Indian Journal of Science and Technology, *Vol 9(46), DOI: 10.17485/ijst/2016/v9i46/105478, December 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

An Approach to Distributed Coordinated Multi-Agent Energy Supply Negotiation and Planning

Alexander Tsarev* and Petr Skobelev

Smart Solutions, Ltd., Samara, Russia; Mail@Identifier.At, Petr.Skobelev@Gmail.Com

Abstract

Objectives: The relevance of the article is determined by the importance of studying the problems of the distributed coordinated muti-agent energy supply planning. **Methods:** This paper describes a multi-agent approach to the distributed energy supply planning addressing the issues identified during the analysis of the use of the previously developed software prototype. **Findings:** The description includes the principle of agent interaction in the distributed environment, the distributed system architecture, and the basics of the interaction API allowing universal connection to the system. The proposed approach addresses several important challenges (scalability, isolated data processing, support of manual planning, extensibility) and allows real-time energy production and distribution management in complex open environments. The proposed API provides an interaction basis for specialized planning systems working together in a heterogeneous open environment. **Applications/Improvements:** The proposed approach to distributed coordinated multi-agent energy supply negotiation and planning can be utilized for creation of solid industrial software platforms.

Keywords: Coordination, Distributed Planning, Energy Production, Energy Supply, Energy Market, Multi-Agent Approach, Real-Time Scheduling, Smart Grid

1. Introduction

Growing complexity and dynamics of modern global market demand new paradigms in resource management[1]. New revolutionary approach to increase efficiency of business is associated today with real-time economy, which requires adaptive reaction to events, ongoing decision making on resource scheduling and optimization and communication results with decision makers.

Energy sector actively develops in the recent years raising many issues related to distributed generation and reverse power flow from local microgrids[2], use of renewable sources, such as photovoltaic and wind generators operated simultaneously with traditional generators, use of batteries and fuel-cells. Smart [3] grids provide new possibilities for interactive customer and supplier communication, and building new market structure with direct participation of smaller customers and suppliers.

There are many unpredictable events that can disrupt smooth delivery of energy, the most important being the change of the production capacity (due to weather conditions, equipment breakdowns or lack of fuel), arrival of an unexpected order (sudden increase of demand in the nearest time) and cancellation or modification of a previously defined demand. With the intensive flow of events, which is normal in modern life, our planned and actual reality will always quickly diverge. The currently proposed schedule optimization methods do not work well for large grids[4].

This paper is based on the experience of energy supply planning using the previously developed multi-agent real-time energy production and distribution planning prototype[5], and proposes further development of the concepts and solutions for the identified challenges. It focuses on the distributed planning as it seems to be the most important issue for industrial application of the solution.

Multi-agent technology is considered as a new design methodology and framework to support distributed problem solving methods in real-time scheduling and optimization of resources[6].

** Author for correspondence*

We use the multi-agent approach to achieve real-time processing on large and interconnected energy grids. An individual software agent represents each unit (site) of the electrical network. Depending on the type of the unit, its agent can request energy supply (with demand and price defined for each time interval) from other agents, or provide energy (with limitations and costs defined for each time interval) to other requesting agents, or both.

The most basic act of multi-agent communication can be described as three steps:

- In order to achieve a result (plan of energy production and distribution) the consumer agents start to request the power supply from the less expensive supplier sites (including generators, storages, energy routers, external networks). The expected cost of supply from each connected site is based on the agent experience of previous negotiations.
- In response to the consumers' requests, the supplier builds its own internal plan of supply for the full time horizon considering the constraints and the possibility to store energy. When the plan is ready, the site agent sends sub-requests to other suppliers and costs with limitations to the consumers.
- If the consumer receives limitations (the energy cannot be fully supplied) or additional costs (including the cost of supply from the next site in the network and cost of all energy losses) it may change the orders fully or partially and request energy from another directly connected site.

Using these three basic steps, all agents asynchronously communicate with each other. The energy supply requests propagate through the network accumulating the costs and limitations until a solution is found. If a specific agent has several options of demand satisfaction, it explores them by trying suppliers subsequently over the processing iterations.

2. Analysis of Complex Network Planning Challenges

Based on the analysis of practical cases using the developed energy supply planning prototype, we identified several points where an improvement is needed.

First, the multi-agent approach that we use for planning is based on the balancing of interests of the individual agents in the supply network with the focus on the total profit (or other performance indicator) of the network as a whole. For the sake of the network total efficiency, an individual agent often sacrifices its own efficiency. E.g., an energy supplier may sell the energy cheaper and go down to a minimal price to win the contract. All agents compete with each other to achieve a better result for the network. The problem is that sometimes it does not fit into the real practice, when, e.g. two energy plants belong to the same company.

The effects of the company's policy are illustrated in the following figures. Figure1 we can see three plants. Each plant has its own generation costs, and declares the minimal price to the consumer to be competitive. As a result, the consumer chooses the plants A and C to cover the demand, as they propose the lowest price.

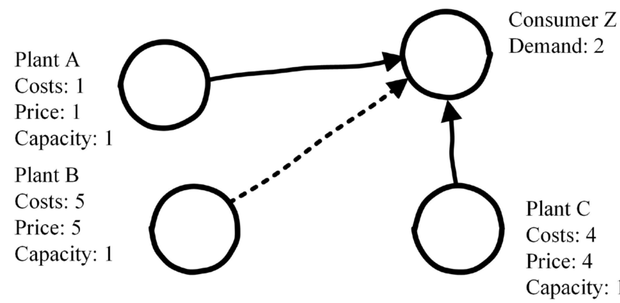

Figure 1. Negotiation between individual agents.

If we consider the plants A and B belong to the same company, it means that the company gets one contract and earns one coin (abstract currency unit). It makes sense for the company to use another price policy for negotiations, which considers the ownership of the plants. If the company sets an average price and it is still competitive, it may take a bigger market share. Figure 2 the price on both plant A and plant B is set to 3 (average generation costs). As a result, the consumer chooses these plants as suppliers, and they earn six coins, which is significant benefit for the company, although the network profit in total is lower.

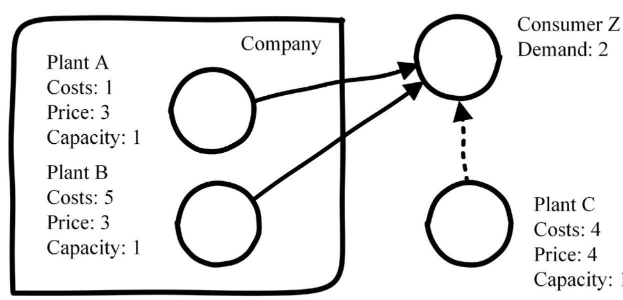

Figure 2. Negotiation between companies.

Besides the ability to define their own policies and agent negotiation logic, the independent companies also require their data to be safe. This means that the way how the agents related to different companies communicate, should guaranty that they do not exchange sensitive data, that can be used (even indirectly) by other companies in competition. In extreme case, the companies require the ability to perform their internal planning in their isolated environment, and only interact with other companies via the coordinated planning system using minimum data needed for negotiation.

Another important technical challenge is the processing time for complex networks. With the increase of the number of participants (e.g., separate micro grids) in the network to be processed, and with the increase of the details taken into consideration, the processing time increases. Figure 3 shows the dependency. The minimal event processing (just reaction to event, finding the closest solution) depends linearly on the network size. However, in practice, a deeper analysis is often required to achieve reasonable planning quality. In this case, the processing time increases faster with the growth of the network complexity, which makes it difficult to use single multi-agent environment for processing.

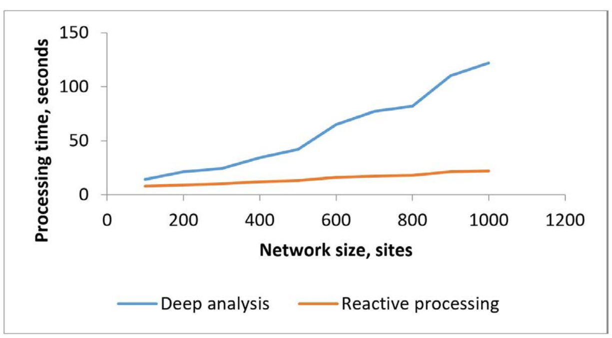

Figure 3. Event processing time for different network sizes.

At the same time, the part of the network affected by the event does not depend much on the network complexity. It may depend significantly on the type of event and its scale, but it does not increase much with the increase of the number of participants (suppliers, consumers). This means, that for the most part of the network the plan will not change after the processing, and the decisions can be made based on the current plan. This makes it reasonable to process parts of the network independently, and then coordinate the results.

Besides the improvement of the processing time, the distributed coordinated planning allows further extension of the network and integration with other sectors (fuel delivery, energy-consuming production) to produce better plans considering limitations on all steps of the processes.

3. Proposed Approach to Distributed Processing

We see that for practical application of the planning software in open market involving several independent companies, it is important to let the companies have different policies, isolated data processing, and standardized API for interaction between different planning systems.

At the same time, it is crucial for planning scalability to perform the planning in a distributed, but coordinated way, to reduce processing time, but keep the plans synchronized.

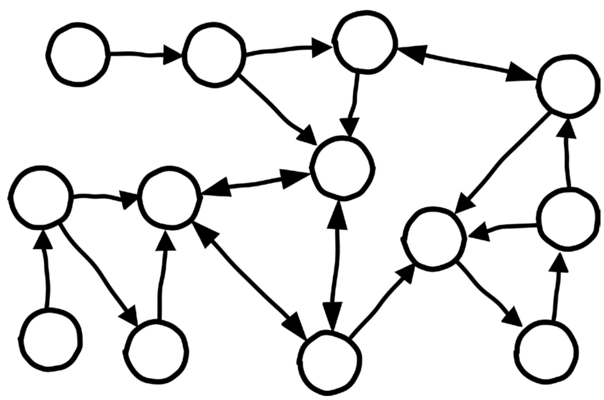

Figure 4. Example of the energy supply network.

Based on the multi-agent approach, it is possible to solve the problem in a distributed way, running software agents on the company-related hardware instead of the central server. This approach is very suitable for solving tasks in open environments. However, it is important to consider that the structure of the energy grid can change dynamically (customers and suppliers can connect and leave the network).

To illustrate the proposed approach to the distributed coordinated planning, let's consider the abstract network example shown in Figure 4. Each node (site) in the network may supply or consume energy, and the channels are used to deliver the energy to other sites. Currently, the developed software prototype uses a separate agent for each site, and the agents communicate and negotiate in the single environment.

The internal constraints of each site in the network are hidden from external actors and affect the result via the restrictions and costs calculated and presented to neighbor sites in the network during negotiations.

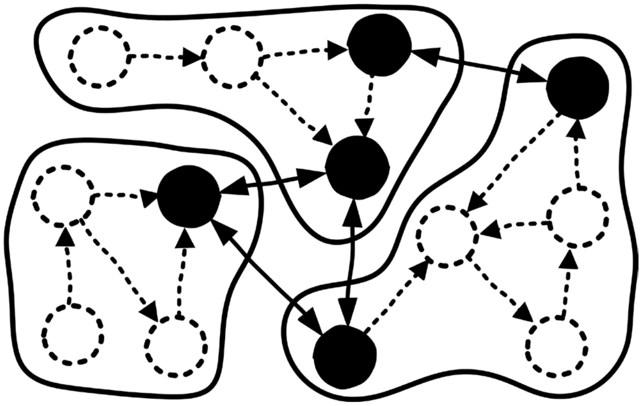

Figure 5. Example of the network segmentation.

The further development of the planning software is seen in the establishment of subnetwork agents or company agents (depending on the situation) that are responsible for separate parts of the network that can work independently. Then, each sub-network may be processed in a separate multi-agent environment, on separate re-mote hardware units interconnected via Internet in a P2P way. Then, the structure of the example changes to the following (Figure 5). The black sites in this figure are the only visible outside of the subnetwork they belong to. The channels available be-tween the subnetworks form the P2P communication channels between the multi-agent environments (swarms).

Thus, in this example we have three subnetwork agents connected via P2P service bus (Figure 6). Each subnetwork agent "sees" the other connected subnetwork agents and the site agents that are directly connected via the channels. The sites that are visible from another subnetwork receive representative agents in this external network. These representative agents have limited functionality and only accept re-quests from other agents in the subnetwork to deliver them later to the real agent in another swarm. They also receive results (including planned limitations and costs) from the real agent and may respond fast to local requests inside the subnetwork swarms they are delegated to considering the previously detected limitations.

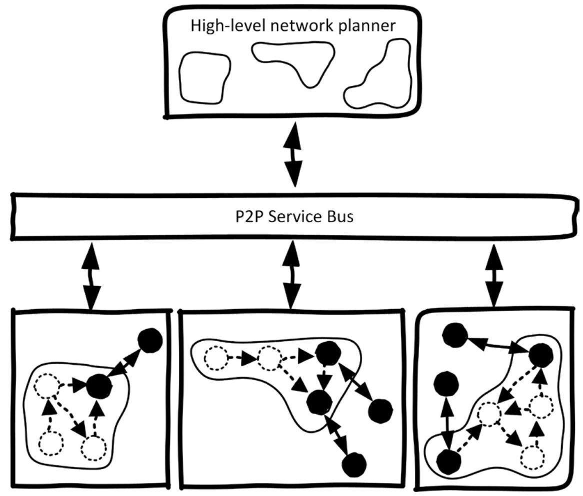

Figure 6. Distributed architecture.

The planning principle is that any event is processed first inside the directly influenced multi-agent environment (planning application). If the result of this processing includes new supply requests to external sites or new supply limitations and costs affecting the existing contracts, the subnetwork agent negotiates changes with other subnetwork agents in the network.

The subnetwork agent is also responsible for the intensity and reasonability of the communication between the subnetworks (or companies). It may introduce addition-al virtual costs on the external requests, and thus motivate the local agents to solve the problems locally. It also decides when to coordinate with other subnetworks de-pending on how many changes (events to be sent) is accumulated and how fast the other agents respond.

It is important to note that the distributed coordinated planning system may include not only automatic multi-agent software planners, but also legacy planning software and even manual planning software depending purely on users. In this case, these software components are also connected to the same service bus using the same protocols and API. Users can see the incoming events, requests and limitations from other subnetworks, and can propose decisions based on the current plans, available options and target KPIs.

4. API Interaction

The important part of the open distributed negotiation and planning system is the standard interface that allows communication between the subnetwork planners. The

API should support several features, including:
- Easy bulk change requests with mixed add, change and remove of the domain objects.
- Support for aliases of the domain objects to easily address objects created externally.
- Uniform representation of demand and plan to allow easy manual rework of the planned operations into the fixed demands with user preferences.
- Support of costs and limitations response to external demands.
- Non-transact processing to let the asynchronous communication and dynamic connection to the system.
- Human-readable requests.
- Support of silent and warning-feedback modes of request processing.

The following domain objects are proposed to model events, requests, and responses sent between the systems connected to the distributed coordinated multi-agent supply negotiation and planning system:
- Site – a node, where a product can be stored, produced (converted) or consumed.
- Channel – an unidirectional supply channel which connects two nodes together.
- Product – a product (type of energy, fuel, etc.).
- Stock – a record about a balance of the certain product at the certain node of the network (e.g., accumulated energy).
- Consumption – demand (forecast or order) of a product at a certain node at time.
- Recipe – description of the necessary raw materials, time, cost and outcome of the production process (generation).
- Transformation – request for production (conversion) of products.
- Delivery – products delivery request for the specific network node.
- Relocation – request for product transportation between network nodes.
- Transformation Limit – temporary production restrictions.
- Relocation Limit – temporary transportation restrictions.
- Calendar – work schedule of the supply chain node.

5. Conclusion

The proposed approach to distributed coordinated multi-agent energy supply negotiation and planning address the following challenges and potentially allows the creation of solid industrial software platform for distributed planning:
- Scalability via distributed planning and communication between remote systems for coordinated results.
- Isolated data processing of independent companies.
- Support of manual planning and conventional planning software as a part of the distributed planning system.
- Extensibility of the system due to its open nature and flexible API for interaction.

6. Acknowledgements

This work was carried out with the financial support of the Ministry of Education and Science of the Russian Federation (contract 14.579.21.0084, project unique id RFMEFI57914X0084).

7. References

1. Park A, Nayyar G, Low P. Supply chain perspectives and issues, a literature review. Fung Global Institute and World Trade Organization; 2013.
2. Vasquez JC, Guerrero JM, Miret J, Castilla M, de Vicuña LG. Hierarchical control of intelligent microgrids. IEEE Industrial Electronics Magazines. 2010; 4(4):23–9.
3. Arai J, Iba K, Funabashi T, Nakanishi Y, Koyanagi K, Yokoyama R. Power electronics and its applications to renewable energy in Japan. IEEE Circuits System Magazines. 2008; 8(3):52–66.
4. Levron Y, Guerrero JM, Beck Y. Optimal power flow in microgrids with energy storage. IEEE Power Systems. 2013; 28(3):3226–34.
5. Skobelev P, Tsarev A. Multi-agent supply scheduling system prototype for energy production and distribution advances in practical applications of scalable multi-agent systems. Switzerland: Springer; 2016. p. 290–3.
6. Mohammadi A, Akl S. Scheduling algorithms for real-time systems [Technical Report no. 2005-499]. Kingston: School of Computing Queen's University; 2005. p. 1–49.