



**AALBORG UNIVERSITY**  
DENMARK

**Aalborg Universitet**

## **Modeling and Managing Energy Flexibility Using FlexOffers**

Pedersen, Torben Bach; Siksnyš, Laurynas; Neupane, Bijay

*Published in:*

IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids

*DOI (link to publication from Publisher):*

[10.1109/SmartGridComm.2018.8587605](https://doi.org/10.1109/SmartGridComm.2018.8587605)

*Creative Commons License*  
CC BY 4.0

*Publication date:*  
2018

*Document Version*  
Accepted author manuscript, peer reviewed version

[Link to publication from Aalborg University](#)

*Citation for published version (APA):*

Pedersen, T. B., Siksnyš, L., & Neupane, B. (2018). Modeling and Managing Energy Flexibility Using FlexOffers. In *IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids: SmartGridComm 2018* (pp. 1-7). IEEE. <https://doi.org/10.1109/SmartGridComm.2018.8587605>

### **General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal -

### **Take down policy**

If you believe that this document breaches copyright please contact us at [vbn@aub.aau.dk](mailto:vbn@aub.aau.dk) providing details, and we will remove access to the work immediately and investigate your claim.

# Modeling and Managing Energy Flexibility Using FlexOffers

Torben Bach Pedersen  
*Department of Computer Science*  
*Aalborg University*  
 Denmark  
 tbp@cs.aau.dk

Laurynas Šikšnys  
*Department of Computer Science*  
*Aalborg University*  
 Denmark  
 siksny@cs.aau.dk

Bijay Neupane  
*Department of Computer Science*  
*Aalborg University*  
 Denmark  
 bn21@cs.aau.dk

**Abstract**—The recent spread of distributed renewable energy sources and smart IoT devices offer exciting new possibilities for the use of energy flexibility, opening a new era of the so-called bottom-up or cellular energy systems. In order to harness the full potential of flexibility, flexibility has to be modeled and represented in a manner that can be efficiently managed, manipulated, and traded on a market. In this paper, we provide a comprehensive overview of the *FlexOffer* concept, which offers an effective way of modeling and managing energy demand and supply flexibilities from a wide range of flexible resources and their aggregates. First, we define the basic concept and present the different phases of the *FlexOffer* life-cycle. Then, we discuss more advanced internal *FlexOffer* constraints as well as algorithms for *FlexOffer* generation, aggregation, disaggregation, and pricing that can significantly reduce energy management and trading complexities and increase overall efficiency. Finally, we present a general decentralized system architecture for trading flexibility (*FlexOffers*) in existing and new markets. Our experimental results show that (1) *FlexOffers* can be extracted with up to 98% accuracy, (2) aggregation and disaggregation can scale to 1000K *FlexOffers* and more, and (3) flexibility can be traded in the NordPool flexi order market while providing up to 89.9% (of optimal) reduction in the energy cost.

## I. INTRODUCTION

Recently, there has been a great focus on a new so-called bottom-up or cellular energy system. A cellular system is an energy system where significant energy production is occurring at lower levels of the grid, and energy market roles such as “producers” and “market operators” have new opportunities at the lower levels of the grid. Each grid level may possess most of the functions present at higher levels. Each portion of a system that contains a minimal set of roles may be called a cell. In the future energy system, such cells may very well sit within other cells in a manner similar to nesting dolls. This cellular approach to arranging the energy system is well suited to incorporating distributed sources of renewable energy. A group of actors capable of meeting their own energy needs can trade energy and flexibility with each other and form their own cell. The cellular system works bottom-up: the smallest cells (subsystems) are commercial and residential buildings, houses and industry plants (e.g. performing as micro-grids). The next level typically correspond to distribution grids and the third level to transmission grids. Flexibility enables an individual

cell to dynamically maintain the demand and supply balance locally. However, to be able to exchange information about energy flexibility among different actors within a cell, there is a need for a common representation of flexible loads. The European project MIRABEL proposed a format to encode this information, called a *FlexOffer* (FO) [2], [3], [5], [8], [16].

The remainder of the paper is organized as follows. Section II presents the FO concept. Section III presents the life-cycle of FO from generation to execution. Section IV presents various FO constraints. Section V presents the general FO generation architecture. Section VI presents details on aggregation and disaggregation of individual FOs into larger FO(s) and vice-versa. Section VII presents the detail on pricing FO. Section VIII presents how the various system components are interfaced. Section IX present the experimental results to demonstrate the viability of FO based flexibility market. Section X provides a discussion on the overall FO approach, including pros and cons. Finally, Section XI concludes the paper and provides directions for future research.

## II. FLEXOFFER CONCEPT OVERVIEW

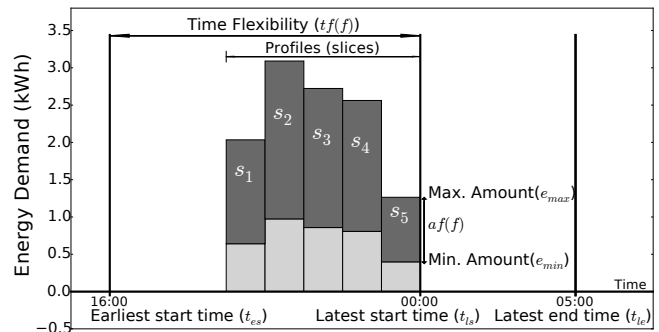


Fig. 1. A visual representation of the simple *FlexOffer*.

A visual representation of a simple FO is shown in Fig. 1. Each bar in the graph corresponds to a time slice of energy consumption, with the lower part representing the minimum amount of energy that a flexible resource needs to provide its service, and the upper part an interval within which it can adjust its consumption, while still satisfying functional constraints (e.g., comfort temperature). This is called amount

(energy) flexibility. Another type of flexibility is time flexibility also shown in Fig. 1. Time flexibility is provided when an energy load can be shifted within a time interval, defined by an earliest start time at which the flexible resource can start its consumption, and a latest end time at which it should be done.

*Definition 1:* A *FlexOffer*  $f$  is a tuple  $f = ([t_{es}, t_{ls}], p)$ , where  $[t_{es}, t_{ls}]$  is the start time flexibility interval and  $p$  is the amount profile. The time is discretized into equal-sized units, e.g., 15 minute intervals. Thus, we use  $t_{es} \in N$  to specify the earliest start time and  $t_{ls} \in N$  to specify the latest start time. The  $p$  is a sequence of slices  $\langle s_1, \dots, s_d \rangle$ , where a slice  $s_t$  is a continuous range  $[e_{min}, e_{max}]$  defined by the minimum  $e_{min}$  and maximum  $e_{max}$  energy bounds, and  $d$  is the number of slices in  $p$ .

Once created, a FO is assigned a baseline schedule that corresponds to the consumption pattern that the associated flexible resource prefers to follow. Updated schedules can be assigned to the FOs to modify the consumption behavior of the flexible resource, utilizing its provided flexibility. Fig. 1 exemplifies the energy demand for a single charging event of an electric vehicle (EV) along with the flexibility. The FO in the figure states that the EV could be charged anytime between 4 PM and 12 AM. The FO has an energy profile with five consecutive slices, where the light-shaded area represents the base demand and the dark-shaded area represents the flexible demand. The time flexibility of the FO is the difference between LST and EST. Similarly, the total amount flexibility is given by the sum of the dark-shaded area of all slices. More advanced forms of the FO exist, and will be discussed later. In the next section, we describe the complete FO life-cycle.

### III. FLEXOFFER LIFE CYCLE

A FO goes through several phases during its life-cycle as depicted in Fig. 2 and described below:

- 1) **Data collection:** The first phase involves collection and storage of energy consumption data from flexible devices (e.g., washing machine, dishwasher, EV, etc.). Cheap off-the-shelf Smart plugs may be used to collect data at device level granularity, whereas energy meters may be deployed to collect data at lower granularity (e.g., entire house).
- 2) **Preprocessing:** The collected energy data may contain outliers caused by abnormal device behavior. It may also suffer from missing data caused by errors during data collection, or users mistakenly interrupting device operation. The outliers and missing data can have an adverse effect on the quality of the FO. Therefore, the purpose of the preprocessing step is to remove outliers and handle missing data.
- 3) **Predictive model building and demand prognosis:** A FO basically contains the future energy and time flexibility information, which requires that the future energy demand and the associated flexibility to be predicted beforehand. Thus, we need to develop Machine Learning

(ML) models to analyze user behavior and predict future energy demand to generate FOs [10].

- 4) **FlexOffer generation:** In this phase, the predicted energy demand and the associated flexibility data is used to generate FOs that are used to exchange flexibility information between different entities.
- 5) **FlexOffer aggregation/disaggregation:** Here, individual FOs are combined into large aggregated FOs to make them more useful and computationally tractable [12], [14]. Details are provided in Section VI.
- 6) **FlexOffer trading:** Aggregated FOs are traded on the market. The market returns aggregated schedules in response to accepted FOs.
- 7) **Schedule execution:** Aggregated schedules are disaggregated and forwarded to the corresponding flexible loads.

Until now, we have discussed a simple FO. However, a FO can take a complex form, which we will discuss in next section.

### IV. MODELING FLEXOFFER CONSTRAINTS

To cover a wide range of flexible resources, a FO identifies different types of flexibilities and uses specialized constraints to characterize each type of flexibility – see Fig. 4. A FO in a simple form (Fig. 1) activates the following types of constraints to capture start time and energy amount flexibilities:

- **Start time constraint** – it is a range defined by the two parameters: *startAfterTime* ( $t_{es}$ ), *startBeforeTime* ( $t_{ls}$ ) – see Fig. 1.
- **Energy amount constraint** – for every discrete (e.g., 15min) interval of an active device operation, energy amount flexibility is characterized by a range with *energyConstraint.lower* and *energyConstraint.upper* as lower and upper bounds, respectively – see Fig. 1.

However, some flexible resources might require additional more advanced (flexibility) constraints to be enabled in a FO to model its flexibility or data exchange flow more accurately:

- **Total energy constraint** – it is a range [*totalEnergyConstraint.lower*, *totalEnergyConstraint.upper*] which bounds the total energy amount requested or offered within the full active operation of a flexible resource.
- **Dependent energy amount constraint** – like the energy amount constraint, this constraint captures the minimum and maximum energy amounts at a discrete interval  $t$  depending on the total energy consumed at the intervals  $1 : t - 1$ . A dependent energy amount constraint for a specific discrete time interval  $t$  is given as 2D energy flexibility polygon – see Fig. 3. This constraint should be used for the most advanced forms of flexible resources (e.g., heat-pumps), where the flexibility changes over time and is dependent on an internal system state (e.g., temperature).

*Definition 2:* A dependency-based FlexOffer (DFO) is a series of polyhedrons (polygons) in  $\mathbb{R}^2$ ,  $P_1, P_2, \dots, P_T$ , which permit the series of energy amounts at  $T$  consecutive time intervals,  $e_1, e_2, \dots, e_T$ , such that  $(\sum_{i=1}^{t-1} e_i, e_t) \in P_t$  for each  $t \in 1 : T$ .

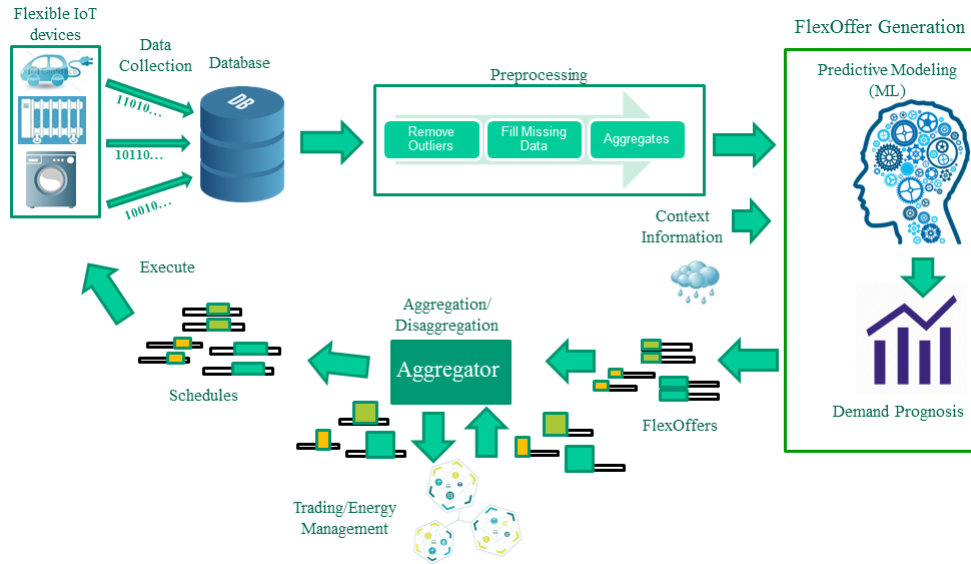


Fig. 2. FlexOffer life-cycle: process view.

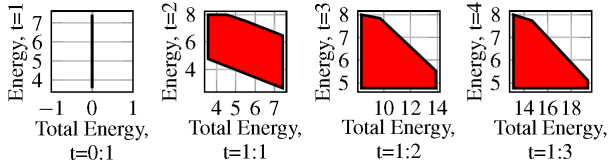


Fig. 3. An example of a FlexOffer with dependent energy amount constraints.

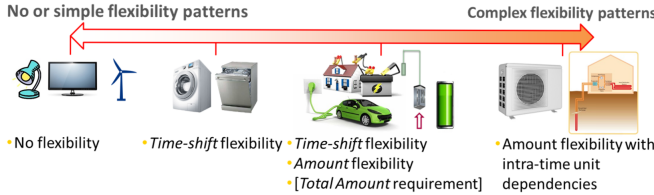


Fig. 4. Complexity of load flexibility patterns supported by FlexOffers.

- **Acceptance time constraint** – defined by the parameter *acceptanceBeforeTime*, this constraint sets the deadline on when a FO receiving party (e.g., Balance Responsible Party (BRP)) should acknowledge successful acceptance or rejection of the FO. A FO rejection may occur if, e. g., FO constraints or other metadata (e.g., prices) are invalid or inappropriate (e.g., quantities are too small, prices are too high).
- **Assignment time constraint** – defined by the parameters *assignmentBeforeTime* and *assignmentBeforeDurationSeconds*, this constraint sets the deadlines on when FO schedule update (assignment) is allowed to be sent by the FO receiving party (BRP) to a FO issuing party (flexible resource). A deadline can be an absolute timestamp (*assignmentBeforeTime*) or a relative duration (*assignmentBefore – DurationSeconds*) with respect to

the scheduled operation activation time. For all these FO constraint types, scalable aggregation and disaggregation techniques have been developed [4], [6], [13]–[15].

## V. FLEXOFFER GENERATION

Due to a difference in operating behavior of devices, the flexibility available from their operation is also different, e.g., dishwasher might only provide time flexibility, heat-pump only amount flexibility, and EV could provide both time and amount flexibility. Hence, a FO generator model should be modular enough to be able to extract flexibility from all device types [11]. Fig. 5 shows a general architecture of the FO generation model. The FO generation process starts with the gathering of the energy demand time series and available context information such as the description of house occupants, house insulation parameters, etc. The next step includes the preprocessing of the raw information into a format required for analyzing and predicting timestamps and values for the actions captured by FOs. The *Model Parameter Estimation and Forecasting* step includes the application of forecast models required for the generation of FOs. Specifically, *Model Parameter Estimation and Forecasting* step includes two modules i) *Time Flexibility Extraction* and ii) *Amount Flexibility Extraction*. Depending on the device type, either both modules get executed or only one of them is executed. For example, to generate FOs from a dishwasher, only time flexibility extraction module is required. As discussed before, FOs are generated well before the actual production and consumption of energy takes place. Hence, each flexibility extraction module, depends on the forecasting component to get a prognosis of future energy demand, associated flexibility, and timestamps. For example, to extract time flexibility from a dishwasher, the *Time Flexibility Extraction* module uses prediction component to get an estimation of earliest and latest start time of the dishwasher operation.

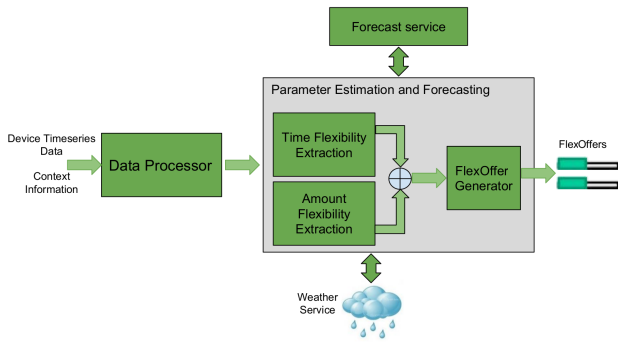


Fig. 5. A generic FlexOffer generation model.

Further, the flexibility extraction module may also require weather and temperature data for estimating amount flexibility from *Thermostatically Controlled Loads* (TCL), e.g., boilers, heat-pumps, etc. The final step combines the outputs of the two modules to generate FOs for the forecasted device operations.

## VI. FLEXOFFER AGGREGATION AND DISAGGREGATION

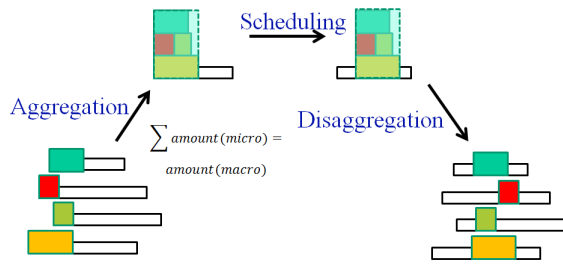


Fig. 6. FlexOffer aggregation, scheduling, and disaggregation process.

In general, the FO representation makes it practical to exchange flexibility information between different entities. However, FOs from individual flexible resources (e.g., heat-pumps, electric vehicles) typically represent small flexible loads. Thus, a single (small) FO has low impact and is of little interest for electricity trading, peak shaving, and balancing demand and supply on the grid, where the required balancing capacities are much higher. At the same time, optimizing energy loads based on a large number of FOs is a computationally hard problem, which requires dealing with many decision variables and constraints originating from many FOs. By utilizing FO aggregation [14], [19], [20], flexibilities from individual appliances can be combined and thus offered in a more useful and effective aggregated form. Such aggregated flexibility can again be represented as FOs – but with much larger energy amounts and flexibility margins. Aggregation is typically performed by BRPs, system operators, and/or entities called *Aggregators*. They receive FOs from individual flexible resources and then aggregate these FOs. The flexibility of aggregated FOs tends to be lower than the joint flexibility of the FOs that compose them. This reduction in flexibility is, however, unavoidable in order to reduce FO scheduling

complexity and to increase their value (e.g., on the flexibility market). After aggregation, schedules are typically assigned to the aggregated FOs (e.g., based on energy sold on the market). By respecting all inherent aggregated FO constraints, a schedule specifies exact start times and aggregated energy amounts to be assigned to the underlying flexible resources. Such schedules are disaggregated to schedules for each individual FO it is composed of. This operation is denoted by FO disaggregation. Disaggregated schedules are finally forwarded to the flexible resources which initially offered flexibility. This FO aggregation, scheduling, and disaggregation process is illustrated in Fig. 6.

## VII. FLEXOFFER PRICING

In addition to flexibility, a FO may capture related information such as price. Each FO may define, a deviation price, which has to be paid to flexibility provider (e.g., Prosumer) by a flexibility consumer (e.g., BRP), for requesting a kWh deviation from the baseline (preferred schedule) which was initially reported by the flexibility provider – see Fig. 1. This yields a number of V-shaped price curves (in the linear case) associated to each discrete time interval of an active device operation – see Fig. 7. The deviation price is paid for

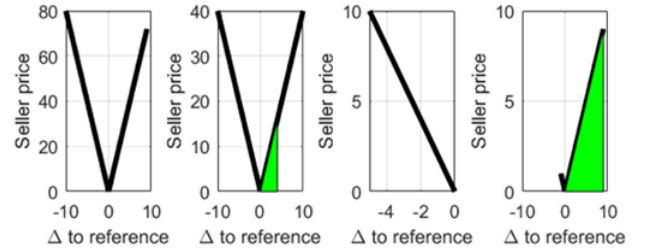


Fig. 7. Pricing of flexibility based on FlexOffer contracts.

the successful activation of flexibility (a deviation from the baseline). Alternatively, flexibility may be treated as an asset and priced in the explicit form (e.g., monthly). For this, the flexibility provider (a flexible resource) may get rewarded by the flexibility consumer (e.g., BRP) based on the number of FOs, total time and energy amount flexibility, etc. offered, see Fig. 8.

Customer: arrowhead_lift		
Item	Value	Price
Number of flexoffers	2	
Fixed reward for all flexoffers		10.00 DKK
Total Time Flexibility	0 time units (15 min)	0.00 DKK
Total Energy Flexibility	68.83 kWh	6.88 DKK
Number of default schedule deviations	0	0.00 DKK
The sum of stat time scheduling deviations with respect to the default schedule	0 time units (15 min)	0.00 DKK
The sum of energy deviations with respect to the default schedule	0.00 kWh	0.00 DKK
Total Reward		16.88 DKK

Fig. 8. An example of a FlexOffer with associated deviation prices.

## VIII. SYSTEMS INTEGRATION

FlexOffer collection, management, and trading is typically realized using the decentralized platform shown in Fig. 9.

The *FlexOffer Agent* allows all kinds of energy consumers and/or producers offering flexibility in the form of FOs. It provisions individual FOs, delivers FOs to the *FlexOffer Manager*, receives disaggregated schedules from the *FlexOffer Manager*, and activates the corresponding flexible loads according to the received schedules.

The *FlexOffer Manager* allows energy market players such as aggregators, suppliers, and balance responsible parties managing flexibilities (FlexOffers) offered by *FlexOffer Agent* users. In simple cases, *FlexOffer Manager* may operate in isolation and offer efficient FlexOffer aggregation and energy optimization functionalities, e.g., for demand-supply balancing, congestion management. In more advanced cases, the *FlexOffer Manager* may first aggregate flexibility (FlexOffers) from many small loads (e.g., household appliances, EVs) and then trade FOs on one or more markets. In such cases, the *FlexOffer Manager* is also responsible for disaggregation of the schedules received from the market. The *FlexOffer Manager* may support direct trading (e.g., local, national, or regional markets like *NordPool*<sup>1</sup>) or indirect trading through an existing market entities such as aggregators or market operators (TotalFlex, Arrowhead, GOFLEX). Flexibility can be traded in day-ahead, intra-day, or intra-hour.

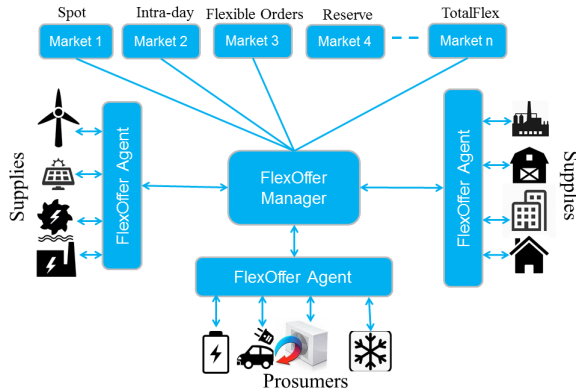


Fig. 9. Interfacing of flexibility market components.

## IX. EXPERIMENTAL RESULTS

In this section, we summarize the experimental results from our previous papers with a focus on quantifying flexibility potential of household TLC and non-TLC devices, evaluating accuracy of FOs generation method, scalability of the FOs aggregation, and the financial benefits of FOs.

**1) Flexibility Potential:** In [9], we analyze device-level energy consumption data from 6 different households [7]. A data aggregation based on device flexibility types, i.e., fully-flexible (e.g., EV, heat-pump, dishwasher), semi-flexible (e.g., oven), and non-flexible (e.g., lights) devices, demonstrate that on

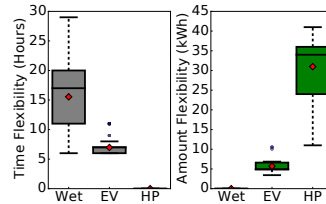


Fig. 10. Average time and amount flexibility: over various device type

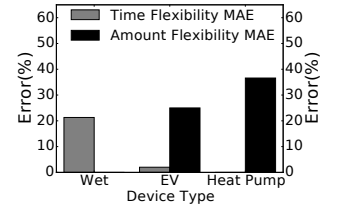


Fig. 11. Time and amount flexibility error: over various device types

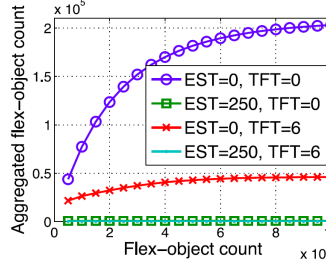


Fig. 12. Flex-offer compression factor: for various parameters

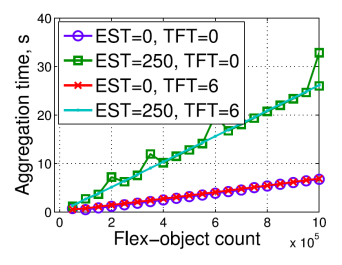


Fig. 13. Flex-offer aggregation time: for various FOs count

average 50% of the energy demand from the household comes from flexible devices.

**2) Flex-offer Generation:** In [11], we analyze the performance of the FO generation model in terms of quantity and quality of extracted flexibilities and generated FOs. The distribution of time and amount flexibilities extracted from 50 heat-pumps, 30 dishwashers, and 100 EVs, using the FO generation model presented in Section V, is shown in Fig. 10. The result demonstrates that on average wet-devices (dishwashers, washer dryer, etc.) and EVs provide 15.31 and 7 hours of time flexibility, respectively. Similarly, heat-pumps and EVs provide 33kWh and 5.7kWh of amount flexibility, during winter and summer, respectively. The accuracy of the proposed FO generation model for extracting flexibilities is illustrated in Fig. 11. The figure illustrates that the FO generation model can extract time and amount flexibility with up to 98% and 79% accuracy, respectively. The regularity in energy consumption pattern for EVs resulted in higher accuracy for extracted flexibilities. On the other hand, uncertainty in usage pattern for wet-devices lowers the accuracy, which has been investigated in [10]. Nevertheless, even with a very low traditional accuracy measure, the proposed FO generation model can still give a good utility, i.e., positive financial benefits to a market. Hence, the achieved accuracy is sufficient to increase market confidence in utilizing FOs for balancing the deviations in the portfolio [10], [11].

**3) Aggregation and Disaggregation:** In [14], we evaluate the strategies for aggregation and disaggregation of FOs while retaining flexibility. To evaluate the performance and scalability of FO compression, a number of experiments are performed with a varying number of FOs, i.e., aggregation and disaggregation are performed with FOs ranging from 50 to 1000k. Aggregation is performed using two different

<sup>1</sup>www.nordpoolspot.com

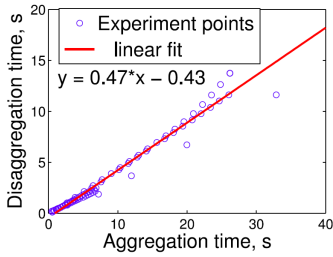


Fig. 14. Comparison of aggregation and disaggregation time

Earliest Start Time Tolerance (EST) and Time Flexibility Tolerance (TFT) parameter values: EST equal to 0 or 250, and TFT equal to 0 or 6. Disaggregation is executed with randomly generated instances of aggregated FOs. Fig. 12 and 13 illustrate that different parameters lead to different compression factors (ratio of normal FOs to aggregated FOs) and aggregation times <sup>2</sup>. Fig. 13 demonstrates that small FOs can be efficiently aggregated into larger FOs in linear time complexity. Fig. 14 demonstrate that disaggregation is approximately two times faster than aggregation regardless of the FOs count and parameter values. The experiments were conducted on a standard laptop.

We also considered an energy planing scenario where an aggregator aggregates, schedules (optimizes), and disaggregates 10K heat-pump-like loads, modeled as approximate standard FOs (see Definition 1) and *dependency-based FOs* (see Definition 2) as well as exact *linear time-invariant state space models* (LTI) and multi-dimensional polyhedrons (POLY) [13]. As seen in Fig. 15, the complete planning task (aggregation, scheduling, and disaggregation) takes less than 10 minutes to complete for the time horizon of 24 hours when standard FOs are used. However, standard FOs loose up to 85% of the flexibility (load scheduling potential), compared to the exact models (LTIs and POLYs). 10K dependency-based flex-offers (DFOs) covering 24 hour time intervals can be aggregated, scheduled, and disaggregated in less than 20 minutes, retaining 50-70% of the load flexibility of the exact models. In contrast, when 10K exact LTI models are used, scheduling of loads based on linear programming becomes infeasible (in more than 1 hour of execution) even for very short time horizons ( $T=3$ ) due to the excessive number of decision variables and constraints (problem complexity). Similarly, the aggregation and disaggregation of 10K polyhedrons (POLYs) is infeasible (in 1 hour of execution) even for very short time horizons ( $T=3$ ), due to the expensive Minkowski sum operation and disaggregation problem solving. Therefore, standard or dependency-based FOs make this task computationally feasible, while offering different levels of trade-off between performance and flexibility loss.

**4) Flexibility Trading:** In [17], [18], We investigated a market-based FO aggregation and trading using the FO model to capture flexible charging loads from EVs. The flexibility

<sup>2</sup>In Fig. 12 and 13, the term flex-objects is used to represent FOs.

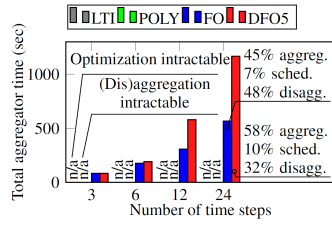


Fig. 15. Total aggregator time (10K models)

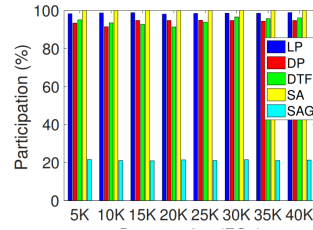


Fig. 16. Average participation of FOs

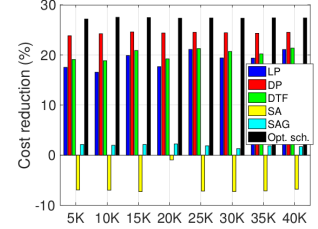


Fig. 17. Average traded energy

from the EVs are aggregated based on a market requirement and are traded in the market as a flexible order – a special order (bid) type of NordPool market [1]. A flexible order is a block order with a fixed price limit and fixed volume with duration of 1 hour. The actual activation of the order is not determined by a user but is automatically decided by the algorithm. Further, we also evaluate the reduction in the cost of charging that can be achieved by various aggregation techniques. Namely, i) Longest Profile (LP), ii) Dynamic Profile (DP), iii) Dynamic Time Flexibility (DTF), iv) Start Alignment (SA) - baseline model, and v) Start-Alignment Grouping (SAG) – see [18] for details. Fig. 16 illustrates the average percentage of energy traded in the market with respect to the size of the aggregated FOs. An aggregated FO can have a very high variance in power assigned to the slices. Since, the flexibility market (flexible order) requires a flat power profile, the power of the highest slice need to be considered for the whole profile of the aggregated FO.

SA positions FOs at their earliest start time ( $t_{es}$ ) generating one large aggregated FO. Since flexible orders are traded at the highest power of the FO slices, SA aggregation technique results in 2.5 times the energy captured by the aggregated FO to be traded. In particular, SA trades 1.52 times more energy than needed to charge the EVs. Thus, the surplus energy is traded in the regulation market, which results in losses for a BRP (see a negative cost reduction in Fig. 17). On the other hand, LP technique focuses on producing aggregated FOs with many slices, making it difficult to fulfill the flexible order amount requirements, especially, the required slice amount equality. DP excludes the FOs with extremely large profiles from aggregation, giving a shorter and balanced aggregated FOs. Hence, DP achieves 88.9% (of the optimal) reduction in the charging cost, with almost 97.5% of energy captured by FOs traded in the market [17], [18].

## X. DISCUSSION

In this section, we will discuss the pros and cons of modeling and managing flexibility using FOs, and discuss implementation challenges.

### Pros:

- FOs enable unified modeling of flexibilities from a variety of loads. Unlike for example polytopes, FOs are messages to communicate flexibility across two parties, with associated techniques for generation, aggregation, optimization, etc.

- FOs enables efficient and scalable generation, aggregation, and optimization of flexibilities from a large number of heterogeneous sources.
- FOs can be used in with different time granularities, ranging from coarse-grained (1 hour and up) to fine-grained (15 mins).

#### Cons:

- FO defines flexibility at consecutive fixed-length time intervals and has a small latency between generation to final execution, hence FO is not suitable for true real-time market (that operates in msec resolution).
- Some flexibility is inevitably lost during FO encoding and aggregation.
- FOs specify energy amounts and do not directly consider active/reactive power and other real-time parameters such as voltage, frequency, etc.

#### Challenges:

- Encoding flexibility for long time horizons requires increasing the number of FO slices or their durations, thus lowering overall performance or optimization accuracy.
- FO defines energy flexibility as a continuous range of options. Other forms of FOs are still being explored - those with discrete energy amount combinations, raising aggregation complexity.

## XI. CONCLUSION AND FUTURE WORK

This paper presents a comprehensive overview of modeling and managing of energy demand and supply flexibilities utilizing FlexOffers. Specifically, it presents general and constraints-based FlexOffers models and their life-cycle from generation to execution. It introduces various constraints to characterize flexibility for a wide range of flexible resources. The paper discusses the FlexOffer generation model along with the required input data and information. Further, it elaborates the need and benefits of aggregating smaller FlexOffers into larger FlexOffer-s, especially in terms of reduced complexity and trading. Thereafter, it discusses FlexOffer pricing mechanism and integration of the FlexOffer system to existing and new energy markets. The experimental results show that heat-pumps and EVs provide significant amount flexibility and the FO generation model can capture flexibility with up to 98% accuracy. The FOs can be efficiently aggregated into larger FOs with a small flexibility loss and high compression ratio. The FO algorithms (aggregation and disaggregation) are highly scalable and can easily scale to 1000K FOs. Further, we demonstrate that 10K dependency-based flex-offers can be aggregated, scheduled, and disaggregated in less than 20 minutes, retaining 50-70% of the load flexibility. Further, FOs can be effectively traded in the various energy markets. The experiments on the NoodPool flexibility market shows that FOs can provide up to 89.9% (of optimal) reduction in the energy cost.

In our future work, we will further explore the flex-offer based modeling of flexibility, along with enriching our techniques for various tasks such as demand prognosis, aggrega-

tion, etc. Furthermore, we will deploy and validate the concept in a real-world implementation.<sup>3</sup>

## REFERENCES

- [1] Nord pool market, 2017. <https://www.nordpoolgroup.com/trading/Day-ahead-trading/Order-types/Flexi-order/>.
- [2] The GOFLEX project, 2017. <https://www.goflex-project.eu>.
- [3] The MIRABEL project, 2013. <http://www.mirabel-project.eu>.
- [4] The virtual market of energy (vme) implementation developed in the arrowhead project. [https://github.com/lawrzs/ARROWHEAD\\_VME](https://github.com/lawrzs/ARROWHEAD_VME).
- [5] M. Boehm, L. Dannecker, A. Doms, E. Dogvan, B. Filipič, U. Fischer, W. Lehner, T. B. Pedersen, Y. Pitarch, L. Šikšnys, et al. Data management in the mirabel smart grid system. In *Proceedings of the 2012 Joint EDBT/ICDT Workshops*, pages 95–102. ACM, 2012.
- [6] L. L. Ferreira, L. Siksnys, P. Pedersen, P. Stluka, C. Chrysoulas, T. Le Guilly, M. Albano, A. Skou, C. Teixeira, and T. Pedersen. Arrowhead compliant virtual market of energy. In *19th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2014*, 2014.
- [7] J. Z. Kolter and M. J. Johnson. REDD: A Public Data Set for Energy Disaggregation Research. In *SustKDD workshop*, 2011.
- [8] T. Le Guilly, L. Siksnys, M. Albano, P. Pedersen, P. Stluka, L. Lino Ferreira, A. Skou, T. B. Pedersen, and P. Olsen. An energy flexibility framework on the internet of things. *The Success of European Projects using New Information and Communication Technologies*, pages 17–37, 2016.
- [9] B. Neupane, T. B. Pedersen, and B. Thiesson. Towards flexibility detection in device-level energy consumption. In *International Workshop on Data Analytics for Renewable Energy Integration*, pages 1–16. Springer, 2014.
- [10] B. Neupane, T. B. Pedersen, and B. Thiesson. Utilizing device-level demand forecasting for flexibility markets. In *Proceedings of the Ninth International Conference on Future Energy Systems, e-Energy '18*, pages 108–118, 2018.
- [11] B. Neupane, L. Šikšnys, and T. B. Pedersen. Generation and evaluation of flex-offers from flexible electrical devices. In *Proceedings of the Eighth International Conference on Future Energy Systems, e-Energy '17*, pages 143–156, 2017.
- [12] L. Šikšnys, M. E. Khalefa, and T. B. Pedersen. Aggregating and disaggregating flexibility objects. In *Scientific and Statistical Database Management*, pages 379–396, 2012.
- [13] L. Šikšnys and T. B. Pedersen. Dependency-based flexoffers: scalable management of flexible loads with dependencies. In *Proceedings of the Seventh International Conference on Future Energy Systems*, page 11. ACM, 2016.
- [14] L. Šikšnys, E. Valsomatzis, K. Hose, and T. B. Pedersen. Aggregating and disaggregating flexibility objects. *IEEE Transactions on Knowledge and Data Engineering*, 27(11):2893–2906, 2015.
- [15] E. Valsomatzis, K. Hose, and T. B. Pedersen. Balancing energy flexibilities through aggregation. In *International Workshop on Data Analytics for Renewable Energy Integration*, pages 17–37, 2014.
- [16] E. Valsomatzis, K. Hose, T. B. Pedersen, and L. Siksnys. Measuring and comparing energy flexibilities. In *EDBT/ICDT Workshops*, pages 78–85, 2015.
- [17] E. Valsomatzis, T. B. Pedersen, and A. Abelló. Day-ahead trading of aggregated energy flexibility. In *Proceedings of the Ninth International Conference on Future Energy Systems, e-Energy '18*, pages 134–138, 2018.
- [18] E. Valsomatzis, T. B. Pedersen, and A. Abello. Day-ahead trading of aggregated energy flexibility-full version. *arXiv preprint arXiv:1805.02301*, 2018.
- [19] E. Valsomatzis, T. B. Pedersen, A. Abelló, and K. Hose. Aggregating energy flexibilities under constraints. In *Smart Grid Communications (SmartGridComm), 2016 IEEE International Conference on*, pages 484–490, 2016.
- [20] E. Valsomatzis, T. B. Pedersen, A. Abelló, K. Hose, and L. Šikšnys. Towards constraint-based aggregation of energy flexibilities. In *Proceedings of the Seventh International Conference on Future Energy Systems Poster Sessions, e-Energy '16*, page 6, 2016.

<sup>3</sup>In the final version, we will compress the paper to 6 pages.