Load Disaggregation with Metaheuristic Optimization

Dominik Egarter and Wilfried Elmenreich

Institute of Networked and Embedded Systems / Lakeside Labs
Alpen-Adria-Universität Klagenfurt, Austria
dominik.egarter@aau.at, wilfried.elmenreich@aau.at

Abstract. Fine-grained power readings based on smart metering are the basis for future energy saving introduced by consumption information on appliance level. To get appliance level information, non-intrusive load monitoring (NILM) can be used to detect which appliance was turned on when. With NILM it is possible to break down the household power draw, power readings of the smart meter, to its appliance-induced components. In this work we introduce a simple load disaggregation approach based on metaheuristic optimization. Six different metaheuristics algorithms are tested to its performance to disaggregate power draws from aggregate power readings. We show how the simple proposed approach based on active power readings can contribute to solve the problem to disaggregate aggregated loads.
Keywords: Load Disaggregation, Metaheuristic Optimization, Smart Metering, Non-Intrusive Load Monitoring

Increasing energy awareness is an important cornerstone for a more efficient usage of energy. In particular, if information about the momentary energy consumption of all her appliances are available to a user, she can make more effective decisions on reducing energy consumption by using energy-hungry devices to a lesser extend or by exchanging these devices with more efficient alternatives. Basically, there are three possibilities to provide a detailed overview of the momentary consumption of different devices: (i) Monitor each appliance in a home by a smart plug or smart socket, (ii) employ smart appliances [6] that can measure and report their consumption, or (iii) infer about the consumption state based on aggregate measurements from a single meter. However, possibility (i) requires considerable hardware effort, installation cost, and might come with significant additional power consumption from the meters in the smart plugs. Solution (ii) comes with high investment costs for smart appliances and requires all appliances of interest to be equipped as smart appliance. Solution (iii), also known as NILM

This work was performed in the research cluster Lakeside Labs funded by the European Regional Development Fund, the Carinthian Economic Promotion Fund (KWF), and the state of Austria under grants 20214/22935/34445 (Smart Microgrid Lab). The authors would like to thank the anonymous reviewers for their valuable comments.

is more cost efficient, but requires good software algorithms to disaggregate the total power draw into separate ones. The approach of NILM is non-intrusive in a way that it does not require the installation of smart appliances or further meters. NILM or load disaggregation, was first introduced by G. Hart [9] in which he used active and reactive power measurements to model and classify appliances. All up to now proposed NILM approaches can be divided into supervised and unsupervised learning and classification approaches. To get an adequate overview of state-of-the-art NILM approaches we refer to [15] and [16].

In this paper, we are approaching the problem of supervised load disaggregation; in particular to disaggregate appliance power data from the aggregate power draw by an optimization approach. The appliance characteristics described by an appliance model (consisting of the power consumption per appliance operation state) are known *a-priori*. We optimize an objective function retrieving the information which appliance is running at which point in time.

According to Hart [9] the load disaggregation problem can be modelled as a subset sum problem. Hart faced the problem that even if all power states of appliances are known small fluctuations and similarities between power states lead to an dramatically decrease in the performance of the approach. Therefore, in this paper we model the load disaggregation problem as a knapsack problem to verify the statement provided by Hart.

The knapsack problem is NP-complete, which means there is no fast (polynomial time) algorithm that guarantees a correct solution. Although for low number of items, or in our case, devices, the knapsack problem can be still solved using exact algorithms, inaccuracies in the measurement or power models cause further problems when applying exact optimization algorithms, iterative methods, or simple heuristics. In contrast, metaheuristic optimization promise a faster solution at the cost of correctness and can handle inaccurate measurement and modeling. Thus, we approach the problem with state-of-the-art metaheuristic optimization approaches such as i) the *evolutionary algorithm* [5], ii) *differential evolution* [2], iii) *particle swarm optimization* [10], iv) *simulated annealing* [8], v) the *cuckoo search algorithm* [7] and vi) *firefly optimization* [14]. We evaluate the performance of the optimization-based load disaggregator on real world data in which similar and realistic consumption behaviors are present.

The remainder of this paper is organized as follows: Section 1 provides a comprehensive summary of all used metaheuristic optimization approaches. Section 2 describes the proposed approach and how to model the load disaggregation problem as a knapsack problem. In Section 3 the evaluation settings are presented. Section 4 provides evaluations of case studies to assess the proposed approach. Finally, Section 5 discusses the presented results and Section 6 summarizes the paper.

1 Meta-Heuristics

1.1 Evolutionary Algorithm

The evolutionary algorithm (EA) is a population-based optimization approach inspired by the evolution of natural life [5]. A set of individuals represents a population. The algorithm aims at optimizing the population according to a fitness or objective function over several generations. The individuals are modified by the evolutionary operators mutation (mutation of individuals), recombination (combination of individuals) and selection (selection of individuals, which will survive, i.e., remain in the population for the next generation).

1.2 Differential Evolution

The differential evolution (DE) [2] is a special form of the EA and, therefore, population-based. Each population of a generation consists of a candidate solution of the objective function. The algorithm maintains candidate solutions and creates new solutions by combining existing ones based on simple formulas. The candidate solutions are evaluated based on a given fitness or objective function as for the evolutionary algorithm.

1.3 Particle Swarm Optimization

The particle swarm optimization (PSO)[10] is a population-based algorithm, in which the population with its candidate solutions is represented as a swarm of particles. The aim of the swarm is it to move around the search space guided by their position and the position of the best candidate solution.

1.4 Simulated Annealing

The simulated annealing (SA)[8] approach is inspired by the annealing of metallurgy, where metal is heated up and slowly cooled down to strengthen the metal structure by rearranging the crystal structure. It implements a random search, where each step decreasing the objective function (in case of a minimization problem) is accepted. A step increasing (worsening) the objective function is accepted probabilistically based on the change of the objective function and a simulated "temperature" which decreases over time as the solution converges.

1.5 Cuckoo Search Algorithm

The cuckoo search algorithm (CS)[7] is inspired by the brood parasitism of cuckoo species laying their eggs into nests of other birds of different species. An egg in a nest is representing a solution of an optimization problem and an cuckoo egg stands for an new solution.

1.6 Firefly Algorithm

The firefly algorithm (FA)[14] is inspired by the flashing behavior of Asian fireflies which flash synchronously to attract other fireflies. The attractiveness is proportional to the brightness of a firefly in which the brightness represents the objective function. In detail, the FA is based on the following rules [14]:

- All fireflies are unisexual. One firefly will be attracted by all other fireflies.

– Attractiveness is proportional to their brightness, thus for any two flashing fireflies, the less brighter one will move towards be brighter one. The attractiveness is proportional to the brightness and the both decrease as their distance increases.
– If there is no brighter one than a particular firefly, it will move randomly.
– The brightness of a firefly is affected or determined by the landscape of the objective function

The objective function is represented by the brightness of the fireflies.

2 Approach

The power demand of a household results form the used appliances. Typically, each appliance has a characteristic way of consuming energy. For example, one appliance is consuming a high amount of energy for a short period of time or another appliance behaves in a multi-state manner consuming in each running state a different amount of power. The total power load can be considered as the superimposition or the aggregation of the power profiles from each appliance over time. G. Hart (et. al [9]) had the idea to use the knowledge of appliance characteristics and the aggregate power demand to introduce the problem of load disaggregation or non-intrusive load monitoring (NILM). NILM breaks down the aggregate power demand to its components on appliance level. He classified the problem to disaggregate appliances. It is computational intractable and belongs to the group of NP-complete problems. The aim of the load disaggregator is to find the best composition of appliance power states to minimize the error between the estimated and the real signal. There are several approaches visible to solve the load disaggregation problem. In this paper we are concentrating on an supervised approach solved by optimization. The first task is it to define an objective function to be optimized. Therefore, the aggregate power draw $P(t)$ can be described as

$$P(t) = p_1(t) + p_2(t) + \cdots + p_n(t) \text{ for } t \in \{1, T\}, \tag{1}$$

where $p_i(t)$ is the power profile of each appliance in the set of N appliances and t represents the discrete time vector from 1 to T. The problem is to find the best set of appliance power profiles whereas each power profile is activated by an appliance being in a on or off state. We can model the problem as

$$e(t) = \mid (P(t) - \sum_{i=1}^{N} p_i \cdot a_i(t)) \mid \tag{2}$$

where $a_i(t)$ represents the appliance state vector (e.g., appliance is on or off). In this work, we are modelling the presented optimization problem as the so-called knapsack problem [4]. The knapsack problem is a well-known optimization problem with the aim of packing a set of n items with a certain weight w_i and profit d_i into a knapsack of capacity C in the most profitable way. If it is possible to place an item into the knapsack without exceeding the capacity C by using

$x_i \in \{0,1\}$, which is responsible for whether or not a certain item is used, a profit d_i is earned. This context can be summarized as follows

$$\text{maximize} \sum_{i=1}^{n} d_i \cdot x_i, \tag{3}$$

$$\text{subject to} \sum_{i=1}^{n} w_i \cdot x_i \leq C. \tag{4}$$

The problem of packing items into a desired shape can be adopted to the load disaggregation problem. NILM has the aim to disaggregate loads from the aggregate power demand according to their own power profile p_i in the measured total load $P(t)$. The power profiles p_i are mainly characterized by their power magnitude m_i and their time of usage. The total power load is given by:

$$P(t) = \sum_{i=1}^{n} P_i \cdot a_i(t) + e(t), \tag{5}$$

where n is the number of known and used power states, $a_i(t) \in [0,1]$ represents the state vector of a power state being on $(a_i(t) = 1)$ or off $(a_i(t) = 0)$. An off/on appliance would be represented with its corresponding power state, while a multi-state appliance, e. g., astove with multiple cooking plates, can correspond to multiple power states. $e(t)$ describes an error term. The general optimization problem of NILM can be formulated as the minimum error $e(t)$ of the total power load and the estimated aggregation of appliance power profiles:

$$e(t) = \arg\min \left| P(t) - \sum_{i=1}^{n} P_i \cdot a_i(t) \right|. \tag{6}$$

The NILM system tries to find the appliance states by $a_i(t)$ to minimize the error between the sum of superimposed appliance power profiles and the total load $P(t)$. This relates to the knapsack problem, where in the case of NILM the capacity C of the knapsack corresponds to the total load $P(t)$ and the items of the knapsack correspond to the appliance power profiles P_i. We assume that the profit d_i equals 1 since we suppose that all appliances in the household are of equal importance concerning their usage. The aim of any optimization approach is to find a composition of power profiles P_i, which can be packed into the measured total load $P(t)$ with minimum error. Therefore, we modify the general knapsack problem by replacing the profit maximization with an error minimization. An illustration of the basic principle can be seen in Figure 1, where a collection of possible power profiles P_i and the trend of the total power load are presented. In detail, the approach tries to find for each point in time the best composition of appliance power profiles described by their power demands. The optimization approach has to optimize the vector $a_i(t)$ represented as a binary vector. The value 1 means an appliance is on at time t and 0 means an appliance is off at time t. The objective function is represented by

$$F_s = -\left| P(t) - \sum_{i=1}^{N} P_i \cdot a_i(t) \right|. \tag{7}$$

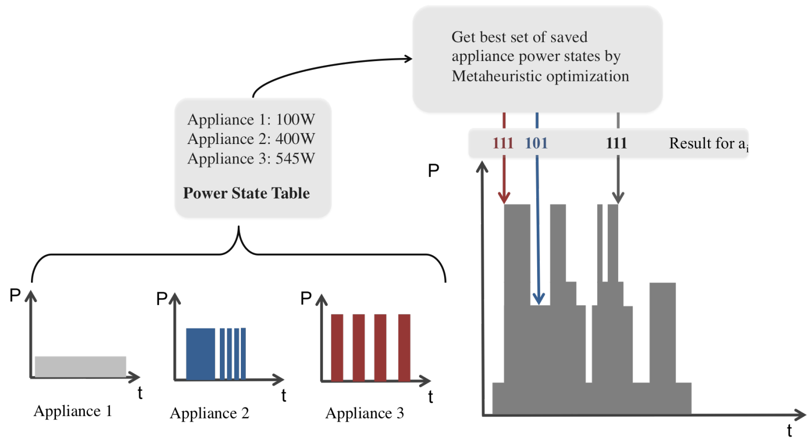

Fig. 1. The approach estimates the optimal set of appliance power states for each used time sample with the table of possible power states

It describes a minimization problem since the optimal fitness is a fitness of 0. The optimization process of the state vector $a_i(t)$ is done on each power sample. We used six different optimization approaches to solve this problem consisting of the i) the *evolutionary algorithm*, ii) *differential evolution*, iii) *particle swarm optimization*, iv) *simulated annealing*, v) the *cuckoo search algorithm* and vi) *fire fly optimization*. If necessary the optimization approaches are modified to be able to work with the problem characteristics. The use of metaheuristics is necessary because the problem suffers on measurement inaccuracies, noisy measurements and modelling errors.

3 Settings

3.1 Algorithm Settings

All metaheuristic optimization approaches are modified to work with discrete inputs. This is mainly done by rounding the results from the continuous case to the discrete values 0 and 1. However, the used metaheuristics are dependent on different parameters. In Table 1 all parameters are presented. The parameters were set due to empirical evaluations.

3.2 Data Settings

To test the proposed approach real measurements of appliances are used. There exists several public available datasets such as the REDD dataset [11], the Eco-dataset [1], the GREEND dataset [13] and the AMPD dataset [12]. We have chosen the REDD dataset as reference dataset. It provides appliance level power measurements in $1s$ resolution for 6 different houses. For our evaluation we took the first house with 6 common appliances which are the oven, the fridge, the dishwasher, the kitchen outlet, the microwave and the washer/dryer. We derived

Table 1. List of parameters for each used metaheuristic optimization approach

Algorithm	*Parameter*	*Value*
EA, DE, PSO FA	No. of generations g	200
EA, DE, PSO	Population size p	100
EA,	Mutation operator	uniform mutation
EA,	Recombination operator	one-point crossover
EA,	Selection operator	elite selection
DE	Crossover probability	0.5
DE	Scaling factor	0.8
PSO	Cognitive parameter $c1$	2
PSO	Social parameter $c2$	1
PSO	Constriction parameter C	1
SA	Cooling steps	200
SA	Maximum initial temperature	100
CS	Number of nests	50
CS	Discovery rate	0.25
FA	Number of fireflies	50
FA	Randomness factor	0.9
FA	Randomness reduction factor	0.95
FA	Absorption coefficient	0.2

the present power states for each appliance. We distinguish between automatic detected power states and power states detected by expert knowledge. In the case of automatic detected power states we used the algorithm presented in [3]. The algorithm consists of de-noising, filtering, edge detection with event detection and clustering of events to appliance states. The states are detected from submetered power draws where similarities between power states are possible. Table 2 lists the used appliances and their characteristics. The expert detected states are identified manually by the human. No similarities between power states were detected. Small power states such as standby power are considered in contrast to automatic detected power states.

3.3 Evaluation Metric

To be able to evaluate the performance of the metaheuristic knapsack approach, we evaluate the energy consumption for each appliance and compare it to the ground truth energy data. The power draw for each power state is optimized individually. Therefore, the power states and their resulting optimization results belonging to an appliance are grouped and compared to the ground truth, respectively.

4 Case studies

To check the applicability of the metaheuristic based load disaggregation approach we introduce two case studies i) appliance set with similar power values and ii) appliance set with unique power values. Effects such as similarities of power values, similar combined power values, and measurement noise affect the optimization result.

4.1 Appliance set with similar power values

In this case study, we decided to use real world consumption data in which appliances have similar consumption behaviour. Thus, the amount of power consumed by an appliance A can be similar to the one consumed by appliance B. The case study should show if the optimization-based approach is able to distinguish between appliances even if they have similar power demands or if the combination of power states leads to another power state. In case of the REDD dataset, we used 6 appliances of house 1 in which we identified the following appliance states for each appliance listed in Table 2 as utilizable.

Table 2. Table of used appliance types, the number of operation states and the corresponding power values for each operation state of the REDD dataset house 1. States were detected by a state detection algorithm

type	power [W]
oven	[0 1690 2455]
fridge	[0 190]
kitchen outlet	[0 210 440 880 1100]
microwave	[0 60 1533]
stove	[0 260 710 1440]
washer/dryer	[0 2712]

Table 3. Table of used appliance types, the number of operation states and the corresponding power values for each operation state of the REDD dataset house 1. States were detected by a human.

type	power [W]
oven	[0 1600]
fridge	[0 8 190 2000]
kitchen outlet	[0 1080]
microwave	[0 5 1550]
stove	[0 1430]
washer/dryer	[0 2700]

The input for each metaheuristic approach are the power states listed in Table 2 and we used an observation window of one day.

In Figure 2, 3 and 4 the energy shares of the optimization approaches and in Figure 5 the ground truth energy shares are presented. The energy shares of the washer/dryer are not shown since the device was off during the whole observation time and the algorithms always detected this fact. We claim detecting an appliance to be off is of the same difficulty as to detect an appliance to be on. There exists no preferred state for the optimization process.

However, the results show that the approach is not able to distinguish between different appliances. It is able to track and to optimize the problem. The mean error between optimized and real power draw is around $13W$ which we consider a satisfying result. Nevertheless, the similarity of power states, the possible representation of a power state by a combination of other states and noise effects are influencing the problem. In detail, the appliances oven, fridge and kitchen outlet have similar power states which severely affects the optimization result. This influence is presented comparing the energy share of the fridge and the kitchen outlet with the real energy shares, respectively.

Moreover, the error in total is comparable high. As reason we claim the influence of noise effects and perfectly modelled appliance states. On the other hand, the influence of different metaheuristic algorithms was less significant. In particular no algorithm was able to handle this problem sufficiently well.

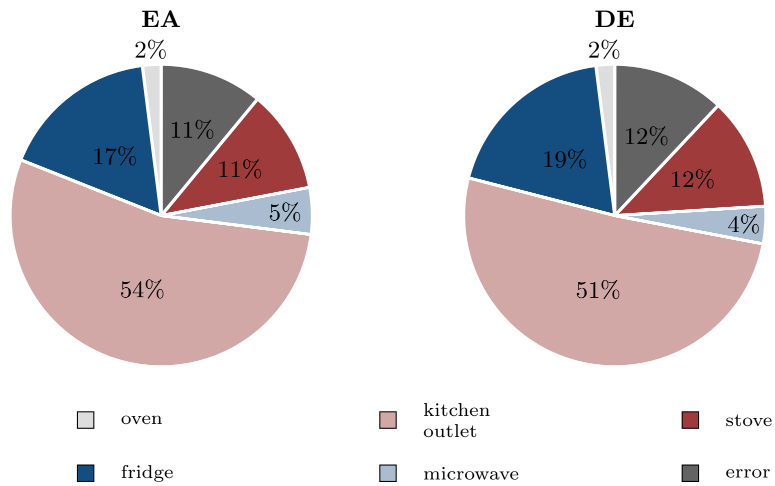

Fig. 2. Energy shares for the optimization results of EA and DE with similar power states

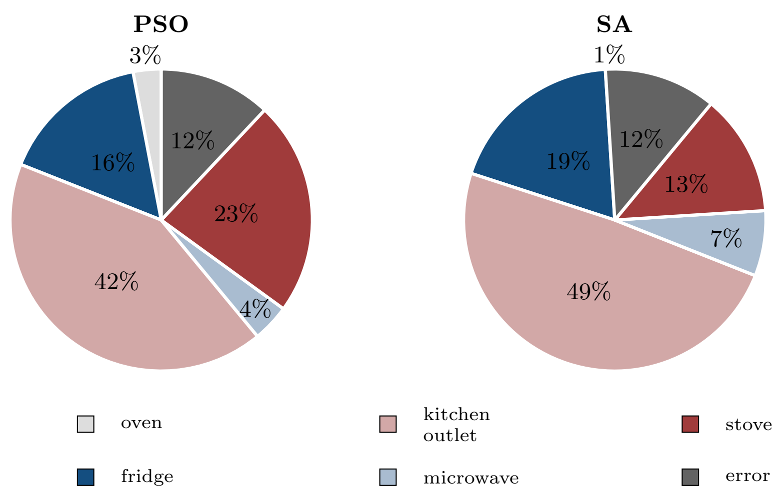

Fig. 3. Energy shares for the optimization results of PSO and SA with similar power states

4.2 Appliance set with unique power states

In contrast to the previous section, we are now considering appliance states which occur almost uniquely. Similarities between appliances are only possible for very small power values such as the standby power. In Table 3 all possible power states are listed. Each state was manually identified by human. The input for the metaheuristic load disaggregator are the power states listed in this table and we used an observation window of one day.

Figure 5 presents the estimated and the real energy share. The results are improved compared to the previous case study. Clearly distinguishable power

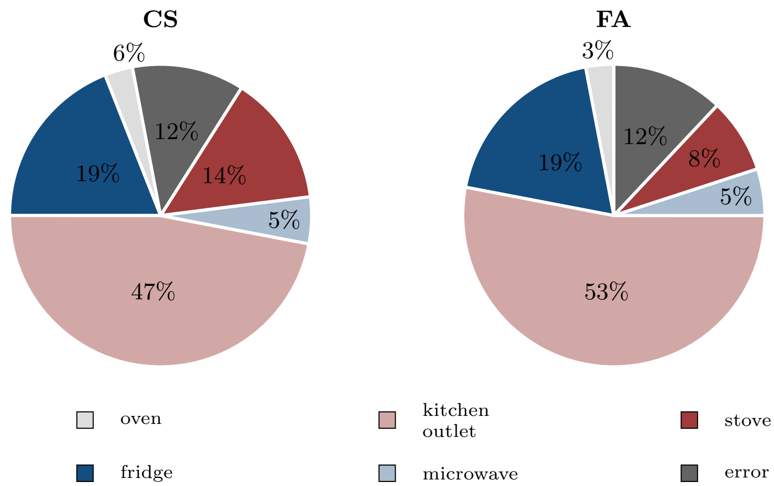

Fig. 4. Energy shares for the optimization results of CS and FA with similar power states

states as for the oven and the fridge can be estimated very well. Nevertheless, also in this case study the effect of similar/recombined of power states and noise effects are present and affect the results. In this evaluation, we took the energy share of the evolutionary algorithm as representative case since all other approaches achieved similar results. Only exception was the result for the simulated annealing case study, where the oven reached only 5% and the stove 14%. The other estimates were very similar. As in the previous case study, the washer/dryer is not shown in this figure since it was not used in the observed time and was well detected by all approaches.

With less similarities in the power draw , the total error of the optimization result is getting better as well as the mean error between optimized and real power draws over the time (error of $7W$). As reason we claim the fact on the one hand the different/dissimilar power states and the lower number of possible power states. The lower the number of considered power states, the better the possible optimization result.

5 Discussion

The presented approaches have not been able to distinguish multiple power states which are similar to each other or can be created by recombining other power states. The reason therefore is the lack of information based on the use of one feature (in our case the power value of a state). As a consequence, the problem has to be modelled by more advanced techniques including appliance structure (e.g., state machine), timing behaviors and probabilistic representation. All presented metaheuristic approaches achieved similar results. The choice of which one taken, can be made by considering the computational time, the number of parameters to be set and the chosen applications. According to our results, we claim that the metaheuristic approach is suitable for NILM applications with a low number

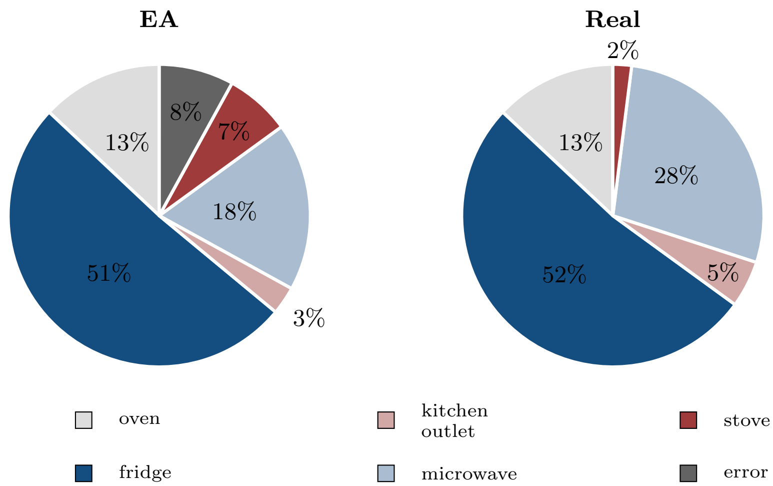

Fig. 5. Energy shares for the optimization results of EA with unique power states and the ground truth energy shares

of appliances with different power states. Possible applications would be a smart power plug or multiple power plug to detect attached appliances.

6 Summary

In this paper a simple load disaggregation approach based on metaheuristic optimization has been presented. Six different metaheuristic algorithms (evolutionary algorithm, differential evolution, particle swarm optimization, simulated annealing, cuckoo search algorithm, firefly optimization) were tested according to their ability to disaggregate loads from the total power demand. Our benchmark problem uses an aggregate power value from a set of appliances and then requires the algorithms to find the underlying composition of appliance power states by minimizing the error between the estimated and real power draw. The approach is related to the well-known knapsack problem and modified according to problem-specific characteristics. The approach was tested on real-word measurements with different sets of appliance power states. The results show that the overall approach provides satisfying results for an appliance set where power draws are easy to separate. For appliances with very similar power draws the optimization approach is not able to disaggregate the overall power draw due to lack of further information which would be necessary to distinguish between similar appliances. Future research will concentrate on the improvement to disaggregate similar appliances from the total power draw. This should be achieved by enlarging the feature set for the optimization approach with further appliance characteristics as reactive power, common time of usage and common time duration in use, essentially then employing a multi-objective optimization approach to solve the disaggregation problem.

References

1. Beckel, C., Kleiminger, W., Cicchetti, R., Staake, T., Santini, S.: The ECO data set and the performance of non-intrusive load monitoring algorithms. In: Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings. BuildSys '14 (2014)
2. Das, S., Suganthan, P.: Differential evolution: A survey of the state-of-the-art. Evolutionary Computation, IEEE Transactions on **15**(1) (Feb 2011) 4–31
3. Egarter, D., Elmenreich, W.: Autonomous load disaggregation approach based on active power measurements. In: Proceeding of the IEEE Workshop on Pervasive Energy Services (PerEnergy). (2015)
4. Egarter, D., Sobe, A., Elmenreich, W.: Evolving non-intrusive load monitoring. In: Proceedings of the 16th European Conference on Applications of Evolutionary Computation. EvoApplications'13, Berlin, Heidelberg, Springer-Verlag (2013) 182–191
5. Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing. SpringerVerlag (2003)
6. Elmenreich, W., Egarter, D.: Design guidelines for smart appliances. In: Proceedings of the 10th International Workshop on Intelligent Solutions in Embedded Systems, Klagenfurt, Austria (July 2012)
7. Gandomi, A.H., Yang, X.S., Alavi, A.H.: Cuckoo search algorithm: A metaheuristic approach to solve structural optimization problems. Engineering with Computers **29**(1) (2013) 17–35
8. Hajek, B.: A tutorial survey of theory and applications of simulated annealing. In: Decision and Control, 1985 24th IEEE Conference on. Volume 24., IEEE (1985) 755–760
9. Hart, G.: Nonintrusive appliance load monitoring. Proceedings of the IEEE **80**(12) (1992) 1870–1891
10. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Neural Networks, 1995. Proceedings., IEEE International Conference on. Volume 4. (Nov 1995) 1942–1948 vol.4
11. Kolter, J.Z., Johnson, M.J.: REDD: A public data set for energy disaggregation research. In: Proceeding of the SustKDD Workshop on Data Mining Applications in Sustainability. (2011)
12. Makonin, S., Popowich, F., Bartram, L., Gill, B., Bajic, I.V.: AMPds: A Public Dataset for Load Disaggregation and Eco-Feedback Research. In: Proceeding of the IEEE Conference on Electrical Power and Energy (EPEC). (2013)
13. Monacchi, A., Egarter, D., Elmenreich, W., D'Alessandro, S., Tonello, A.M.: GREEND: An energy consumption dataset of households in italy and austria. In: Proceedings of IEEE International Conference on Smart Grid Communications (SmartGridComm'14), Venice, Italy (2014)
14. Yang, X.S.: Firefly algorithm, levy flights and global optimization. In: Research and Development in Intelligent Systems XXVI. Springer (2010) 209–218
15. Zeifman, M., Roth, K.: Nonintrusive appliance load monitoring: Review and outlook. IEEE Trans. Consum. Electron. **57**(1) (february 2011) 76 –84
16. Zoha, A., Gluhak, A., Imran, M.A., Rajasegarar, S.: Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. Sensors **12**(12) (2012) 16838–16866