefore, we analyze the nature of the proposed optimization method using random mesh topologies with different size of nodes and edges. Initial flows are given to all the edges $E$ of $G$ with $-50 \leq \xi_k \leq 50$ $(k = 1, 2, ..., m)$ (A). Experiment data are taken when all the tie-sets satisfy (5.22). We conducted 20 times of experiments for each simulation network to calculate the average value.

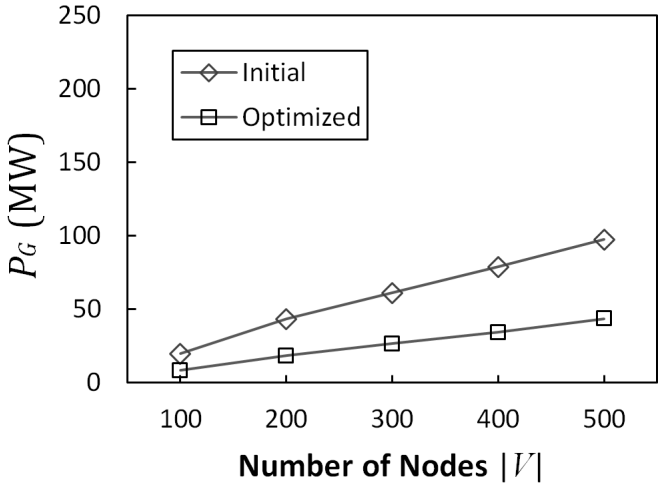
#### Power Loss with Different Number of Links

In this experiment, we conducted a simulation to examine the power loss $P_G$ of an entire graph with the different number of edges with a 100-bus grid where $|E| = 100, 200, 300, 400,$ and 500. Each link resistance $r_k$ is randomly given between 0 to 0.1 ($\Omega$).

As seen in Figure 5.8, the total power loss $P_G$ by initial (not-optimized) flows shows large increase at a rate proportional to the number of edges, whereas the loss by optimized flows




(a) Initial flows are given to all the edges $E$ of $G$.




(b) Initial flows are only given to the edges of tree $T$ of $G$.

Figure 5.9: The total power loss $P_G$ with the different number of nodes (100 to 500).

shows a subtle decrease as the number of edges increases. The delta between the loss by initial flows and optimized flows significantly increases as the number of power lines becomes larger. This result indicates that the larger the number of lines becomes, the more amount of the power loss we can reduce. However, we need to consider the cost of installing power lines when we configure a power network.

**Power Loss with Different Network Sizes**

Next, we conducted a simulation to look at the power loss $P_G$ of an entire graph with the different number of nodes where $|V| = 100, 200, 300, 400$, and 500 with the number of links $|E| \approx 200, 400, 600, 800$, and 1000. Each link resistance $r_k$ is randomly given between 0 to 0.5 ($\Omega$).

In the experiment shown in Figure 5.9(a), initial edge flows are given to all the edges $E$ of $G$, whereas initial flows in Figure 5.9(b) are only given to the edges of tree $T$ of $G$. Figure 5.9(a) indicates that the power loss $P_G$ is proportionate to the number of nodes, and the total loss by optimized flows is less than the half of the total loss by initial (not-optimized) flows. Figure 5.9(b) shows similar tendency to the result of Figure 5.9(a). From the result shown in Figure 5.9(b), the traditional tree or bus topology is quite inefficient for future grids with bi-directional flows. This result gives rise to use of mesh topologies to future power networks.

## 5.4 Summary

In this chapter, a power loss minimization (PLM) problem is first formulated on the basis of tie-set graph theory to distribute electricity with optimized link flows targeting prospective future grids. Then, we proposed the decentralized algorithms based on tie-sets that represent loops in a power network to solve the PLM problem and described how to realize autonomous distributed control with autonomous agents navigating the tie-sets.

The simulation results show that local optimizations constantly iterated within a fundamental system of tie-sets calculate optimal edge flows of entire grids based on the IEEE bus test systems. The results also show that the proposed method realizes more than a 50% reduction of power loss for a grid with 100 to 500 nodes.

We are integrating this proposed method with an optimal real-time power flow model that is also being developed to compensate the renewable intermittency by using battery systems. Our parallel optimization algorithm to minimize the power flow loss will realize effective use of renewables, which we will consider with random power inputs and outputs of loads and renewables determined at each time step in simulation. We will also consider the stochastic situations with time varying conditions.

# Chapter 6

# Related Work

In this chapter, the research work related to the topics of the thesis is discussed in terms of real-time power flow control with energy storage and optimal power flow (OPF) for minimizing power flow loss.

## 6.1 ORDER

Demand response is playing an important role in smart grid by which we can cut the peak load and adapt elastic demand. There are a lot of papers that deal with demand response such as [47, 48]. [47, 48] are studied at Lawrence Berkeley National Laboratory where they consider the electricity load control with thermal mass in buildings. [49] considers the coordination of charging plug-in hybrid electric vehicle (PHEV) with other electric appliances. Various sophisticated techniques in residential and power networks to realize optimal demand response have been proposed as in [50, 10]. [50] presents a new energy management system for residential demand response. The algorithm in [50] reduces residential energy costs and smooths energy usage with an online learning application that can implicitly estimates the

impact of future energy prices as well as consumer's decision on how they use residential devices. In [10], it is studied how energy storage at utility premises maximize the efficiency in demand response. It is based on the fact that each household may have a different objective for different appliances besides waiting time for the operation of the appliance. Our work considers complicated network topology cases when fully applied to the future complex power networks.

The use of battery systems has been a key to realize optimal distribution of renewables. In [51], optimal management strategy to improve renewable energy integration based on storage systems in distribution networks has been discussed. A dynamic programing based approach seen in [52] is applied to [51] to define the optimal generation profile of the whole generation facility. This gives us a motivation to compare our approach with the dynamic programming algorithm, which has been widely used in solving distributed problems.

## 6.2 ORPF

Optimal power flow (OPF) problems, first formulated and studied by Carpentier [53], have been studied for more than half a century. Surveys and reviews of OPF can be found in [54, 55, 56, 57]. According to the literature, the optimization schemes of OPF has been sophisticated over those period. Many of the earlier models for OPF focused on static optimizations, i.e., optimizations for isolated periods of time in which power supply and demand must be balanced at every period.

The complexity of the optimal power flow (OPF) increases dramatically once real-time events and security are taken into consideration. For example, [58] modeled dynamic constrained optimal power flow as a semi-infinite programming problem to replace the infinite constraints for dynamic constraints with equivalent finite constraints. In [59], a synthetic dynamics

power flow (SDPF) approach is explored based on the principle that the dynamic response of a stable power system always decays to a stable equilibrium point. More recently, a two-level architecture has been studied in [60] where it distributes the control and computation burden to multiple area controllers for dynamic stochastic optimal power flow (DSOPF), and reduces the training difficulty for implementing the DSOPF control for a large power network.

While dynamic OPF has been intensely studied, it is increasingly common for energy storage devices such as batteries to be installed and used in power grids [61, 62]. By charging and discharging the batteries, mismatches between instantaneous power supply and demand can be balanced. As a result, the requirement of power balance at every instant in the OPF formulation can be relaxed. Multi-period versions of OPF problems can be considered. in which the battery charging/discharging profile becomes one of the control variables over which the optimization is defined.

The work [63] copes with the OPF with energy storage with deterministic demand process and simple network topologies. In this paper, we adopt a model similar to that of [63] by utilizing energy storage systems. The model explored in this paper captures the fully intermittent (stochastic) issues when integrating renewable resources as well as focuses on the effect of energy storage to dynamic optimal power flow. Therefore, we exploit the model with stochastic renewable generation and demand process and do not assume the full knowledge of the demand schedule.

The work in [29] deals with balanced allocation of static distributed energy resources stored in energy storage systems and introduces the simulation results showing that the iterative optimization within the system of fundamental system of tie-sets leads to global solution whose theoretical property can be verified in [27]. On the basis of [29], the work achieves the balanced allocation of renewables with the complete automation of a prospective future grid that integrates batteries and end-use devices to capture stochastic process of loads

and renewable generations automatically [31]. Our previous work on decentralized optimal real-time power flow (ORPF) control is proposed aiming at optimizing power flows with renewable integration [32]. This paper extends the model of ORPF so that this dynamic OPF technique can be applied to a variety of power network settings with more flexible functionalities.

## 6.3 PLM

Many efforts have been made in the domain of power loss minimization as in [55] even before the development of communications systems that are being integrated into the power systems domain now. Power flow loss minimization for current power systems is discussed in [64, 65]. [65] basically presents voltage control methods to improve the voltage level of a power system whilst minimizing losses. As in the work, the Optimal Power Flow (OPF) problem classically is nonlinear and non-convex aiming to minimize the power generation costs and transmission loss in a power network. It has been subject to physical constraints based on Kirchhoff's and Ohm's law [55, 53, 66].

In the past decade, devising efficient algorithms with guaranteed performance has been much attention for the OPF problem. For example, nonlinear interior-point algorithms for an equivalent current injection model of the problem have recently proposed in [62] and [67].

The recent work [68] has improved implementation of the automatic differentiation technique for the OPF problem. To convexify the OPF problem, the work [69] justifies that the load flow problem of a radial distribution system can be modeled as a convex optimization problem in the form of a conic program. To overcome the nonlinearity, the majority of these works focus on linearization and approximation schemes to simplify the OPF. The work [70], for example, simplifies the OPF problem by the small angle approximation. Some other

relaxation methods can be found in [71, 72]. However, because of the presence of arctangent equality constraints, those results struggle to hold for a meshed network [73].

The recent work also proves that a closely related problem of finding an optimal operating point of a radiating antenna circuit is an NP-complete problem, by reducing the number partitioning problem to the antenna problem [74]. In [75], the DC optimal power flow (OPF) problem that minimizes the power loss in an electrical network is proposed by optimizing the voltage and power delivered at the network nodes. [75] studies the direct current special case by leveraging recent developments on the zero duality gap of OPF [74], and presents a decentralized algorithm based on updating primal-dual variables.

Different from prior work on decentralized algorithms for OPF, for example, [75], which mainly focus on updating nodal variables like voltage and dual variables attached to nodes, in this paper we design a decentralized algorithm based on updating the loop variables. Intuitively, our method should be more efficient and cost less computational time given the sparse structure of power grids, i.e., there are usually much fewer loops than the nodes. The OPF model in this paper is a convex optimization model with quadratic flow costs, and [17] implies that a practical power loss minimization problem is likely to be solvable using a convex algorithm. By focusing on the notion of tie-sets, we can effectively divide the grid and form a $\mu$-dimensional linear vector space[1]. Our objective is to find an optimal flow that minimizes the power flow line losses in a meshed power network with distributed computations. We formulate the power loss minimization problem based on the currents, which has been common formulation in circuits and systems and also is applicable to the future power systems. It is elegantly formulated with redirection of currents, which has played significant role in circuits and systems [40].

[1] $\mu$ is the nullity of a graph as defined in Section 2.1.2

# Chapter 7

# Conclusions

This thesis proposes a real-time flow control and power production response model for managing optimal future power networks and a model for designing future power networks that make the most use of

- Millions of real-time end-use devices;
- Individual storage devices;
- Renewable energy produced locally;
- Bi-directional energy flow among individual nodes.

TADiC proposed in chapter 2 is advocated as the basic control model to take full advantage of the features above and applied to each problem discussed chapter 3 - 4.

**ORDER**

The decentralized model for ORDER is provided in chapter 3 by integrating TADiC. The application of TADiC achieves the balanced allocation of distributed energy resources stored

in batteries at consumer premises. In one hundred-node simulation network, convergence of node energies are done within 300 minutes and excessive power supply has been minimized by the convergence effect. Once the energy converges on the Lower Energy Limit (LEL), the energy level keeps to be stable since TADiC constantly dispatches the power from CGF to prevent shortage of energy in the storage systems. Each node also distributes power independently based on the commands from the leader node in each tie-set, which completely automates the future grid. The simulated behavior with different RPRs (Renewable Penetration Rates) demonstrates that when the RPR is 100% or more, we will not need to produce power at CGF since the decentralized model constantly distributes the energies from the nodes with a lot of energy to the nodes in short of energy. Even though we consider the real power flow model with voltages and phase angles, the behavior of convergence and CGF power supply does not differ significantly from the power flow model with independent power lines. Comparison experiments with other schemes such as centralized power provision method and distributed linear programing approach also show that the proposed model is effective in terms of minimizing CGF power and convergence behavior. Therefore, it has been substantiated by simulation experiments that global optimization of balanced distribution of dispersed static energies is achieved by iterative local optimization within a fundamental system of tie-sets.

**ORPF**

An Optimal Real-Time Power Flow (ORPF) model is discussed by defining each tie-set as linear vector space and minimizing the costs of CGF and batteries. The simulated power stimulus response at each node demonstrates the effective use of each resource. For example, the battery energy level at a node at each time fluctuates depending on the sudden load peak and renewable generation since the completely random process is produced in experiments. However, the energy level indicates that it never runs out and keeps stable behavior because

of the constant optimization in the tie-set, and the other stimulus responses make up for the load and renewable peaks by adjusting the amount of CGF and in-and-out power flows. The result at root node demonstrates the different power responses since it is involved in the optimization precess many times so that the generation of CGF and the amount of outgoing power flow are much larger than the other nodes. Nonetheless, the average power of all the nodes keeps stable once it becomes close to the capacity of energy storage, which enables sustainable grid even if the size of the network becomes large. Therefore, we have substantiated that tie-set based optimization realizes a reliable and sustainable use of battery systems where the energy in the storage systems never runs out even if the level of energy fluctuates depending on the load process, renewable generation, and CGF generation.

This model is expanded as in the later part of the chapter where we classify the nodes into two types; generation nodes and demand nodes. The power supply at demand nodes should be exactly met at all the times, which has been demonstrated with the simulation results. The stimulus power responses are similar to the results of the case that all the nodes have energy storage as well as functions of power flows, loads, and renewable generation. The scalability of the proposed model has also been verified where the nature of $\mu$-dimensional optimization reduces the complexity of mesh networks.

**PLM**

Chapter 5 proposes how to minimize the power loss by focusing the notion call "tie-set current", which is assigned to each tie-set. The optimal values of tie-set currents assure the minimization of power flow loss so that the computational objective is to calculate the values of those currents close to the optimal values within a short period.

Simulation results demonstrate the balanced allocation of link currents where the power loss at each link has been minimized. The experiments using IEEE bus test systems are

conducted to see effectiveness of the optimization based on tie-sets in real topological power networks. The power loss is reduced with more than 50% in 14-buss system and almost 50% in other bus systems. This result shows a significant loss reduction by taking full advantage of mesh structure of a network, and the proposed method only takes less than 350 milliseconds to realize the loss minimization. Distributed parallel control works better than the centralized scheme in convergence time since the decentralized scheme assigns as many tie-sets as possible within each step of optimization. We also tested the topological aspect of the power loss problem where the connections among nodes affect the optimum value of power loss of an entire network; there is a proportionate relationship between the number of nodes/links and the power loss. Convergence time is enough small so that this scheme should be directly applicable to a dynamic power flow model as described in ORDER and ORPF chapters.

**Expected Impacts and Future Work**

The goal of the research is to establish a distributed smart grid management model that

- Uses energy storage systems effectively to optimize DERs;
- Minimizes power transmission loss;
- Realizes a reliable and sustainable future grid;
- Prevents blackouts;
- Reduces $CO_2$ emission and cost of excessive power generation.

We have dealt with essential aspects of prospective power networks so that optimization theories and algorithms being studied in the boundary domain between power systems and

distributed network systems can be integrated into the proposed models. With the characteristics of prospective power systems such as reactive power and power delivery costs, we will extend the algorithmic solution and simulation environment to incorporate them. In addition, we also need to consider the profiles of demand and/or renewable generation so that optimization time span can be adjusted to reduce computation and distribution power loss costs. After verifying the proposed models with simulation, we aim to develop and deploy real systems that realize the theories and models.

# Bibliography


[1] A. Richter, E. van der Laan, W. Ketter, and K. Valogianni. Transitioning from the traditional to the smart grid: Lessons learned from closed-loop supply chains. In *2012 International Conference on Smart Grid Technology, Economics and Policies (SG-TEP)*, pages 1–7, Dec 2012.

[2] J.A. Momoh. Smart grid design for efficient and flexible power networks operation and control. In *Power Systems Conference and Exposition, 2009. PSCE '09. IEEE/PES*, pages 1–8, 2009.

[3] Don Von Dollen. Report to nist on the smart grid interoperability standards roadmap. *Electric Power Research Institute (EPRI) and National Institute of Standards and Technology*, 2009.

[4] S.Y.R. Hui, H.S.-H. Chung, and Siu-Chung Yip. A bidirectional ac-dc power converter with power factor correction. *IEEE Transactions on Power Electronics*, 15(5):942–948, 2000.

[5] Brian Tarroja, Fabian Mueller, Joshua D. Eichman, Jack Brouwer, and Scott Samuelsen. Spatial and temporal analysis of electric wind generation intermittency and dynamics. *Renewable Energy*, 36(12):3424 – 3432, 2011.

[6] Ryan Wiser. 2010 wind technologies market report. 2012.

[7] Jean-Paul Maton, Li Zhao, and Jacob Brouwer. Dynamic modeling of compressed gas energy storage tocomplement renewable wind power intermittency. *International Journal of Hydrogen Energy*, 38(19):7867 – 7880, 2013.

[8] Cristina L Archer and Mark Z Jacobson. Evaluation of global wind power. *Journal of Geophysical Research*, 110(D12):D12110, 2005.

[9] DM Rastler. *Electricity energy storage technology options: a white paper primer on applications, costs and benefits*. Electric Power Research Institute, 2010.

[10] Na Li, Lijun Chen, and S.H. Low. Optimal demand response based on utility maximization in power networks. In *2011 IEEE Power and Energy Society General Meeting*, pages 1–8, 2011.

[11] David Connolly. A review of energy storage technologies: For the integration of fluctuating renewable energy. Technical report, 2010.

[12] JM Bermúdez, E Ruisánchez, A Arenillas, AH Moreno, and JA Menéndez. New concept for energy storage: Microwave-induced carbon gasification with co2. *Energy Conversion and Management*, 78:559–564, 2014.

[13] Tao Ma, Hongxing Yang, and Lin Lu. Feasibility study and economic analysis of pumped hydro storage and battery storage for a renewable energy powered island. *Energy Conversion and Management*, 79(0):387 – 397, 2014.

[14] S. Galli, A. Scaglione, and Zhifang Wang. For the grid and through the grid: The role of power line communications in the smart grid. *Proceedings of the IEEE*, 99(6):998–1027, 2011.

[15] Zhifang Wang, A. Scaglione, and R.J. Thomas. Generating statistically correct random topologies for testing smart grid communication and control networks. *IEEE Transactions on Smart Grid*, 1(1):28–39, 2010.

[16] Pacific Crest Mosaic. Mesh networks is communications winner in utility survey. 2009.

[17] J. Lavaei and S.H. Low. Relationship between power loss and network topology in power systems. In *2010 49th IEEE Conference on Decision and Control (CDC)*, pages 4004–4011, 2010.

[18] Norihiko Shinomiya, Toshio Koide, and Hitoshi Watanabe. A theory of tie-set graph and its application to information network management. *International Journal of Circuit Theory and Applications*, 29(4):367–379, 2001.

[19] L.F. Bic. Distributed computing using autonomous objects. In *Proceedings of the Fifth IEEE Computer Society Workshop on Future Trends of Distributed Computing Systems*, pages 160–168, 1995.

[20] Toshio Koide, Haruki Kubo, and Hitoshi Watanabe. A study on the tie-set graph theory and network flow optimization problems. *International Journal of Circuit Theory and Applications*, 32(6):447–470, 2004.

[21] J. Malinowski. A new efficient algorithm for generating all minimal tie-sets connecting selected nodes in a mesh-structured network. *IEEE Transactions on Reliability*, 59(1):203–211, 2010.

[22] Studiosus Kirchhoff. Ueber den durchgang eines elektrischen stromes durch eine ebene, insbesondere durch eine kreisfrmige. *Annalen der Physik*, 140(4):497–514, 1845.

[23] Hitoshi Watanabe and Shoji Shinoda. Soul of circuit theory. a review on research activities of graphs and circuits in japan. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 46(1):83–94, 1999.

[24] Hitoshi Watanabe. A basic theory of information network. *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, 76(3):265–276, 1993.

[25] Tatsuo Ohtsuki, Yasutoshi Ishizaki, and Hitoshi Watanabe. Network analysis and topological degrees of freedom. *ELECTRONICS & COMMUNICATIONS IN JAPAN*, 51(6):33, 1968.

[26] Ji-Yuan Fan, Lan Zhang, and J.D. McDonald. Distribution network reconfiguration: single loop optimization. *IEEE Transactions on Power Systems*, 11(3):1643–1647, Aug 1996.

[27] Yoichi Sakai, Kiyoshi Nakayama, and Norihiko Shinomiya. A node-weight equalization problem with circuit-based computations. In *2013 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 2525–2528, 2013.

[28] Yoichi Sakai, Kiyoshi Nakayama, and Norihiko Shinomiya. A property verification of node-weight equalization focusing on cycles of a graph. In *Proc. of 27th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, pages P–T2–236, 2012.

[29] Kiyoshi Nakayama, Norihiko Shinomiya, and Hitoshi Watanabe. An autonomous distributed control method based on tie-set graph theory in future grid. *International Journal of Circuit Theory and Applications*, 41(11):1154 – 1174, 2013.

[30] K. Nakayama and N. Shinomiya. Distributed control based on tie-set graph theory for smart grid networks. In *2010 International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, pages 957–964, Oct 2010.

[31] K. Nakayama, K.E. Benson, L.F. Bic, and M.B. Dillencourt. Complete automation of future grid for optimal real-time distribution of renewables. In *2012 IEEE Third International Conference on Smart Grid Communications (SmartGridComm)*, pages 418–423, 2012.

[32] K. Nakayama, C. Zhao, L.F. Bic, M.B. Dillencourt, and J. Brouwer. Distributed real-time power flow control with renewable integration. In *IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 516–521, 2013.

[33] K. Nakayama, N. Shinomiya, and H. Watanabe. An autonomous distributed control method for link failure based on tie-set graph theory. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 59(11):2727–2737, 2012.

[34] Kiyoshi Nakayama, Kyle E. Benson, Vahe Avagyan, Michael B. Dillencourt, Lubomir F. Bic, and Nalini Venkatasubramanian. Tie-set based fault tolerance for autonomous recovery of double-link failures. In *IEEE Symposium on Computers and Communications (ISCC)*, pages 391–397, July 2013.

[35] K. Nakayama and N. Shinomiya. Autonomous recovery for link failure based on tie-sets in information networks. In *2011 IEEE Symposium on Computers and Communications (ISCC)*, pages 671–676, June 2011.

[36] K. Nakayama, N. Shinomiya, and H. Watanabe. Distributed control for link failure based on tie-sets in information networks. In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 3913–3916, May 2010.

[37] K. Kadena and N. Shinomiya. Distributed network failure recovery with tie-sets. In *2012 44th Southeastern Symposium on System Theory (SSST)*, pages 190–194, March 2012.

[38] J. Nagano and N. Shinomiya. A failure recovery method based on cycle structure and its verification by openflow. In *Advanced Information Networking and Applications (AINA), 2013 IEEE 27th International Conference on*, pages 298–303, March 2013.

[39] Ruofei Ma, Hsiao-Hwa Chen, Yu-Ren Huang, and Weixiao Meng. Smart grid communication: Its challenges and opportunities. *IEEE Transactions on Smart Grid*, 4(1):36–46, 2013.

[40] M. Iri, I. Shirakawa, Y. Kajitani, and S. Shinoda. *Graph Theory with Exercises*. CORONA Pub: Japan, 1983.

[41] N Shinomiya, T Koide, M Kuroha, and H Watanabe. Tie-set graph, meta-tie-set graph and information network. *IEICE/IEEK Proceedings of ITC-CSCC99*, 2:872–875, 1999.

[42] Nancy A. Lynch. *Distributed Algorithms*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1996.

[43] Ram Rajagopal, Eilyan Bitar, Pravin Varaiya, and Felix Wu. Risk-limiting dispatch for integrating renewable power. *International Journal of Electrical Power and Energy Systems*, 44(1):615 – 628, 2013.

[44] Brian Tarroja, Fabian Mueller, Joshua D. Eichman, Jack Brouwer, and Scott Samuelsen. Spatial and temporal analysis of electric wind generation intermittency and dynamics. *Renewable Energy*, 36(12):3424 – 3432, 2011.

[45] Irvine smart grid demonstrate project. Technical report, EDISON and EPRI, November 2010.

[46] Van Mieghem. *Graph spectra for complex networks*. Cambridge University Press, 2011.

[47] Peng Xu. Demand shifting with thermal mass in large commercial buildings in a california hot climate zone. 2010.

[48] James E. Braun and Ray W. Herrick Laboratories. Load control using building thermal mass.

[49] K. Clement-Nyns, E. Haesen, and J. Driesen. The impact of charging plug-in hybrid electric vehicles on a residential distribution grid. *IEEE Transactions on Power Systems*, 25(1):371–380, Feb 2010.

[50] D. O'Neill, M. Levorato, A. Goldsmith, and U. Mitra. Residential demand response using reinforcement learning. In *2010 First IEEE International Conference on Smart Grid Communications (SmartGridComm)*, pages 409–414, Oct 2010.

[51] S. Grillo, M. Marinelli, S. Massucco, and F. Silvestro. Optimal management strategy of a battery-based storage system to improve renewable energy integration in distribution networks. *IEEE Transactions on Smart Grid*, 3(2):950–958, June 2012.

[52] Dimitri P Bertsekas, Dimitri P Bertsekas, Dimitri P Bertsekas, and Dimitri P Bertsekas. *Dynamic programming and optimal control*, volume 1. Athena Scientific Belmont, MA, 1995.

[53] J. Carpentier. Contribution to the economic dispatch problem. *Bulletin de la Societe Francoise des Electriciens*, 3(8):431–447, 1962.

[54] B.H. Chowdhury and S. Rahman. A review of recent advances in economic dispatch. *IEEE Transactions on Power Systems*, 5(4):1248–1259, 1990.

[55] M. Huneault and F.D. Galiana. A survey of the optimal power flow literature. *IEEE Transactions on Power Systems*, 6(2):762–770, 1991.

[56] J.A. Momoh, M.E. El-Hawary, and R. Adapa. A review of selected optimal power flow literature to 1993. ii. newton, linear programming and interior point methods. *IEEE Transactions on Power Systems*, 14(1):105–111, 1999.

[57] James A Momoh. *Electric power system applications of optimization*, volume 11. CRC Press, 2001.

[58] Yan Xia and Ka Wing Chan. Dynamic constrained optimal power flow using semi-infinite programming. *IEEE Transactions on Power Systems*, 21(3):1455–1457, 2006.

[59] J. Jardim and B. Stott. Synthetic dynamics power flow. In *Power Engineering Society General Meeting, 2005. IEEE*, pages 479–484 Vol. 1, 2005.

[60] Jiaqi Liang, D.D. Molina, G.K. Venayagamoorthy, and R.G. Harley. Two-level dynamic stochastic optimal power flow control for power systems with intermittent renewable generation. *IEEE Transactions on Power Systems*, 28(3):2670–2678, 2013.

[61] P.F. Ribeiro, B.K. Johnson, M.L. Crow, A. Arsoy, and Y. Liu. Energy storage systems for advanced power applications. *Proceedings of the IEEE*, 89(12):1744–1756, 2001.

[62] Whei-Min Lin, Cong-Hui Huang, and Tung-Sheng Zhan. A hybrid current-power optimal power flow technique. *IEEE Transactions on Power Systems*, 23(1):177–185, 2008.

[63] K.M. Chandy, S.H. Low, Ufuk Topcu, and Huan Xu. A simple optimal power flow model with energy storage. In *2010 49th IEEE Conference on Decision and Control (CDC)*, pages 1051–1057, 2010.

[64] R. Saino, T. Takeshita, N. Izuhara, and F. Ueda. Power flow estimation and line-loss-minimization control using upfc in loop distribution system. In *IEEE 6th International on Power Electronics and Motion Control Conference (IPEMC)*, pages 2426–2431, 2009.

[65] William D Stevenson. *Elements of power system analysis-4/E.* McGraw-Hill Kogakusha, Ltd., 1982.

[66] Arthur R Bergen. *Power Systems Analysis, 2/E.* Pearson Education India, 2009.

[67] Q.Y. Jiang, N.D. Chiang, C.X. Guo, and Y.J. Cao. Power-current hybrid rectangular formulation for interior-point optimal power flow. *Generation, Transmission Distribution, IET*, 3(8):748–756, 2009.

[68] Quanyuan Jiang, Guangchao Geng, Chuangxin Guo, and Yijia Cao. An efficient implementation of automatic differentiation in interior point optimal power flow. *IEEE Transactions on Power Systems*, 25(1):147–155, 2010.

[69] R.A. Jabr. Radial distribution load flow using conic programming. *IEEE Transactions on Power Systems*, 21(3):1458–1459, 2006.

[70] R.D. Christie, B.F. Wollenberg, and I. Wangensteen. Transmission management in the deregulated environment. *Proceedings of the IEEE*, 88(2):170–195, 2000.

[71] J.A. Aguado and V.H. Quintana. Inter-utilities power-exchange coordination: a market-oriented approach. *IEEE Transactions on Power Systems*, 16(3):513–519, 2001.

[72] A.J. Conejo and J.A. Aguado. Multi-area coordinated decentralized dc optimal power flow. *IEEE Transactions on Power Systems*, 13(4):1272–1278, 1998.

[73] R.A. Jabr. Optimal power flow using an extended conic quadratic formulation. *IEEE Transactions on Power Systems*, 23(3):1000–1008, 2008.

[74] J. Lavaei, A. Babakhani, A. Hajimiri, and J.C. Doyle. Solving large-scale linear circuit problems via convex optimization. In *Proceedings of IEEE Conference on Decision and Control*, pages 4977–4984, 2009.

[75] Chee Wei Tan, D.W.H. Cai, and Xin Lou. Dc optimal power flow: Uniqueness and algorithms. In *2012 IEEE Third International Conference on Smart Grid Communications (SmartGridComm)*, pages 641–646, 2012.