Solving Customer-Driven Microgrid Optimization Problems as DCOPs

Saurabh Gupta†, Palak Jain‡, William Yeoh†, Satish J. Ranade‡, and Enrico Pontelli†

†Department of Computer Science
New Mexico State University
Las Cruces, NM 88003
{saurabh,wyeoh,epontell}@cs.nmsu.edu

‡Klipsch School of Electrical and Computer Engineering
New Mexico State University
Las Cruces, NM 88003
{palak,sranade}@nmsu.edu

Abstract. In response to the challenge by Ramchurn *et al.* to solve smart grid optimization problems with artificial intelligence techniques [21, 23], we investigate the feasibility of solving two common smart grid optimization problems as distributed constraint optimization problems (DCOPs). Specifically, we look at two common customer-driven microgrid (CDMG) optimization problems – a comprehensive CDMG optimization problem and an islanding problem. We show how one can model both problems as DCOPs and solve them using off-the-shelf DCOP algorithms, thus showing that researchers in the distributed constraint reasoning community are in a unique position to contribute towards this challenge.

1 Introduction

There is a growing consensus that there is an urgent need to move away from fossil fuel to renewable energy resources given that the demand for fossil fuel will soon outstrip supply and the carbon emissions from burning fossil fuel will have long lasting impact on global warming [21]. In order to meet the demand of power, a large number of renewable power generators, which may likely be co-located with power consumers, will need to be incorporated into the power grid. These renewable generators (e.g., solar panels, wind farms) may be located at the same locations as the power consumers (e.g., homes, factories, office buildings, etc.). Thus, unlike the power grids of today, where we have limited control and communication capabilities, the power grids of the future will need new mechanisms to control and coordinate the power generation as well as consumption.

One vision of this future power grid is a *smart grid*, which the US Department of Energy defined as: "*A fully automated power delivery network that monitors and controls every customer and node, ensuring a two-way flow of electricity* *and information between the power plant and the appliance, and all points in between*" [2].

While there is a large body of work by researchers in power systems towards realizing this vision [11, 12, 10, 1, 5, 17, 7, 9, 8, 3, 20], there has been very little inter-disciplinary interactions with researchers in the artificial intelligence (AI) community. Thus, Ramchurn *et al.* have recently challenged the AI community to work towards realizing this future as well and develop new algorithms and mechanisms that can solve problems involving large numbers of highly heterogeneous agents (e.g., consumers with different demand profiles and generators with different capabilities), each with their own aims and objectives, and having to operate under uncertainty [21, 23].

Researchers in distributed constraint reasoning [24] are in a unique position to answer this challenge as a number of optimization problems in smart grids can be modeled as distributed constraint satisfaction problems (DCSPs) and distributed constraint optimization problems (DCOPs). Furthermore, the decentralized formulation is desirable because this problem is inherently distributed; each agent (e.g., home owners, factory owners, and power plant managers) controls a set of local variables (e.g., the amount of power consumed in an agent's home) and is constrained to other agents in the problem. In this paper, we present a way to use DCOPs to model two such optimization problems – (1) the problem of optimizing power generation, consumption and delivery in a power network, and (2) the problem of identifying if there is a need to island a group of agents (i.e., a need to disconnect the group of agents from the overall power grid) under unexpected circumstances (e.g., shut down of power plants). We focus our application domain on *customer-driven microgrids* (CDMGs), which can be viewed as one possible instantiation of the smart grid, whereby control of power production, consumption and delivery is distributed amongst agents in the system [22].

The structure of this paper is as follows. We first provide some background on CDMGs, describe the two optimization problems mentioned above, and provide a brief overview on DCOPs in Section 2. We then show how we can use DCOPs to model both optimization problems in Section 3 and discuss some related work in Section 4. Finally, we show some empirical evaluations of our models in Section 5 and conclude in Section 6.

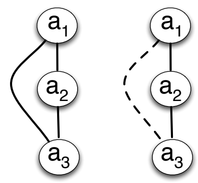
2 Background

We now provide some background on DCOPs and CDMGs, and describe the two optimization problems mentioned above.

2.1 Distributed Constraint Optimization Problems

A *distributed constraint optimization problem* (DCOP) [18, 19, 25, 4, 24] is defined by a tuple $\langle \mathcal{A}, \mathcal{X}, \alpha, \mathcal{D}, \mathcal{F} \rangle$, where:

- $\mathcal{A} = \{a_1, \ldots, a_p\}$ is a set of agents.



for $i < j$

a_i	a_j	Costs
0	0	5
0	1	8
1	0	20
1	1	3

(a) (b) (c)

Fig. 1: Example DCOP

- $\mathcal{X} = \{x_1, \ldots, x_m\}$ is a set of variables.
- $\alpha : \mathcal{X} \to \mathcal{A}$ maps each variable to an agent.
- $\mathcal{D} = \{D_1, \ldots, D_m\}$ is a set of finite domains, where D_i is the domain assigned to the variable x_i.
- $\mathcal{F} = \{F_1, \ldots, F_k\}$ is a set of n-ary constraints, where each constraint $F_i : D_{i_1} \times \ldots \times D_{i_n} \mapsto \mathbb{N}^+ \cup \{0, \infty\}$ specifies the cost of each combination of values of variables x_{i_1} to x_{i_n}.

A *solution* is a value assignment for a subset of variables. The cost of the solution is the evaluation of all constraints on that solution. A solution is *complete* iff it is a value assignment for all variables. The goal is to find a cost-minimal complete solution X^*:

$$X^* = \arg\min_X \sum_{F_i \in \mathcal{F}} F_i(x_{i_1}, \ldots, x_{i_n} \mid x_{i_1}, \ldots, x_{i_n} \in X) \tag{1}$$

A *constraint graph* visualizes a DCOP instance, where nodes in the graph correspond to variables in the DCOP and edges connect pairs of variables appearing in the same constraint. A *DFS pseudo-tree* arrangement has the same nodes and edges as the constraint graph, and: **(i)** there is a subset of edges, called *tree edges*, that form a rooted tree, and **(ii)** two variables in a constraint appear in the same branch of that tree. The other edges are called *backedges*. Tree edges connect parent-child nodes, while backedges connect a node with its pseudo-parents and its pseudo-children. A DFS pseudo-tree arrangement can be constructed using distributed DFS algorithms [6].

Figure 1(a) shows the constraint graph of a sample DCOP, with three agents controlling variables with domain $\{0, 1\}$. Figure 1(b) shows one possible pseudo-tree (the dotted line is a *backedge*), and the third column in Figure 1(c) shows the utilities of the three constraints.

2.2 Customer-Driven Microgrids

A *customer-driven microgrid* (CDMG) can be viewed as one possible instantiation of the smart grid, whereby control of power production, consumption and delivery is distributed amongst agents in the system [22]. Figure 2 shows an example CDMG with six homes and a coal power plant. Each home has its individual power demands (e.g., washing machine, dryer, air-conditioner, heater,

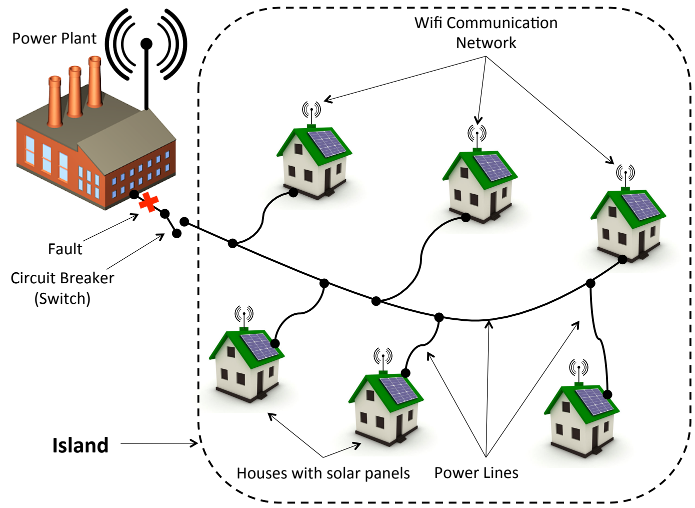


Fig. 2: Illustration of a Customer-Driven Microgrid

etc.) and generation capacities (e.g., solar panels). The homes and power plant are connected to each other via a communication network (e.g., wireless network), which allows them to communicate with each other, as well as via a power network (i.e., network of power lines), which enables them to transfer power to one another. Each power line has a power carrying capacity due to physical constraints such as thermal limits that prevent them from carrying more power. In normal operations, this CDMG stays connected with the power plant, but in the case of a fault between the homes and the power plant, the homes can isolate themselves by opening a circuit breaker (switch) and *island* themselves as a new CDMG. This islanded CDMG now needs to solve various power management and control problems, preferably in a distributed manner, because a central controller can be lost due to a fault in this scenario.

2.3 Comprehensive CDMG Optimization Problem

We now describe a comprehensive CDMG optimization problem that has been shown to subsume several classical power systems subproblems such as load shedding, demand response, and restoration [10]. At a high level, the optimization problem frames the question: "*How should a CDMG be operated such that collective cost is minimized for all users in the system subject to domain-specific constraints*?" Figure 3 shows a mathematical formulation of the problem, where

- n is the number of nodes (e.g., power plant, homes) in the network;
- Pg_i is the power generation at node i;
- Pl_i is the power consumption at node i;
- Pt_{ij} is the transmission line flow between nodes i and j;
- $\widehat{Pg_i}$ is the maximum generation capacity at node i;

$$
\begin{aligned}
\textbf{Minimize} \quad & \sum_{i=1}^{n} \Bigg[\overbrace{e^{-A_i Pl_i} \left(Pl_i - \widehat{Pl_i}\right)^2}^{\text{consumption preference}} + \overbrace{\left(1 - e^{-B_i Pg_i}\right)\left(Pg_i - \widehat{Pg_i}\right)^2}^{\text{generation preference}} \\
& \qquad + \underbrace{\left(\alpha_i + \beta_i Pg_i + \gamma_i Pg_i^2\right)}_{\text{cost function}} \Bigg] \qquad (2)
\end{aligned}
$$

subject to the following constraints :

$$Pg_i - Pl_i - \sum_j Pt_{ij} = 0 \qquad \forall i \quad (3)$$

$$Pt_{ij} + Pt_{ji} = 0 \qquad \forall i, j \quad (4)$$

$$Pg_i \in \{0, \ldots, \widehat{Pg_i}\} \qquad \forall i \quad (5)$$

$$Pl_i \in \{0, \widehat{pl_i}, \ldots, \widehat{Pl_i}\} \qquad \forall i \quad (6)$$

$$Pt_{ij} \in \{-\widehat{Pt_{ij}}, \ldots, \widehat{Pt_{ij}}\} \qquad \forall i \quad (7)$$

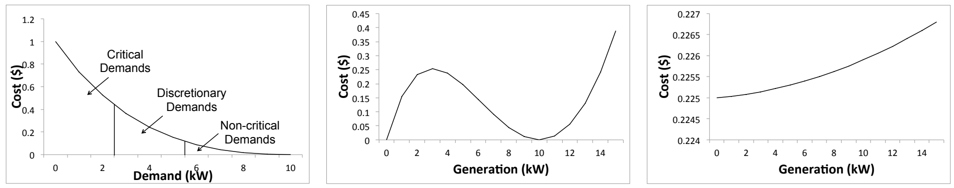
Fig. 3: Mathematical Program for the Comprehensive CDMG Optimization Problem

- $\widehat{Pl_i}$ and $\widehat{pl_i}$ is the maximum and minimum power consumption at node i, respectively;
- $\widehat{Pt_{ij}}$ is the maximum transmission line capacity between nodes i and j;
- α_i, β_i, and γ_i are cost coefficients; and
- A_i and B_i are weights.

The objective function in Equation (2) is the summation of three parts: the agents' consumption preference, the agents' generation preference, and a global cost function. Figure 4a shows a representative example of the first component of the objective function – the agent's consumption preference. The graph plots the cost of power (in dollars) against the amount of power consumed (in kWs). As the cost of power increases, the demand for power will also decrease. If the cost of power is at its highest, only a small amount of power will be consumed to satisfy only critical demands. If the cost of power is slightly cheaper, then more power will be consumed to also satisfy discretionary demands. Finally, if the cost of power is at its lowest, then more power will be consumed to also satisfy non-critical demands.

Figure 4b shows a representative example of the second component of the objective function – the agent's generation preference. The graph plots the cost of power (in dollars) against the amount of power generated (in kWs). When the generator is switched off (generation is at 0 kW) or operating at peak capacity (generation is at 10 kW), then the cost is minimal. If the generator operates below peak capacity, then the cost peaks in between the peak and minimal capacity. If the generator operates above peak capacity, then the cost grows exponentially.

Figure 4c shows a representative example of the third component of the objective function – the global cost function. It grows at a polynomial rate with



(a) (b) (c)

Fig. 4: Comprehensive CDMG Optimization Objective Subfunctions

respect to the overall generation, which is typically used by electric utilities to model the operating cost of its generators. This cost includes the cost of fuel, cost to maintain the generators and the power network, etc. These three example graphs can vary depending on the preferences and needs of agents as well as types of power generators.

The optimization problem is subject to the following constraints. The constraint in Equation (3) represents the power balance principle at each node i, where the access power (amount of power generated minus the amount of power consumed) must be transferred out of the node to its neighboring nodes. The constraint in Equation (4) ensures that there is no power loss in the system; the power flowing from node i to node j equals the power received by node j from node i. The constraints in Equations (5) and (6) represent the generation and consumption limits at node i, and the constraint in Equation (7) represents the maximum current carrying capacity of the power line between nodes i and j.

2.4 Islanding Problem

We now describe another important problem in microgrids, namely the *islanding* problem. Consider the power network shown in Figure 2. If the main power supply to a neighborhood is interrupted due to a fault, then this neighborhood will have to rely on its own power generation. This neighborhood will isolate itself from the rest of the network by opening a switch. This isolated neighborhood is called an island. Now, if the total maximum power generation capacity of this neighborhood is not enough to satisfy its minimum power demands, then it will result in a black out. To avoid such failures, we modeled switches at both ends of each power line, which can be opened and closed according to the situation, to facilitate the formation of islands in the neighborhood. A viable island is one that has enough power generation capacity to satisfy its own minimum power demands. The islanding problem frames the following question: "*Which circuit breaker to break such that the impact is smallest*?" For example, if impact is measured by the amount of collective unsatisfied power consumption, then Figure 5 shows a mathematical formulation of the problem, where S_{ij} is the switch between nodes i and j, and all other variables are the same as that in Figure 3.

The objective function in Equation (8) is to minimize the difference between the largest amount of possible power consumed or, alternatively, the maximum

$$
\begin{aligned}
&\textbf{Minimize} \quad \sum_n (\widehat{Pl_i} - Pl_i) && && (8)\\
&\textbf{subject to the following constraints}:\\
&Pg_i - Pl_i - \sum_j Pt_{ij} = 0 && \forall i && (9)\\
&Pt_{ij} + Pt_{ji} = 0 && \forall i,j && (10)\\
&Pg_i \in \{0, \ldots, \widehat{Pg_i}\} && \forall i && (11)\\
&Pl_i \in \{0, \widehat{pl_i}, \ldots, \widehat{Pl_i}\} && \forall i && (12)\\
&Pt_{ij} \in \{-\widehat{Pt_{ij}}, \ldots, \widehat{Pt_{ij}}\} && \forall i && (13)\\
&-S_{ij} \times \widehat{Pt_{ij}} \leq Pt_{ij} \leq S_{ij} \times \widehat{Pt_{ij}} && \forall i,j && (14)\\
&S_{ij} \leq |Pt_{ij}| && \forall i,j && (15)\\
&S_{ij} \in \{0,1\} && \forall i,j && (16)
\end{aligned}
$$

Fig. 5: Mathematical Program for the Islanding Problem

power demand in the network and the largest amount of possible power produced or, alternatively, the maximum production capacity in the network. The optimization problem is subject to the same constraints as that in Figure 3 except for the constraints in Equations (14), (15), and (16). Equation (16) restricts the possible values of S_{ij} to 0 and 1, where a value of 0 indicates that the switch between nodes i and j is open or, alternatively, the transmission line between the two nodes is severed, and a value of 1 indicates otherwise. The constraint in Equation (14) ensures that power is flowing between nodes i and j only if the switch between the two nodes is closed, and the constrain in Equation (15) ensures that a switch is open if no power is flowing through it.

We have actually solved a more detailed version of this problem in a distributed manner using a distributed version of the Gauss-Seidel method [12]. However, this method does not have any convergence guarantees and usually takes a long time to converge in practice. Therefore, in this paper, we introduce the simpler version described above as another example smart grid optimization problem that can be modeled as a DCOP.

3 CDMG Problems as DCOPs

We now show how one can model the CDMG problems described in Sections 2.3 and 2.4 as a DCOP and use existing DCOP algorithms to solve it. We will illustrate this model using a 4-node network as an example. However, the techniques proposed here is easily generalizable to a real network. Figure 6 shows our example, where each node is assumed to have an autonomous agent that is capable of controlling the power generation and consumption at its node.

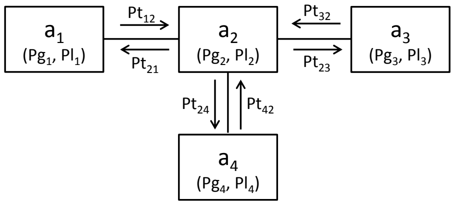


Fig. 6: Example 4-Node Network for the Comprehensive CDMG Optimization Problem

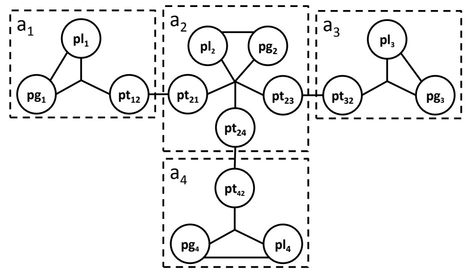


Fig. 7: Constraint Graph for the Network in Figure 6

3.1 Modeling the Comprehensive CDMG Optimization Problem

Using the definition of DCOPs described in Section 2.1, we will now define the tuple $\langle \mathcal{A}, \mathcal{X}, \alpha, \mathcal{D}, \mathcal{F} \rangle$ for our example:

- $\mathcal{A}$ consists of four agents $\{a_1, a_2, a_3, a_4\}$, one for each node of the network.
- $\mathcal{X}$ consists of variables Pg_i, Pl_i and Pt_{ij} for each agent a_i and its neighboring agent a_j.
- α maps each variable Pg_i, Pl_i and Pt_{ij} to agent a_i.
- $\mathcal{D}$ consists of the domains D_{Pg_i}, D_{Pl_i} and $D_{Pt_{ij}}$ for each agent a_i and its neighboring agent a_j, where:
  - $D_{Pg_i} = \{0, \ldots, \widehat{Pg_i}\}$ is the domain of variable Pg_i;
  - $D_{Pl_i} = \{\widehat{pl_i}, \ldots, \widehat{Pl_i}\}$ is the domain of variable Pl_i;
  - $D_{Pt_{ij}} = \{-\widehat{Pt_{ij}}, \ldots, \widehat{Pt_{ij}}\}$ is the domain of variable Pt_{ij}.
- $\mathcal{F}$ consists of the following n-ary constraints for each agent a_i:
  - Hard constraint: $Pg_i - Pl_i - \sum_{a_j \in N_i} Pt_{ij} = 0$, where N_i is the set of neighboring agents of agent a_i.
  - Hard constraint: $Pt_{ij} + Pt_{ji} = 0$ for each neighboring agent a_j.
  - Soft constraint: $e^{-A_i Pl_i}(Pl_i - \widehat{Pl_i})^2 + (1 - e^{-B_i Pg_i})(Pg_i - \widehat{Pg_i})^2 + \alpha_i + \beta_i Pg_i + \gamma_i Pg_i^2$.

  The cost of a hard constraint is 0 if the constraint is satisfied and ∞ otherwise. The cost of a soft constraint is exactly the evaluation of the constraint.

The solution to such DCOP maps to a solution of the CDMG optimization problem because of the following:

- The objective function of the CDMG optimization problem in Equation (2) can be decomposed into the sum of individual functions for each agent, and each such function is represented as a soft constraint. A DCOP solution minimizes the sum of constraint costs, and thus minimizes the objective function as well.
- The constraints in Equations (3) and (4) are satisfied, because the DCOP solution will have a cost of ∞ otherwise, due to the hard constraints.

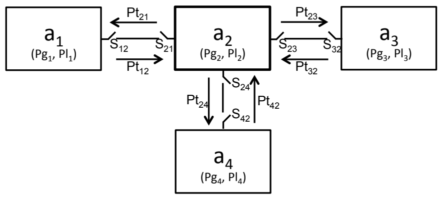


Fig. 8: Example 4-Node Network for the Islanding Problem

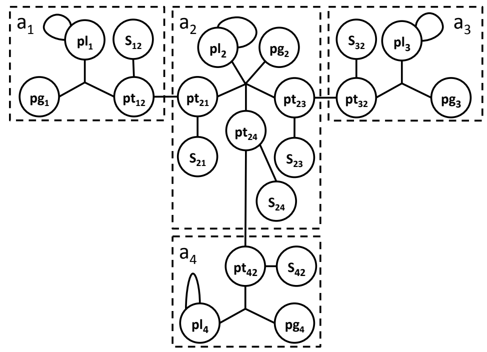


Fig. 9: Constraint Graph for the Network in Figure 8

– The constraints in Equations (5), (6) and (7) are satisfied, because the possible values of the variables Pg_i, Pt_i and Pt_{ij} are bounded from above by $\widehat{Pg_i}$, $\widehat{Pl_i}$ and $\widehat{Pt_{ij}}$, respectively, and from below by 0, $\widehat{pl_i}$, $-\widehat{Pt_{ij}}$, respectively.

Given this model, the constraint graph for our example network is shown in Figure 7, where nodes correspond to variables and (hyper-)edges correspond to constraints. We also group the variables that are owned by the same agent in a dotted box and the agent owning those variables are shown in the top-left corner of each box.

3.2 Modeling the Islanding Problem

For the islanding problem, our example network will include two new variables S_{ij} and S_{ji} for each transmission line between nodes i and j. Figure 8 shows the example. The DCOP $\langle \mathcal{A}, \mathcal{X}, \alpha, \mathcal{D}, \mathcal{F} \rangle$ is exactly the same as that described in Section 3, except for the following:

– $\mathcal{X}$ also consists of variables S_{ij} for each agent a_i and its neighboring agent a_j.
– α also maps each variable S_{ij} to agent a_i.
– $\mathcal{D}$ also consists of the domains $D_{S_{ij}} = \{0, 1\}$ for each agent a_i and its neighboring agent a_j.
– $\mathcal{F}$ also consists of the following n-ary constraints for each agent a_i:
– Hard constraint: $-S_{ij} \times \widehat{Pt_{ij}} \leq Pt_{ij} \leq S_{ij} \times \widehat{Pt_{ij}}$ for each neighboring agent a_j. The cost of this constraint is 0 if the constraint is satisfied and ∞ otherwise.

The solution to such DCOP maps to a solution of the islanding problem for similar reason as described in Section 3. Given this model, the constraint graph for our example network is shown in Figure 9.

4 Related Work

Researchers have used DCOPs to model some smart grid optimization problems and used DCOP algorithms to solve them with some success. For example, Miller *et al.* have used max-sum to solve the problem of optimizing power generation and transmission such that carbon dioxide emissions from power generators such as coal power plants is minimized, assuming the power consumption of each agent that needs to be satisfied is given as an input [16]; and Kumar *et al.* have used DPOP to solve the problem of reconfiguring power networks to restore power to nodes when power lines fail [13].

The two problems above are similar to ours in that all the problems model constraints that are derived from the power network (e.g., flow conservation constraints). However, there are two main differences between their work and ours: (1) Their objective functions are linear while ours can be non-linear with multiple local optima, and (2) they assume that the consumption of agents are given as an input while we allow them to be optimized as part of the problem. As a result of the second difference, our constraint graph can have multiple cycles unlike previous constraint graphs [16].

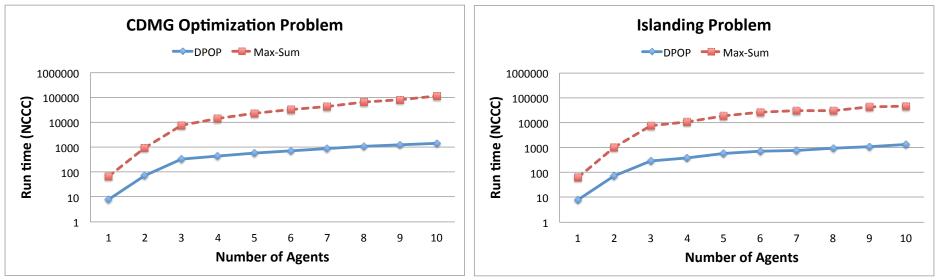
5 Experimental Results

We now describe some of our experimental results on synthetic problems that are motivated by real-world power networks for both the comprehensive CDMG optimization problem and islanding problem. We choose DPOP [19], an optimal DCOP algorithm, and max-sum [4], a sub-optimal DCOP algorithm, to show that these problems can be solved using off-the-shelf DCOP algorithms. We use publicly-available implementations of both algorithms [14]. We measured the performance of the algorithms in terms of quality of solutions found, runtime measured in non-concurrent constraint checks (NCCCs) [15], and network load measured in the total amount of information exchanged. To test the scalability of DPOP and max-sum, we varied the problem size and complexity along three dimensions: (*i*) number of agents, (*ii*) size of the domains, and (*iii*) complexity of topology configurations. We average each data point over 10 runs.

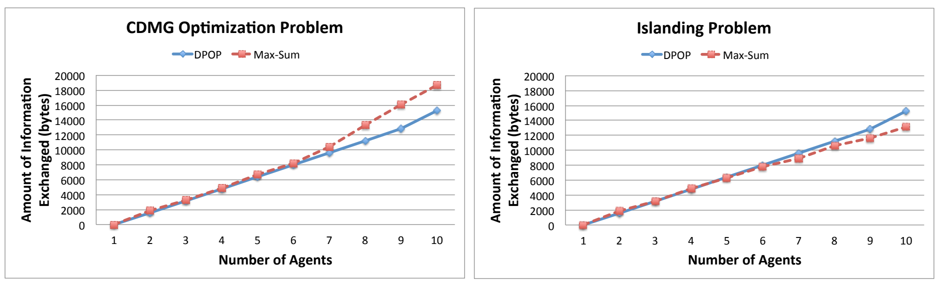
5.1 Varying Number of Agents

For this set of experiments, we fix the parameters to the following values: $\widehat{Pg_i} = i$, $\widehat{Pl_i} = 11 - i$, $\widehat{Pt_{ij}} = 2$, $\alpha_i = 10$, $\beta_i = 0.01$, $\gamma_i = 0.001$, $A_i = 1/\widehat{Pl_i}$, $B_i = 1/\widehat{Pg_i}$ for all nodes i and j. We vary the number of agents from 1 to 10 and arrange them in a chain (see Configuration 1 in Figure 12).

Figure 10a shows the runtime of both algorithms. As expected, the runtime of both algorithms increases with the number of agents. However, DPOP is faster than max-sum. The reason is the following: the computational cost of some messages in both algorithms (UTIL messages in DPOP and function-to-variable messages in max-sum) is exponential in the arity of the constraints [19,



(a)



(b)

Fig. 10: Experiments Varying the Number of Agents

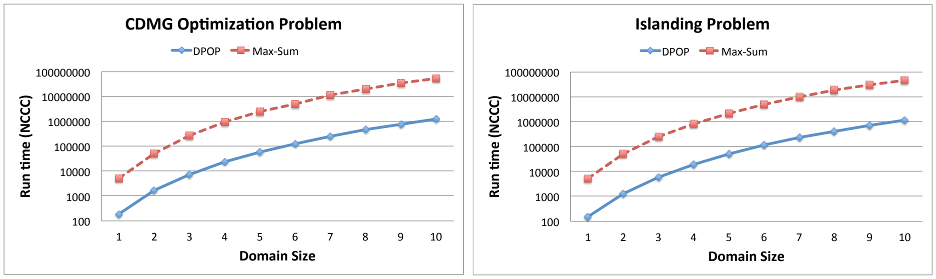
4]. However, the number of UTIL messages that DPOP constructs is linear in the number of agents in the problem, and the number of function-to-variable messages that max-sum constructs depends on the number of iterations it takes to converge, which is typically larger than the number of agents in the problem.

Figure 10b shows the network load, where both algorithms exchange a comparable amount of information in total. In terms of solution quality, DPOP found optimal solutions for all problem instances and max-sum found optimal solutions for 40% of the comprehensive CDMG optimization problem instances and 30% of the islanding problem instances. Max-sum found infeasible solutions for the other problem instances.

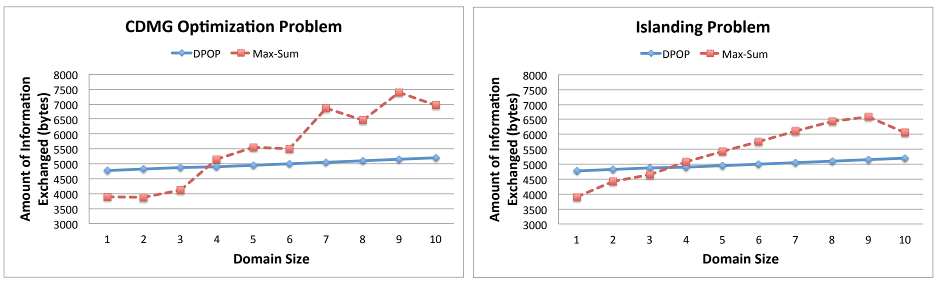
5.2 Varying Domain Sizes

For this set of experiments, we fix the number of agents to 4 and arrange them in a "T" (see Configuration 2 in Figure 12 with agents A1, A2, A3, and A7). We fix the domain sizes to the following values: $\widehat{Pg_i} = k$, $\widehat{Pl_i} = k$, and $\widehat{Pt_{ij}} = k$, where we vary k from 1 to 10. We fix the other parameters to the same values as in the first set of experiments.

Figure 11 shows our results, where the trends are similar to those in Figure 10. In terms of solution quality, DPOP found optimal solutions for all problem instances and max-sum found optimal solutions for 10% of the comprehensive



(a)



(b)

Fig. 11: Experiments Varying the Domain Size

CDMG optimization problem instances and 100% of the islanding problem instances. Max-sum found infeasible solutions for the other problem instances.

5.3 Varying Complexity of Topology Configurations

For this set of experiments, we fix the number of agents to 7. We fix the domain sizes to the following values: $\widehat{Pg_i} = 4$, $\widehat{Pl_i} = 4$, and $\widehat{Pt_{ij}} = 4$, and we fix the other parameters to the same values as in the first set of experiments. We vary the topology of the network in 5 configurations as illustrated in Figure 12. The different configurations show a different levels of problem complexity in terms of different constraint arity.

Figure 13 shows our results, where the trends are similar to those in Figure 10. In terms of solution quality, DPOP found optimal solutions for all problem instances and max-sum found optimal solutions for 0% of the comprehensive CDMG optimization problem instances and 30% of the islanding problem instances. Max-sum found infeasible solutions for the other problem instances.

Overall, these sets of experiments show that it is feasible to map CDMG optimization problems and islanding problems as DCOPs and solve them using existing DCOP algorithms. However, some of our trends in our results require more analysis and a deeper understanding for their behavior.

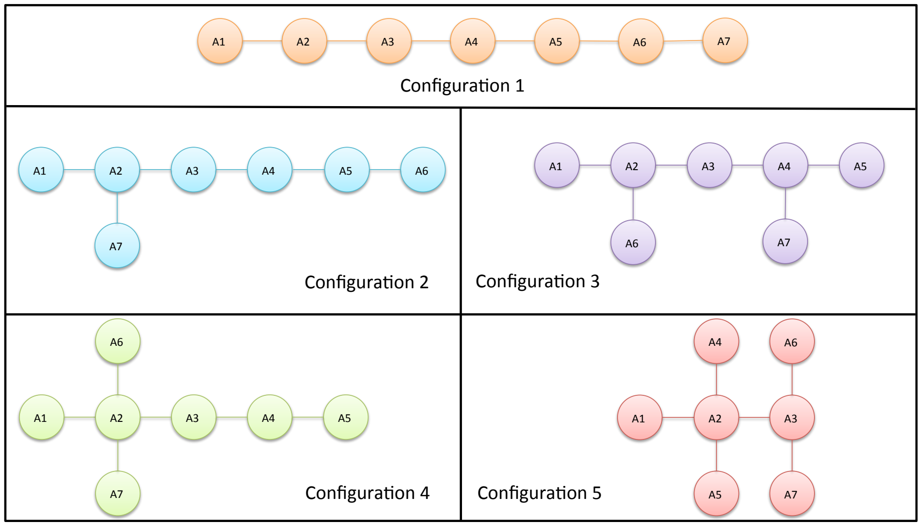


Fig. 12: Topology Configurations

6 Conclusions and Discussions

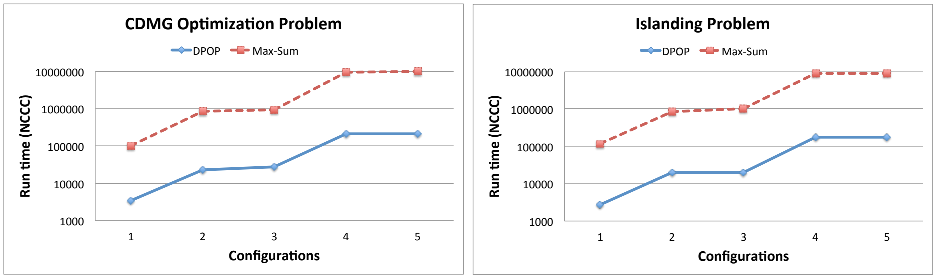
In this paper, we attempt to answer the challenge posed by Ramchurn *et al.* to solve smart grid optimization problems with artificial intelligence techniques [21, 23] by showing how one can model two types of customer-driven microgrid (CDMG) optimization problems, namely the comprehensive CDMG optimization problem and the islanding problem, as distributed constraint optimization problems (DCOPs) and solve them using existing off-the-shelf DCOP algorithms. This decentralized formulation is appealing as it allows the problem to be solved in parallel and maintains the privacy of users by providing only local information to each agent instead of the information of the entire problem. One of the challenges for our future work is to exploit domain properties to increase the scalability of these algorithms in order to deploy them in the real world.

Acknowledgment

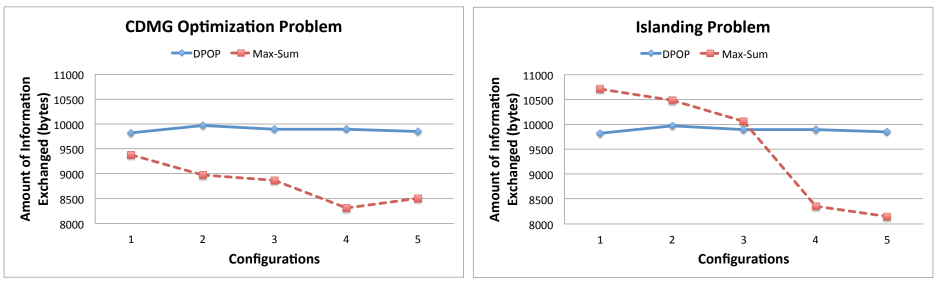
This work is partially supported by the Department of Energy under the microgrid education and research grant DE-OE0000098.

References

1. S. Bukowski and S. Ranade. Communication network requirements for the smart grid and a path for an IP based protocol for customer driven microgrids. In *Proceedings of the IEEE Energytech*, pages 1–6, 2012.
2. U. S. Department of Energy. Grid 2030: A national vision for electricity's second 100 years. Technical report, Department of Energy, United States of America, 2003.



(a)



(b)

Fig. 13: Experiments Varying the Topology Configuration

3. P. Du and J. K. Nelson. Two-step solution to optimal load shedding in a micro-grid. In *Proceedings of the IEEE Power and Energy Society (PES) Power Systems Conference and Exposition (PSCE)*, pages 1–9, 2009.
4. A. Farinelli, A. Rogers, A. Petcu, and N. Jennings. Decentralised coordination of low-power embedded devices using the Max-Sum algorithm. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 639–646, 2008.
5. K. Gampa, S. Ranade, P. Jain, M. Balakrishnan, and S. Yemewar. Performance analysis of capacity discovery algorithm on hardware platform. In *Proceedings of the North American Power Symposium (NAPS)*, pages 1–7, 2010.
6. Y. Hamadi, C. Bessière, and J. Quinqueton. Distributed intelligent backtracking. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, pages 219–223, 1998.
7. M. Hassanzahraee and A. Bakhshai. Transient droop control strategy for parallel operation of voltage source converters in an islanded mode microgrid. In *Proceedings of the IEEE International Telecommunications Energy Conference (INTELEC)*, pages 1–9, 2011.
8. Y.-Y. Hong and S.-Y. Ho. Determination of network configuration considering multiobjective in distribution systems using genetic algorithms. *IEEE Transactions on Power Systems*, 20(2):1062–1069, 2005.
9. O. Ipinnimo, S. Chowdhury, and S. P. Chowdhury. Voltage dip mitigation with dg integration: A comprehensive review. In *Proceedings of the IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)*, pages 1–10, 2010.

10. P. Jain, S. Gupta, S. Ranade, and E. Pontelli. Optimum operation of a customer-driven microgrid: A comprehensive approach. In *Proceedings of the IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)*, 2012.
11. P. Jain and S. Ranade. Capacity discovery in customer-driven micro-grids. In *Proceedings of the North American Power Symposium (NAPS)*, pages 1–6, 2009.
12. P. Jain, S. Ranade, and S. Srivastava. Island identification in customer-driven micro-grids. In *Proceedings of the IEEE Power and Energy Society (PES) Transmission and Distribution Conference and Exposition*, pages 1–7, 2010.
13. A. Kumar, B. Faltings, and A. Petcu. Distributed constraint optimization with structured resource constraints. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 923–930, 2009.
14. T. Léauté, B. Ottens, and R. Szymanek. FRODO 2.0: An open-source framework for distributed constraint optimization. In *Proceedings of the Distributed Constraint Reasoning Workshop*, pages 160–164, 2009.
15. A. Meisels, E. Kaplansky, I. Razgon, and R. Zivan. Comparing performance of distributed constraints processing algorithms. In *Proceedings of the Distributed Constraint Reasoning Workshop*, pages 86–93, 2002.
16. S. Miller, S. Ramchurn, and A. Rogers. Optimal decentralised dispatch of embedded generation in the smart grid. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 281–288, 2012.
17. J. Mitra and M. R. Vallem. Determination of storage required to meet reliability guarantees on island-capable microgrids with intermittent sources. *IEEE Transactions on Power Systems*, 27(4):2360–2367, 2012.
18. P. Modi, W.-M. Shen, M. Tambe, and M. Yokoo. ADOPT: Asynchronous distributed constraint optimization with quality guarantees. *Artificial Intelligence*, 161(1–2):149–180, 2005.
19. A. Petcu and B. Faltings. A scalable method for multiagent constraint optimization. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1413–1420, 2005.
20. S. A. Pourmousavi and M. H. Nehrir. Demand response for smart microgrid: Initial results. In *Proceedings of the IEEE Power and Energy Society (PES) Innovative Smart Grid Technologies*, pages 1–6, 2011.
21. S. Ramchurn, Perukrishnen, Vytelingum, A. Rogers, and N. Jennings. Putting the 'smarts' into the smart grid: A grand challenge for artificial intelligence. *Communications of the ACM*, 55(4):86–97, 2012.
22. S. Ranade, J. Mitra, S. Suryanarayanan, and P. Riberio. A holistic approach to customer-driven microgrids. Technical Report NSF Grants ECCS#0702208 and #0757956, 2011.
23. A. Rogers, S. Ramchurn, and N. Jennings. Delivering the smart grid: Challenges for autonomous agents and multi-agent systems research. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 2166–2172, 2012.
24. W. Yeoh and M. Yokoo. Distributed problem solving. *AI Magazine*, 33(3):53–65, 2012.
25. R. Zivan. Anytime local search for distributed constraint optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, pages 393–398, 2008.