Published in: I. Troch, F. Breitenecker (Eds.), 2006:
Proc. 5th MathMod Vienna. Vol. 2,
Argesim Report No. 30, Argesim-Verlag Vienna, ISBN 3-901608-30-3

OPTIMISED CONTROL OF ADAPTIVE CURRENT CONSUMERS - A FIRST APPROACH

U. Vogel, J. Pei, M. Sonnenschein, Univ. Oldenburg, Germany

Corresponding Author: U. Vogel,
University of Oldenburg, Dept. of Computing Science
Uhlhornsweg 84, D-26111 Oldenburg, Germany
Phone: +49 441 798 2752, Fax: +49 441 798 2756
email: ute.vogel@informatik.uni-oldenburg.de

Abstract: Nowadays, power generation is mostly regulated by the temporal varying power demand. The increasing importance of regenerative, unregulated and fluctuating power stations intensifies the temporal divergence of power supply and power demand and therefore complicates the control of the power supply [6]. As there are many current consuming tasks, which are not bound to be performed at a fixed point of time, but only during a certain time interval, we introduce the problem how to schedule such adaptive power consuming jobs to a given, temporal varying power supply. After a formalisation of the problem, the performances of ten different instantiations of the meta-heuristic Tabu Search are compared regarding the best choice of parameters.

1. Introduction

Conventional methods for producing electrical energy from fossil fuels or nuclear energy depend on limited resources. Experts expect that the peak in world oil production will be reached in the near future and recommend economies to decrease their dependencies on fossil energy resources [3]. Renewable energy sources i.e. wind, waterpower, and biomass, mostly are attributable to solar radiation and hence will be available nearly for ever. It is assessed that the solar energy which daily reaches the earth is about 15.000 times the energy which is actually consumed by all 6 billion people [2]. The solar radiation can be absorbed directly through collectors, wind through rotors, and waterpower through current or tidal power stations; biomass energy is made available through burning (wood) or fermentation as well as gasification (silage, liquid manure) [9].

One mayor drawback of most of these renewable energy sources is their temporal availability: Energy availability from renewable resources usually depends on weather conditions that affect e.g. the effectiveness of solar energy electric power stations and windmills. So, there is no guarantee that renewable energy converters produce the amount of electrical energy which is needed at a certain time. The storage of electrical energy, which would overcome this problem, is yet not sufficiently feasible.

Therefore, short time power supply management (24 h in advance) including renewable energy sources depends on detailed demand estimation and on an estimation of the supply of wind or solar energy converters. For stability control of energy supply on this time scale on one hand the supplier can affect i.e. the output of hydraulic power plant or gas turbines, or buy and sell electricity at the energy exchange market [4]. On the other hand, it is also reasonable to control power consumers in their temporal demand to reduce peaks in power demand, compensate temporal losses in power supply, or generally 'shape' load curves corresponding to supply curves [1].

2. Intention of this paper

Our approach addressed in this paper is to explore the possibilities of shifting the consumption of electrical energy with respect to a prognosis of energy availability for a fixed time interval. This approach seems to be worth to be investigated for several reasons:

- There are energy consuming tasks which need not to be started at a fixed point of time but have to be performed in a certain time interval. Examples of such tasks are found in industrial processes (e.g. paper mills) as well as in the private area (e.g. dish washing).
- The availability of energy from renewable sources, e.g. wind energy, can be predicted with sufficient precision, see e.g. [7].
- The infrastructure which allows the communication between consumers and suppliers is enhancing and is becoming available for a distributed supply site/demand site management [12, 14]. Intelligent home systems are already built, which will allow a simple and convenient control of the entire technology within a building, e.g. the visiomatic® home system. So, the adaptation of tasks to the fluctuating energy supply can be automated and does not need steady human interaction.

For this kind of demand site management (DSM) in power supply networks two alternatives can be examined. On one hand, consumers can control their energy demand 'autonomously' by reacting on 'real-time' signals of the energy supplier (see e.g. [15]). On the other hand, the supplier can directly control energy consumers by shifting their power demand on a given time scale. In this paper we investigate an optimisation method for the second alternative.

First we will introduce a formal model of the optimisation problem (TJAP), which is obviously NP-hard. As this model restricts to schedule only a subset of jobs, we proceed to a more realistic model (JAP), which schedules all jobs and tries to minimise the temporal difference between power demand and supply. The meta-heuristic Tabu Search will be used for the search for an optimal schedule: so, we present the various instantiations of Tabu Search and analyse which of these instantiations is most appropriate for solving the problem.

3. The Model

3.1 Adaptive consumers

Many energy consuming tasks are not bound to be started at a certain point of time. Often it is sufficient, if these tasks are fulfilled in a given time interval. So, throughout this paper, we will consider such tasks as "jobs" j = (a(j), e(j), d(j), m(j)), which are characterized by

- an earliest starting time $a(j) \in T$,
- a latest due time $e(j) \in T$, $a(j) \leq e(j)$,
- a duration $d(j) \in T$, where $d(j) \leq e(j) - a(j)$,
- a demand $m(j) \in \mathbb{R}$ of energy that must be supplied for the duration of the job.

We regard time as a discrete set $T = \{0,..,T_{max}\}$ of points in time and $a(j), e(j), d(j) \in T$.

Let $J \subset T \times T \times T \times \mathbb{R}$ the finite set of all jobs which have to be scheduled in order to find an optimal match to the given power supply in the interval T.

3.2 Power supply model

As we have noted above, the power supply can be predicted with acceptable precision for time intervals of about 24 h in advance. So, for a time interval T we assume that the energy supply is a known function $C: T \rightarrow \mathbb{R}$.

This single, temporal varying resource is a very important aspect, which we have not found to be addressed in related problems.

3.2 The theoretical optimisation problem

Given the set of Jobs J, and the current supply $C: T \rightarrow \mathbb{R}$, an optimal adaptation of jobs to the predicted electrical supply means to find mappings

- $x : J \rightarrow \{0, 1\}$, where $x(j) = 1$ indicates the job j has been chosen, and
- $s : J \rightarrow T$, where $a(j) \leq s(j) \leq e(j) - d(j)$ holds for each chosen job,

so that the demand never exceeds the supply:

$$(1) \qquad \forall t \in T: \ \sum_{s(j) \leq t \leq s(j) + d(j)} m(j) \cdot x(j) \leq C(t)$$

and the overall amount of energy granted to jobs is maximised:

$$(2) \qquad f(x) = \sum_{j \in J} d(j) \cdot m(j) \cdot x(j) \rightarrow \max.$$

In the following, we will refer to this problem as Theoretical Job Adaptation Problem (TJAP).

The well known NP-hard knapsack problem is special case of the TJAP problem. The knapsack problem consists of a knapsack, characterised by a maximum capacity K, and a set O of n objects $o = (w_o, p_o)$, where p_o is the profit and w_o is the (positive) weight of object o. The goal is to find a vector $x=(x_o)_{o \in O}$, $x_o \in \{0,1\}$ that guarantees $\sum_{o \in O} w_o x_o \leq K$, and that maximises the profit $\sum_{o \in O} p_o x_o$.

With the settings

- $T = \{0,.., T_{max}\}$ and $T_{max} = \max\{ p(o)/ w(o) \mid o \in O\}$
- $\forall t \in T$: $C(t) = K$
- $\forall o=(w(o), p(o)) \in O$: $j_o =(a(j_o), e(j_o), d(j_o), m(j_o))$ with
    - earliest starting time $a(j_o) = 0 = s(j_o)$,
    - latest due time $e(j_o) = p(o) / w(o)$,
    - demand for energy $m(j_o) = w(o)$ and
    - duration of the job $d(j_o) = p(o) / w(o)$,

it follows: if x is the mapping of an optimal solution of the TJAP, then $(x_o)_{o\in O} = (x(j_o))_{o\in O}$ describes an optimal solution of the knapsack problem.

3.3 The (more) realistic optimisation problem

In the formalisation TJAP of the job adaptation problem jobs j with x(j)=0 are not scheduled: hence, there is no energy supplied for these jobs and they will never be executed. In a realistic scenario this situation is not acceptable: If the demand exceeds the energy supply, additional energy is bought on the energy market at high costs or produced by expensive, short-term reactive power stations, e.g. gas turbines or reservoir power stations.

To model this situation in our optimisation we scheduled all jobs $j\in J$ and chose as goal function f a measure $\|.\|$ of the distance between energy supply and energy demand:
we introduce an energy demand function $B_{J,s} : T \rightarrow \mathbb{R}$ with $B_{J,s}(t) = \sum_{j\in J,\, s(j) \le t \le s(j) + d(j)} m(j)$ for all $t \in T$

Instead of determining a subset of jobs and optimal starting times, now only optimal starting times for all jobs have to be found in order to minimise the distance between supply and demand.

(1') $x(j) = 1$

(2') $\sum_{t \in T} \| B_{J,s}(t) - C(t) \| \rightarrow \min.$

We choose this formulation of the job adaptation problem (JAP) for further research. Figure 1 shows exemplarily the energy supply curve C(t) (red, square markers), the initial solution $B_0(t)$ (green, triangular markers), and the solution $B_k(t)$ after optimisation (blue, circles).

4. Job Adaptation Problem vs. Scheduling Problems

The Job Adaptation Problem (JAP) means to schedule all jobs for being provided with energy power. Hence, it seems natural to interpret the problem as a scheduling problem.

Scheduling problems are well researched in process planning, e.g. in production planning [5], where a fixed number of resources (machines) can be used for fulfilling a given number of production jobs. Such a job is characterized by its earliest starting time, its duration and its latest due time and its resource demand. Hard constraints describe restrictions which have to be included in a solution schedule while soft constraints may be violated by a solution schedule. The objective is to find an optimal schedule, which fulfils all hard constraints and most soft constraint. Often, the goal function which has to be maximised in order to assess the optimality of a solution is the number of products which can be produced following the given schedule in a given time period. Refined online-scheduling algorithms also handle changes in the planning environment, e.g. they update the number and characteristics of jobs and resources during the execution time of a schedule plan. Many scheduling problems are known for being NP-hard [13].

Although the jobs in the JAP and in scheduling problems are characterized by the same parameters, the JAP cannot be solved in an obvious manner by known scheduling algorithms, as the resource set of scheduling algorithms is always a discrete number of items (i.e. of machines) and the assignment of each job to a resource set is fixed, while the resource set of the JAP is a time-varying amount of energy shared by all scheduled jobs.

Furthermore, we allow that the demand for energy $B_{J,s}(t)$ at time t exceeds the available energy C(t) and focus to minimise the difference between $B_{J,s}(t)$ and C(t), whereas common scheduling algorithms don't allow to exceed the maximum number of available resources.

5. Tabu Search

As a first approach for determining a solution of the given optimisation problem, the well-known 'Tabu Search' was investigated: Tabu Search (see e.g. [8]) is a meta-heuristic: it is superimposed on another heuristic and improves the underlying heuristic for an optimisation problem by allowing temporarily worse solutions on the one hand and inhibiting cycles in the search process on the other hand. The overall idea is to complement the basic heuristic, a local search algorithm, by a so called Tabu List, which stores moves which lead to already inspected solutions. So, if the algorithm starts, the Tabu List is empty and an arbitrary solution, which is probably not optimal, is chosen as current solution. Starting from the current solution, the possible moves to neighboured solutions are evaluated by an objective function. From all moves which are not stored in the Tabu List, the move, which leads to the best solution, is chosen to become the new current solution. So, Tabu Search enforces that local minima of the objective function can be left by means of the Tabu List.
Tabu Search has been adapted for a lot of problems. It has been instantiated with the following settings:

- Neighbourhood generator:

  Depending on the given problem, the neighbourhood of the current solution s must be determined. A solution belongs to the neighbourhood, if it differs only slightly, i.e. by a single move, from the current solution. In our scenario, a neighbour differs from the current solution only by the shift of a single job j, i.e. the actual starting time s(j) in the current solution is increased or decreased by a fixed time unit.

- Objective function:

  The objective function or goal function assesses the quality of solutions in the neighbourhood.
  Let B_k: $T \rightarrow \mathbb{R}$ the demand of solution k. We compared the performance of the following goal functions:
  - City-Block metric: $CB(k) = \sum_{t \in T} |B_k(t) - C(t)|$
  - Euclidean distance: $ED(k) = (\sum_{t \in T} (B_k(t) - C(t))^2)^{½}$
  - As a consequence of results from first experiments, the variant "Euclidean distance with penalty factor f" was added as a third goal function:
    $EDP(k, f) = (\sum_{t \in T,\, B_k(t) \geq C(t)} f \cdot (B_k(t) - C(t))^2 + \sum_{t \in T,\, B_k(t) < C(t)} (B_k(t) - C(t))^2)^{½}$

- Content of the Tabu List:

  As a move is defined by shifting the starting time of one job, we investigated the consequences of storing the job identifier vs. the resulting solution in the Tabu List. The first alternative has the advantage of low storage, but inhibits neighbours which result from further moves of a job. The second alternative needs more storage and effort for comparisons, but inhibits fewer neighbours. We compared the differences between these two alternatives by some experiments.

- Length of the Tabu List (TL):

  Each chosen move or solution is stored in the Tabu List, which has a given maximum length TL. So, when storing the TL+1st solution, the first solution is deleted from the Tabu List and can be chosen again.
  We experimented with different Tabu List length and tried to estimate a good relation between the Tabu List length and the number of jobs.

- Aspiration criterion

  The intention of the Tabu List is to inhibit cycles in the search path by preventing the choice of a move which has been chosen before. Unfortunately, it is possible that the Tabu Lists impedes moves to the best neighbour. In this situation, the aspiration criterion allows to accept neighbours as new current solution if they are better than the best known solution. In our experiments, we tested the use of the aspiration criterion when storing just job identifiers in the Tabu List.

- Number of iterations

  With an increased number of iterations, Tabu Search will find improved solutions. As the globally best solution is not known, we analysed the relation between the number of iterations and the quality of the solution.

6. Experiments

In [11], we set up experiments for assessing the suitability of Tabu Search for determining solutions of the JAP. This means on the one hand to evaluate the various possibilities to instantiate Tabu Search. On the other hand, some test cases had to be generated for benchmarking the different Tabu Search instantiations.

6.1 Instantiation of Tabu Search

Table 1a) shows an overview of the eight different variations of Tabu Search, which we primarily evaluated in order to find the appropriate set up for the job adaptation problem:

Abbreviation	Tabu List Content	Goal function	Aspiration Criterion
JID,CB,+Asp	(JID) Job ID	(CB) City-Block	yes
JID,CB,-Asp	(JID) Job ID	(CB) City-Block	no
JID,ED,+Asp	(JID) Job ID	(ED) Euclidean Distance	yes
JID,ED,-Asp	(JID) Job ID	(ED) Euclidean Distance	no
SID,CB,-Asp	(SID) Solution ID	(CB) City-Block	no
SID,ED,-Asp	(SID) Solution ID	(ED) Euclidean Distance	no

Table 1a: Different instantiations of the Tabu Search method

Taking into account that there are theoretically 2*2*2 = 8 possible combinations, we ruled out all combinations of storing solution identifiers (SID) and using the aspiration criterion. As the storage of solution IDs in the Tabu List means that this solution has once been chosen as new current solution, this solution can never be better than the best solution. Hence the aspiration criterion will not achieve any advancement.

As mentioned before, the Euclidean distance with penalty factors has also been investigated as a goal function. Since in all primary tests job IDs as Tabu List content had a better performance than solution IDs and the City Block metric performed better Euclidean Distance, we restricted to the cases in Table 1b).

JID,EDP,+Asp	(JID) Job ID	(EDP) Euclidean Distance with penalty factors	yes
JID,EDP,-Asp	(JID) Job ID	(EDP) Euclidean Distance with penalty factors	no

Table 1b: Different Instantiations of the Tabu Search method with EDP

We evaluated both instantiation of Tabu Search with two different penalty factors f. So, in summary, we analysed ten different instantiations of Tabu Search.

6.2 Generation of Test Cases

Each instantiation of Tabu Search has been tested in experiments using five test cases which have been generated randomly: Each test case TC consists of tuples ($J_{\#t,z,1}$, .., $J_{\#t,z,5}$, C_z), where $J_{\#t,z,v}$ is a set of Jobs, $|J_{\#t,z,v}| = \#t$, and C_z a possible energy supply over a time interval [0,z]. As these scenarios are randomly generated, we generated five different job sets with parameters #t and z, i.e. $J_{\#t,z,1}$, .., $J_{\#t,z,5}$. When comparing the quality of solutions, the average result for these job sets was taken into account.

Table 2 shows the set of parameters which have been used to set up the five test cases and the parameters of Tabu Search which have been used in the optimisation process:

Test case	Parameters for test case generation		Parameters used in the Tabu Search optimisation		
	#t	length of time interval, z ∈	# of iterations	Tabu List length	penalities for EDP, f ∈
TC_1	20	{30, 40, 60, 70, 80, 90, 100, 110, 120, 130, 150}	{20*i | i=1,..,14}	{5*i | i=1,..,7}	{2, 3}
TC_2	30	{20, 30, 40, 60, 80, 100, 120, 140, 160, 180}	{30*i | i=1,..,26}	{5*i | i=1,..,11}	{2, 5}
TC_3	40	{30, 50, 60, 80, 90, 100}	{30*i | i=1,..,39}	{5*i | i=1,..,13}	{2, 5}
TC_4	60	{30, 40, 60, 80, 100}	{30*i | i=1,..,26}	{5*i | i=1,..,15}	{2, 5}
TC_5	60	{120, 150, 200, 250}	{30*i | i=1,..,49}	{5*i | i=1,..,15}	{2, 5}

Table 2: Overview over the parameters of the test cases

The energy supply function C_z was also generated randomly by determining a time point for the change in the energy supply and the altered amount of supplied energy.

The Tabu Search heuristic was implemented using the Java Tabu Search framework OpenTS [10] which is available under the Common Public License (CPL v0.5). OpenTS allows generic elements for the Tabu List content, for the goal function, the neighbourhood and move definition.

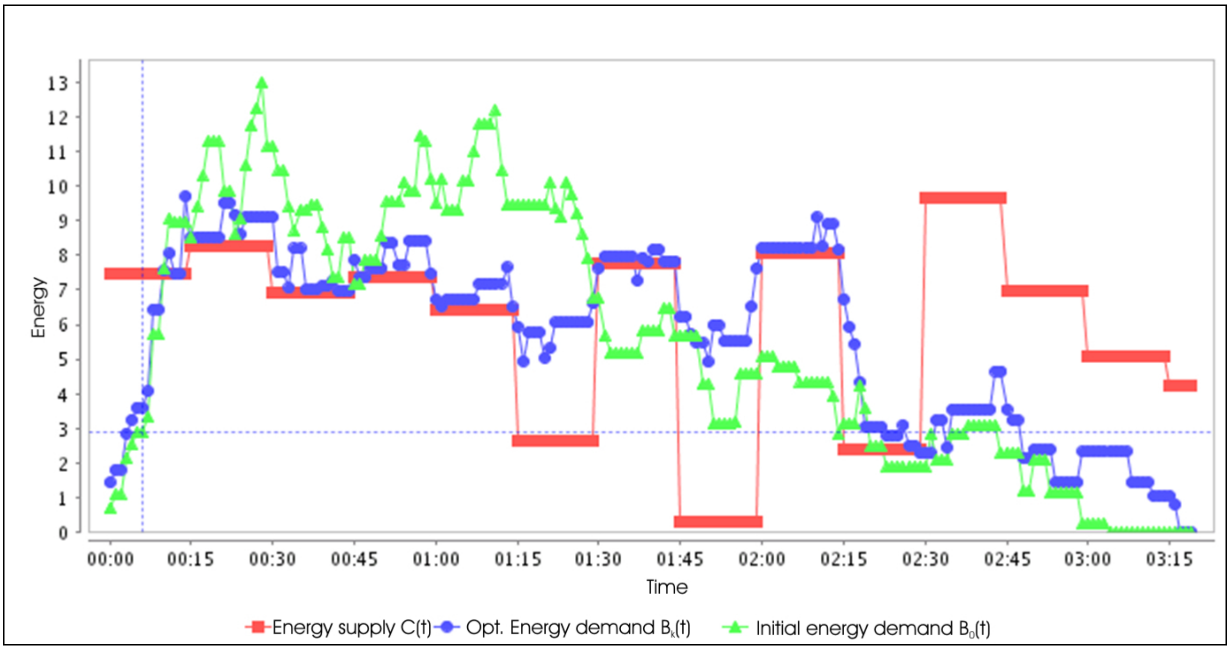

Figure 1: Energy supply and energy demand before ($B_0(t)$) and after optimisation ($B_k(t)$)

7. Results

We evaluated the given test cases for all instantiations of Tabu Search shown in Table 1a) and 1b) [11].
As a criterion for comparing different instantiations, the relative improvement of the solution was used. If different instantiations did not sufficiently differ regarding this primary criterion, the number of iterations for achieving this improvement was taken into account as secondary criterion.

For determining of the best instantiation of Tabu Search for the job adaptation problem we analyzed the influence of the following aspects:

1. content of the Tabu List,
2. application of the aspiration criterion,
3. choice of goal function,
4. usage of penalty factors.

In each step, we compared experiments which only differed in the chosen aspect of consideration. The results gave cause to the following ratings:

- Tabu Search instantiations using job identifiers as Tabu List's content perform better than those using solutions.
- The application of the aspiration criterion leads to better solutions.
- The City Block-metric (CB) as goal function performs slightly better than the Euclidean distance.
- The usage of Euclidean distance (ED) vs. Euclidean distance with penalties (EDP) as goal function leads to solutions of similar quality, but higher penalty factors speed up the convergence to good solutions.

Hence, according to the experiments, the best instantiation of Tabu Search seems to be (CB, JID, +Asp), i.e. using the City Block metric, storage of job IDs in the Tabu List, and using the aspiration criterion.

This instantiation of Tabu Search was additionally assessed regarding the relation between Tabu List length on the one hand, and quality of the solutions depending of the number of iterations on the other hand. Figure 2 shows the experimental results for the test cases TC_2 with 30 jobs (left diagram) and TC_4 with 60 jobs (right diagram). The diagrams show the development of the solutions' quality depending on the number of iterations. The different curves show the performance of the Tabu List length, which is varying between 5 and 65 in TC_2 and between 5 and 75 in TC_4. The comparison shows that solutions reach highest quality, if the Tabu List length is about ½ to ⅔ of the number of jobs. Higher Tabu List lengths are not advantageous, as the corresponding experiments do not lead to solutions with better quality.

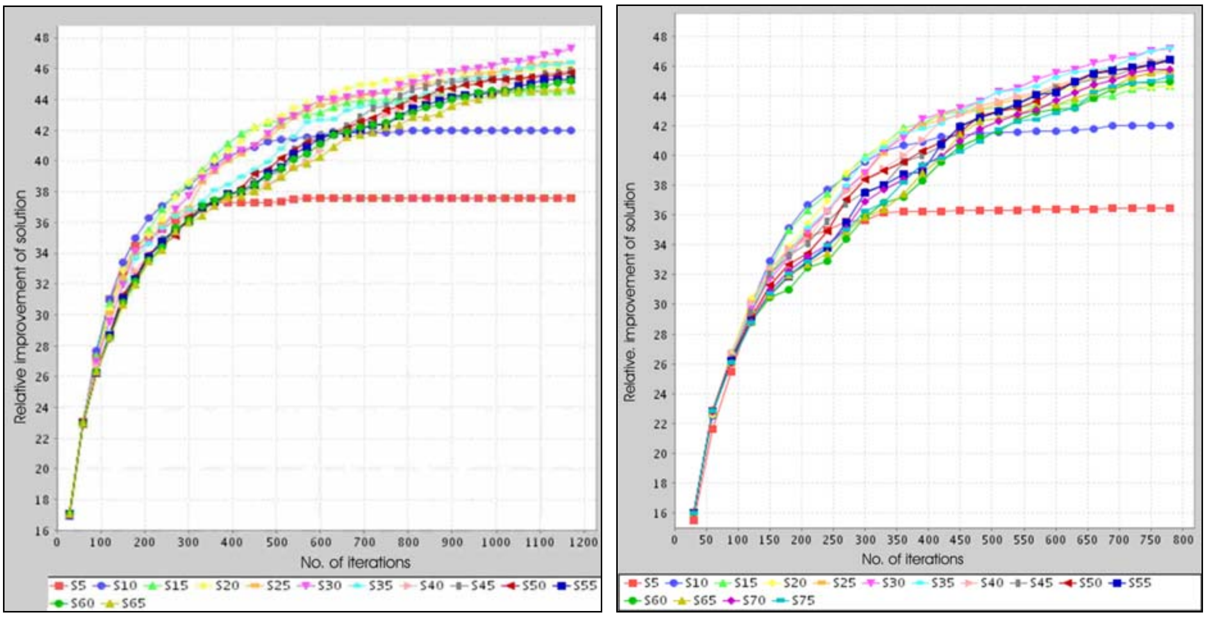

Figure 2: Relation between no. of iterations and quality of a solution for different values of the Tabu List length. (Left diagram: Test case TC_2, right diagram: TC_4)

8. Conclusion and further work

In this paper, we presented our first approach to the adaptation of a set of energy consuming jobs to a given, time varying energy supply curve. We introduced a simplified formal model to this problem which is NP hard. The more refined model was built and the performance of different instantiations of the Tabu Search optimisation method were analysed. The research on Tabu Search has resulted in first insights how to instantiate this heuristic for smaller numbers of jobs.

However, this approach has some drawbacks which we will try to overcome in our further work. First of all, the implemented method was not able to schedule an appropriate number of tasks. We assessed job sets of 20-60 jobs which correspond to the control of some large-scale current consumers. Scenarios which also take the control of households into account address job sets of much higher cardinality.

Further research will mainly focus on two aspects: On the one hand, other heuristics for optimisation, e.g. genetic algorithms, ant colony algorithms, simulated annealing, and the combination of these heuristics, as well as hierarchical approaches acting on successively refined time scales will be applied to the problem. On the other hand, we assume that the optimisation process could profit if the power supply curve is analysed beforehand, so that knowledge about the characteristics of the curve can be used in setting up the optimisation algorithms.

9. References

[1] Born, F: Aiding Renewable Energy Integration through Complementary Demand-Supply Matching. PhD Thesis, Glasgow: University of Strathclyde, 2001

[2] Boyle, G.: Renewable Energy. Oxford University Press, 1996

[3] Cavallo, A. I.: Predicting the Peak in World Oil Production. Natural Resources Research, Vol. 11, No. 3, September, 2002

[4] Crastan, V.: Elektrische Energieversorgung. 2. Auflage, Springer, Berlin, 2004

[5] Domschke, W., Drexel, A.: Einführung in das Operations Research. Springer, Berlin, 2004

[6] Ernst, B., Rohrig, K. and Jursa, R.: Online-Monitoring and Prediction of Wind Power in German Transmission System Operation Centres. First Joint Action Symposium on "Wind Forecasting Techniques", Norrköping, Schweden, 2002

[7] Focken, U., Lange, M., Moennich, K., Waldl, H.-P., Beyer, H.-G. and Luig, A., Short-term prediction of the aggregated power output of wind farms - a statistical analysis of the reduction of the prediction error by spatial smoothing effects. J. Wind Eng. Ind. Aerodyn. 90, 2002, 231-246.

[8] Glover, F. W., and Laguna, M., Tabu Search. Kluwer Acad. Publishers, Boston, 2001.

[9] Kaltschmitt, M., Wiese, A. and Streicher, W.: Erneuerbare Energien. 3. Auflage, Springer, 2003

[10] Harder, R., OpenTS – Java Tabu Search Framework. URL: http://www.coin-or.org/OpenTS/, 30.11.2005

[11] Pei, J., Optimierungsmethoden zur zentralisierten Laststeuerung in Stromversorgungsnetzen mit fluktuierender Einspeisung. Master Thesis, University of Oldenburg, 2005

[12] Penya, Y., Palensky P. and Lobashov, M.: Requirements and Prospects for Consumers of Electrical Energy Regarding Demand Side Management. In: 3. Intern. Energiewirtschaftstagung IEWT'03, TU Wien, Vienna, Austria, 2004

[13] Pinedo, M.: Scheduling – Theory, Algorithms, and Systems. 2nd ed., Prentice Hall, 2002

[14] Stadler, M., Palensky, P., Lorenz, B., Weihs, M. and Roesener, C., Integral Resource Optimization Networks and their Techno-Economic Constraints. Int. Journal of Distributed Energy Resources. Vol. 1, No. 4, 2005, pp. 299-320

[15] Ygge, F., Akkermans, H., Andersson, A., Krejic, M. and Boertjes, E., The HomeBots System and Field Test: A Multi-Commodity Market for Predictive Power Loead Management. In: Proceedings of the PAAM'99, London, UK, 1999, pp. 363-382