Performance of Piecewise Linear Models in MILP Unit Commitment: Difference of Convex vs. J1 Approximation

Felix Birkelbacha,\*, Victor São Paulo Ruelaa,b, René Hofmanna

aTU Wien, Institute of Energy Systems and Thermodynamics, Vienna, Austria

b Tata Steel Ijmuiden, Ceramics Research Centre, Velsen-Noord, The Netherlands

felix.birkelbach@tuwien.ac.at

Abstract

For unit commitment problems, mixed integer linear programming (MILP) is the most widely used solution approach, mainly due to the availability of fast reliable solvers that can handle large problem instances. One challenge with this approach is to incorporate complex non-linear operating behavior in the model. Typically, this is done by employing piecewise-linear models to approximate the non-linear functions and then use specialized MILP formulations to incorporate them in the MILP model.

The most efficient way to incorporate piecewise linear functions with more than two variables in MILP is the logarithmic formulation by Vielma et al. It has been shown to outperform other formulations by a large margin. Even so, it is not routinely used in literature – probably because it puts stringent requirements on the approximation. Specifically, the triangulation has to be compatible with the J1 triangulation.

Recently an algorithm for finding piecewise-linear approximations has been proposed: The difference of convex (DoC) method. It has been shown to achieve high accuracies with only a few linear pieces, thus promising fast performance in MILP. However, DoC approximations are not compatible with the fast Log-formulation.

In this paper, we compare the MILP performance of DoC approximation with J1 approximation. As a use case, we consider a unit commitment problem of the energy system with a thermal storage unit. The two approximation methods are used to model non-linear operating behavior of a thermal storage unit. The solving times for different problem sizes and approximation accuracies are compared and guidelines for achieving the best performance for unit commitment are proposed.

**Keywords**: energy system, unit commitment, MILP, piecewise linear approximation.

* 1. Introduction

Increasingly strict emission targets put a lot of pressure on industry to operate their energy supply systems more efficiently. To make sure that energy demand is covered at all times, the operation of energy system has to be planned accurately. This planning task is usually formulated as a unit commitment (UC) problem. One of the most popular methods to formulate and solve UC problems is mixed integer linear programming (MILP). The huge advances in MILP algorithms in recent years allows it to solve UC problems very quickly (Koch et al., 2022). MILP has the advantage (compared to evolutionary algorithms and heuristic approaches) that it reliably finds the optimal operation trajectory because, as a deterministic optimization method, it provides global optimality guarantees. UC can be used to organize energy supply in a cost-optimal manner and to ensure that energy demand is always met.

To compute reliable operation trajectories, the technical limitations of the units in the energy systems have to be represented accurately. Since the functions that describe the operating behavior are typically non-linear, implementing them in MILP is not straightforward. The operating behavior has to be approximated with a piecewise linear function. Unfortunately, such piecewise linear functions can impact the solving time of the UC problem severely. The reason is that in order to represent a piecewise linear function in MILP, auxiliary variables have to be introduced to distinguish the pieces. Various MILP formulations for piecewise linear functions, which differ in the number of auxiliary variables and constraints, have been proposed (Vielma et al., 2010) and their performance was assessed (Brito et al., 2020; Silva and Camponogara, 2014). Even though all three cited studies found that logarithmic coding outperforms the other formulations by a large margin, it is not routinely used – probably because it puts stringent requirements on the approximation. Specifically, that the triangulation has to be compatible with the J1 triangulation.

Finding an accurate piecewise linear approximation with the fewest number of linear pieces is a challenge in itself, on par with solving the UC problem. The challenge is to divide the domain of the function into pieces and to assign a linear function to each of them so that the difference between the approximation and the function is minimal. Basic linear elements are referred to as simplices (triangles in 2 dimensions). The naïve approach to compute a piecewise linear approximation is to cover the domain of the function with a grid, which is in turn used to construct a triangulation. Unfortunately, this approach generally requires a lot of linear pieces to reach the desired accuracy and thus the MILP performance will be poor. For that reason, specialized algorithms have been proposed. One approach revolves around recursively subdividing triangles into smaller triangles until the target accuracy is reached (Rebennack and Kallrath, 2015), another uses iterative mesh refinement (Obermeier et al., 2021). Even though both approaches can find good piecewise linear approximations, neither produces approximations that are compatible with logarithmic coding. Recently, we compared the J1 approach with the one by Rebennack & Kallrath in a UC problem and found that, even if approximation with J1 needs more linear pieces, the solving time of the UC problem is only a fraction thanks to compatibility with logarithmic coding (Birkelbach et al., 2023). For that reason, we developed a method to compute J1 compatible approximations and thus allows for logarithmic coding (Birkelbach, 2024).

Recently, another interesting approach to achieve faster solving times has been proposed: the difference of convex (DoC) approach (Kazda and Li, 2021). They use polyhedra (polygons) instead of simplices (triangles) to significantly reduce the number of linear pieces to reach the target accuracy. They demonstrated their approach by comparing it with the one by Rebennack & Kallrath. Because of the impressive results, also faster solving times can be expected compared to general simplex models. The key question now is: Does approximation with polyhedra or a J1 triangulation yield the fastest solving time in a UC problem?

In this paper, we compare the DoC approach with the J1 approach by applying them to a use case from one of our projects. We briefly introduce the relevant background on MILP formulations and the approximation methods. Then we discuss the effect that the various combinations of approximation and MILP formulation have on the UC problem. Finally, we assess the solving time of the UC problem and propose some recommendations to achieve the fastest solving time with piece-wise linear models.

* 1. Methods

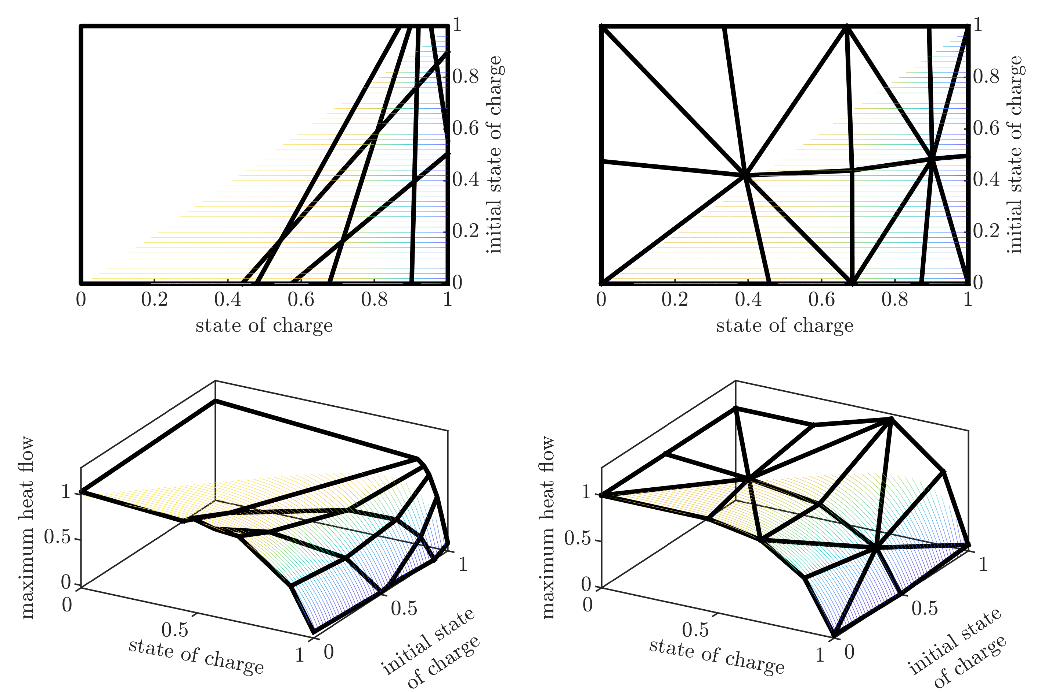


Figure 1 Illustration of the piecewise linear approximation of the function that describes the maximum charging rate of the heat storage with a target RMSE of 0.02 (axes are normalized). Difference of convex (left), J1 triangulation (right). Top view (top), isometric view (bottom).

In this section, we will briefly introduce the methods that were used in this study. Where a detailed discussion is not possible due to space restrictions, we will provide references.

* + 1. MILP formulations for piecewise linear functions

In their seminal paper “Mixed-Integer Models for Nonseparable Piecewise-Linear Optimization” (Vielma et al., 2010) discuss and analyze six different formulations for representing piecewise linear functions in MILP. For the present study, three of these formulations are relevant: Multiple Choice (MC), disaggregated convex combination with logarithmic coding (DLog) and the logarithmic branching convex combination (Log). The three key differences are: 1) the number of auxiliary binary variables grows linearly for MC and logarithmically for DLog and Log. 2) the number of auxiliary continuous variables increases considerably with the number of linear pieces for DLog than for MC and Log. 3) MC and DLog can be used with any type of convex polytope, while Log requires a J1 triangulation.

Performance studies showed that for functions with only a few linear pieces, MC yields the best performance, while DLog and Log have an advantage if the function is comprised of many linear pieces. In general, Log outperformed DLog. (Vielma et al., 2010)

* + 1. Piecewise linear approximation

The goal of piecewise linear approximation for MILP is to find a piecewise linear function that fulfills the accuracy requirements, typically in terms of the root mean square error (RMSE) or the maximum absolute error (MAE), and that will yield the best possible solving time for the MILP problem. In general this means that the approximation should require as few linear pieces as possible, but compatibility with efficient MILP formulation does also play a role.

The difference of convex (DoC) method (Kazda and Li, 2021) finds a piecewise linear approximation with an heuristic using the difference of two convex functions to represent any non-convex function. The resulting linear pieces are polyhedra, i.e. polygons in two dimensions (see Figure 1, left). They showed that DoC produces approximations with significantly fewer linear pieces than the method by Kallrath and Rebennack. To represent these approximations in MILP, either the MC or DLog can be used.

To compute piecewise linear approximations that are compatible with the Log formulation, we used the algorithm that we published recently (Birkelbach, 2024). It uses a combination of a gradient based optimizer with a heuristic to find a good approximation. Figure 1 (right) shows such an approximation where the typical “union jack” pattern is clearly visible. This approximation is compatible with all MILP formulations proposed in (Vielma et al., 2010). In this study we consider MC and Log. DLog was omitted because it is outperformed by Log (Vielma et al., 2010).

* 1. Use Case

For evaluating the performance of each modeling approach and the effect of the MILP formulation, we used the same UC model as (Koller et al., 2019). In this UC model, a very simple energy system (see sketch in Figure 2) consisting of a generating unit (GU) and a packed bed regenerator (PBR) as heat storage unit. The GU produces both heat and electricity. Electricity is sold at the electricity market at a fluctuating but known price. To allow the GU to shut down during times of low electricity prices, the PBR is used to store heat and supply it later. The UC model has a prediction horizon of 8 days with time steps of 1 hour. This results in 192 time steps. More details on the model in (Koller et al., 2019).

The typical output of the UC problem is shown in Figure 2. The diagram on the top shows how the heat demand is covered by either a heat flow from the GU or from the PBR. The fluctuating electricity price that makes operation of the GU uneconomic at some times is also shown. In the bottom diagram, the operating trajectory of the PBR is shown. The bars illustrate heat flow to and from the storage. The line shows the state of charge.

To model the behavior of the PBR in the UC model comprehensively, the dependence of the maximum charging and discharging power on the state of charge has to be considered. The maximum power diminishes, when the storage is almost fully charged or discharged, because the power depends on the position of the thermocline inside the packed bed. When the storage operates in partial charging and discharging cycles (which will usually be the case), the state of charge at the time of switching also has to be considered. (Koller et al., 2019)

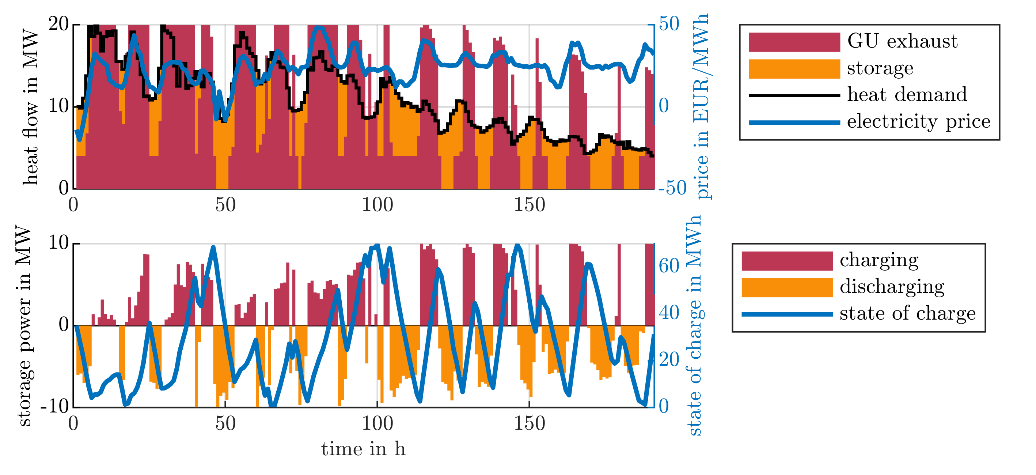


Figure 2 Illustration of the output of the unit commitment problem.

* 1. Results

The UC problem and the DoC algorithm were formulated using YALMIP R20210331 (Löfberg, 2004) in Matlab R2023b. The problems were solved using Gurobi 10.0.0 on a 128-core system (AMD EPYC 7702P) with 256 GB RAM. The J1 approximations were computed using Matlab’s optimization toolbox.

For our use case, we used a simulation model to generate a data set that gives the maximum charging/discharging rate of the storage for various initial states of charge (colored dots in Figure 1). We then approximated this dataset with both the DoC and the J1 method. As accuracy metric we chose the root mean square error (RMSE) and performed the approximation with three target accuracies 0.08, 0.04 and 0.02. Table 1 shows number of linear pieces that each method requires to approximate both the charging and discharging behavior with the target accuracy. The results show, that the DoC method indeed requires fewer linear pieces than the triangulation approach (see Figure 4, top).

Table 1 also shows the number of auxiliary variables that are required to implement each of the approximations in MILP depending on the MILP formulation. The number of binary variables can be reduced significantly with both logarithmic methods especially for the detailed approximations. The Log method stands out, because it reduces both the number of binary and continuous variables.

In the next step, we used each approximation in the UC problem and measured the time it took to solve the model to an optimality gap of 10-2. The results are shown in the last column in Table 1. For RMSE 0.08, DoC performs significantly better than J1 because it can reach the target accuracy with very few linear pieces. (So few actually that some of the auxiliary variables can be eliminated during pre-solve). Starting at moderate accuracies, the J1 method starts to outperform the DoC method, even though it uses more linear pieces. At an RMSE of 0.02 the UC problem with the J1 approximation is three times faster than the one with DoC.

Table 1 Performance metrics of the two approximation methods. Best performing method for each accuracy is highlighted bold.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Target RMSE | Number of linear pieces | MILP formulation | Number of binary variables | Number of continuous variables | Solving time of UC in s |
| DoC | 0.08 | 4 | MC | 4 | 8 | **0.7283** |
| DLog | 2 | 16 | 2.5547 |
| 0.04 | 12 | MC | 12 | 24 | 13.0714 |
| DLog | 6 | 48 | 35.2150 |
| 0.02 | 26 | MC | 26 | 52 | 103.0004 |
| DLog | 8 | 104 | 416.1805 |
| J1 | 0.08 | 10 | MC | 10 | 20 | 12.6132 |
| Log | 5 | 14 | 4.4169 |
| 0.04 | 20 | MC | 20 | 40 | 131.5463 |
| Log | 7 | 22 | **5.7963** |
| 0.02 | 32 | MC | 32 | 64 | 551.9540 |
| Log | 8 | 30 | **36.6738** |

* 1. Conclusions

We studied the performance of two methods to compute piecewise linear approximations of non-linear functions for MILP. While the difference of convex (DoC) approach is capable of finding approximations with fewer linear pieces than the J1 approach, this advantage is in large parts offset by effect of the MILP formulation. While the DoC yielded the best performance with the crude approximation, the J1 approach performed significantly better for the more detailed ones.

The results also show that –at least in the accuracy range that we studied– DoC does not experience speedup with logarithmic coding (DLog); MC was the best choice throughout. The reason is that MC requires much fewer auxiliary continuous variables than DLog. For the J1 approach, on the other hand, the Log formulations always performed better.

To summarize, our results suggest as a general rule of thumb that DoC is the best choice for problem instances where a few linear regions suffice to reach the target accuracy. DoC models should be paired with the MC formulation. In case more than a few linear pieces are required, J1 approximation is the method of choice. Even if it needs a few more linear pieces, this is outweighed by its compatibility with the Log formulation.

References

Birkelbach F. Piecewise linear approximation for MILP using J1 compatible triangulations [in preparation] 2024.

Birkelbach F, Kasper L, Schwarzmayr P, Hofmann R. Operation Planning with Thermal Storage Units Using MILP: Comparison of Heuristics for Approximating Non-Linear Operating Behavior. Proc. ECOS 2023, Las Palmas De Gran Canaria, Spain: 2023, p. 1345–50. https://doi.org/10.52202/069564-0122.

Brito BH, Finardi EC, Takigawa FYK. Mixed-integer nonseparable piecewise linear models for the hydropower production function in the Unit Commitment problem. Electr Power Syst Res 2020;182:106234. https://doi.org/10.1016/j.epsr.2020.106234.

Kazda K, Li X. Nonconvex multivariate piecewise-linear fitting using the difference-of-convex representation. Comput Chem Eng 2021;150:107310. https://doi.org/10.1016/j.compchemeng.2021.107310.

Koch T, Berthold T, Pedersen J, Vanaret C. Progress in mathematical programming solvers from 2001 to 2020. EURO J Comput Optim 2022;10:100031. https://doi.org/10.1016/j.ejco.2022.100031.

Koller M, Hofmann R, Walter H. MILP model for a packed bed sensible thermal energy storage. Comput Chem Eng 2019;125:40–53. https://doi.org/10.1016/j.compchemeng.2019.03.007.

Löfberg J. YALMIP : a toolbox for modeling and optimization in MATLAB. 2004 IEEE Int. Conf. Robot. Autom. IEEE Cat No04CH37508, Taipei, Taiwan: IEEE; 2004, p. 284–9. https://doi.org/10.1109/CACSD.2004.1393890.

Obermeier A, Vollmer N, Windmeier C, Esche E, Repke J-U. Generation of linear-based surrogate models from non-linear functional relationships for use in scheduling formulation. Comput Chem Eng 2021;146:107203. https://doi.org/10.1016/j.compchemeng.2020.107203.

Rebennack S, Kallrath J. Continuous Piecewise Linear Delta-Approximations for Bivariate and Multivariate Functions. J Optim Theory Appl 2015;167:102–17. https://doi.org/10.1007/s10957-014-0688-2.

Silva TL, Camponogara E. A computational analysis of multidimensional piecewise-linear models with applications to oil production optimization. Eur J Oper Res 2014;232:630–42. https://doi.org/10.1016/j.ejor.2013.07.040.

Vielma JP, Ahmed S, Nemhauser G. Mixed-Integer Models for Nonseparable Piecewise-Linear Optimization: Unifying Framework and Extensions. Oper Res 2010;58:303–15. https://doi.org/10.1287/opre.1090.0721.