A Comparison of Different Many-Objective Optimization Algorithms for Energy System Optimization

Tobias Rodemann

Honda Research Institute Europe
Carl-Legien-Strasse 30
63073 Offenbach/ Main
Germany
tobias.rodemann@honda-ri.de

Abstract. The usage of renewable energy sources, storage devices, and flexible loads has the potential to greatly improve the overall efficiency of a building complex or factory. However, one needs to consider a multitude of upgrade options and several performance criteria. We therefore formulated this task as a many-objective optimization problem with 10 design parameters and 5 objectives (investment cost, yearly energy costs, CO_2 emissions, system resilience, and battery lifetime). Our target was to investigate the variations in the outputs of different optimization algorithms. For this we tested several many-objective optimization algorithms in terms of their hypervolume performance and the practical relevance of their results. We found substantial performance variations between the algorithms, both regarding hypervolume and in the basic distribution of solutions in objective space. Also the concept of desirabilities was employed to better visualize and assess the quality of solutions found.

Keywords: Many-Objective Optimization, Energy Management, Desirabilities

1 Introduction

Many-objective optimization algorithms are a hot topic in the field of evolutionary computation, due to both increasing interest from the application side as well as a number of unsolved issues. In this work we will evaluate the performance of several well-known many-objective optimization (MAO) algorithms on a challenging real-world problem: finding the optimal configuration of the energy system of a heterogeneous business building complex. The usage of local energy production and storage facilities has become increasingly interesting both in terms of energy costs and CO_2 emissions. Facility management is therefore looking how to optimally invest in extensions to the current building energy system. Example extensions are large-scale Photo Voltaic (PV) systems or battery storage capacity. There is typically a large number of options for adding

new modules or optimizing the usage of existing ones. With increasing system complexity the number of objectives to be considered increases, resulting in a difficult optimization problem for the decision makers, especially considering the high investment costs. One has to consider that different modules might be linked, for example optimal battery and PV system size. A serious problem for both optimization and analysis of solutions is the wide range of objective values, which can vary over many orders of magnitude. In order to deal with the latter issue we propose to use desirability functions as discussed in [1].

There is a number of recent works targeting many-objective optimization of energy management systems [2, 3]. For single objective optimization of energy systems, methods from the field of Linear Programming (LP) are commonly used since they are comparatively fast and provide performance guarantees [4, 5]. However, LP approaches are based on simplified system models where many aspects are difficult to consider such as battery aging, temperature-dependent efficiencies, or dynamic user behavior. For this reason we employ a very detailed building simulation based on the Modelica standard [6, 7] that can model most real-world effects. The simulator is treated as a black box by the optimization, meaning that we do not make any assumptions about the structure of the problem. The ability of evolutionary algorithms to handle these types of problems is one of their greatest assets. Please consult Fig. 1 for a view of the simulation environment with our building model and Fig. 2 for a sketch of the information flow from optimizer to simulator and back.

We will first introduce the specific application example. Afterwards we are going to sketch the optimization algorithms used in this paper and the experimental setting. We will then show results of different MAO algorithms using the hypervolume indicator. Furthermore, we will compare results found by the optimization with manually selected baseline results. Finally, we summarize our results and present some future work.

Note that our target is not a ranking of optimizers, but a better understanding if, from a practical point of view, the use of different optimizers for this specific problem is necessary.

An initial study on this application was already published earlier [8]. In the present work we compare a much larger variety of optimization algorithms since prior experiments suggested that some optimizers struggled with this application. We furthermore used findings from this earlier work to adapt the optimization problem (different parameter ranges), added new parameters and a new objective to include battery aging.

In order to capture essential seasonal effects one would normally consider at least one full year but to be able to perform more test runs we reduced the simulated time from one year to a single month. This could in practice lead to sub-optimal performance of the selected configuration, but allows us to thoroughly compare different optimizers. In the real application case, one would obviously revert to the simulation of at least one year using one of the better performing algorithms as identified by this study.

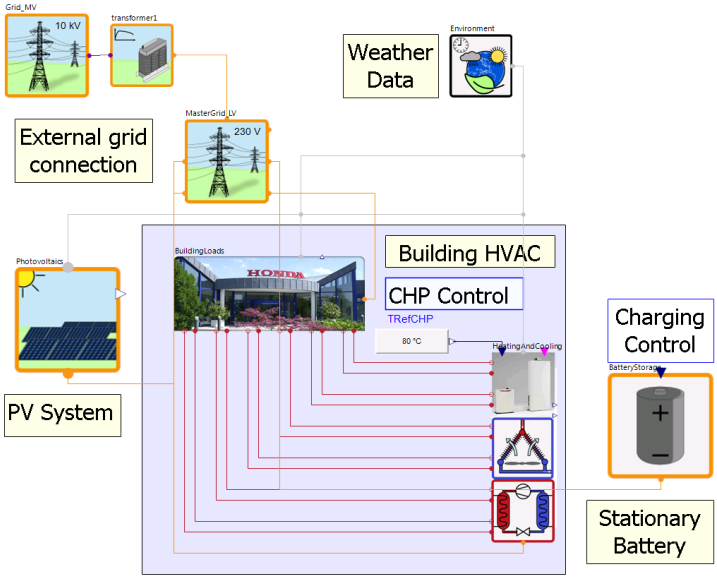

Fig. 1. System view of building simulator. The model contains a detailed simulation of the building's Heating-Ventilation-A/C system, including heat storage and co-generator (CHP), plus modules for Photo Voltaics (PV), grid connection, and a stationary battery including corresponding controller.

1.1 Overview

The target of the optimization is to find optimal configurations for the energy layout of a medium-sized research campus with about 200 employees. With configuration we mean a combination of different modules like PV system, battery or heat storage, and settings of a controller.

The building has a rather conventional load profile with peak loads above 500 kW mostly around noon, due to HVAC (heating / ventilation / air -conditioning) demands and a baseload of about 200 kW. Total annual energy costs for gas and electricity are in the range of several 100,000 Euros. The campus is already equipped with a co-generator for heat and power (CHP) that provides 200 kW electric and 300 kW thermal output. Further improvement options have been identified and are described below.

1.2 Investment Options

There are several promising areas for investment that define the search space for the configuration optimization presented here:

1. A large-scale PV system on the building roof or car-port

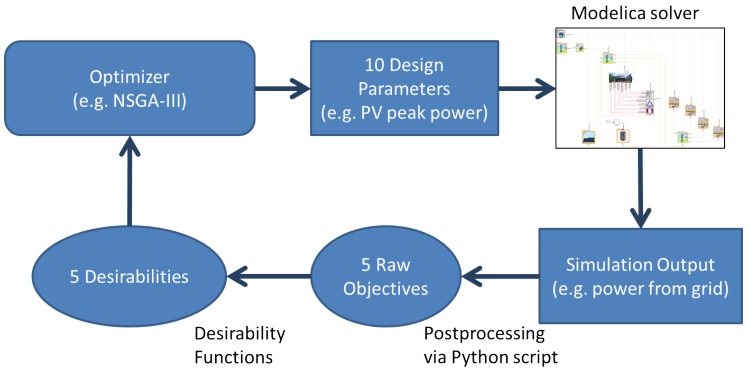

Fig. 2. Overview of optimization loop.

2. An extension of the internal heat storage
3. A stationary battery with a control algorithm that charges the battery in times of low energy demand and discharges when energy demand is high
4. Optimization of the operation of the CHP

1.3 Design Parameters

Based on the above defined investment options we decided to consider in total 10 independent parameters:

1. Inclination angle α_{PV}, orientation angle β_{PV}, and peak output power $PPeak_{PV}$ (in kW) of PV system.
2. Capacity E_{Batt} (in kWh), and linked to that maximum charging/discharging power P_{batt}^{max} (in kW), of a stationary battery storage system, where we assume a maximum charging power of 1 C (i.e. battery fully charged or discharged within one hour).
3. Min and max battery state-of-charge (SOC) level SOC_{min} and SOC_{max}.
4. Charging u_{Batt}^{charge} and discharging $u_{Batt}^{discharge}$ threshold for the battery controller. The battery will discharge when current load exceeds $u_{Batt}^{discharge}$, and correspondingly charge when demand is below u_{Batt}^{charge}. The battery will charge or discharge to keep the load below the given charging threshold level (as long as the battery is within its SOC limits and P_{batt}^{max} allows). The charging controller is visualized in Fig. 3.
5. Total volume of heat storage in m^3, V_{stor}. From this value we also derive the diameter of the heat storage d_{stor} as $d_{stor} = \sqrt{V_{stor}/3}$.
6. Operation threshold of the CHP system. The generator will only turn on if ambient temperature is below this level. This was implemented to avoid too frequent on/off switches (resulting in high maintenance costs) and is realized by using an upper (u_{CHP}^{high}) and lower (u_{CHP}^{low}) temperature threshold with a

difference of 1.0 degree as $u_{CHP}^{low} = u_{CHP}^{high} - 1.0$. The CHP is turned off when ambient temperature exceeds u_{CHP}^{high} and restarted when it falls below u_{CHP}^{low}.

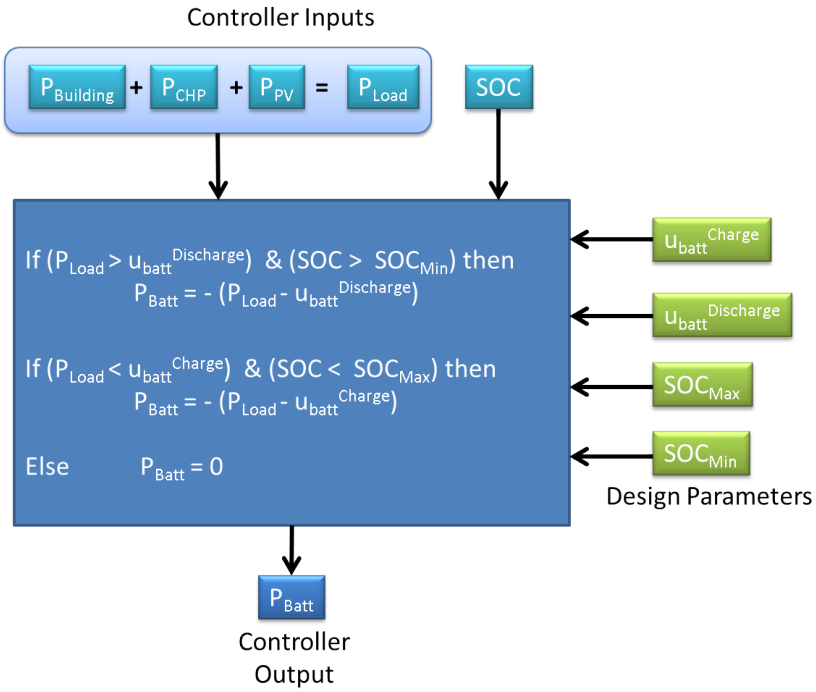

Fig. 3. Controller logics for stationary battery. The main controller inputs are the state-of-charge (SOC) of the stationary battery and the sum of building energy consumption and energy produced by CHP and PV (P_{CHP} and P_{PV}) (note that produced energy is by convention negative). The controller output is the charging power of the battery P_{Batt}.

Please consult Tab. 1 for a list of minimum and maximum values for all design parameters. Those ranges have been defined in cooperation with a building energy specialist considering actual physical constraints (e.g. available space) of the building.

1.4 Objectives

The quality of an investment solution will depend on a number of factors. We have chosen the following five objectives which cover the main factors in the respective domains. Values are computed based on outputs of the simulator.

Parameter	Min	Max	Parameter	Min	Max
α_{PV}	0.0°	45°	β_{PV}	120.0°	250°
$PPeak_{PV}$	0 kW	450 kW	E_{Batt}	5 kWh	400 kWh
SOC_{min}	0%	40%	SOC_{max}	50%	95%
u_{Batt}^{charge}	-500 kW	0 kW	$u_{Batt}^{discharge}$	300 kW	700 kW
V_{stor}	1 m^3	5 m^3	u_{CHP}^{high}	10°C	25°C

Table 1. Parameters and their ranges.

1. Initial investment cost. We limit ourselves to the main (hardware) purchasing costs: PV system (as a function of peak power) $C_{PV} = 1000$ Euro $\cdot PPeak_{PV}$ (in kW), battery (total capacity) $C_{Batt} = 250$ Euro $\cdot E_{Batt}$ (in kWh), and finally heat storage (volume) $C_{HeatStor} = 700$ Euro $\cdot V_{stor}$ (in m^3). Total investment cost is the sum of all module costs (in Euro):

$$C_{Invest} = C_{PV} + C_{Batt} + C_{HeatStor} \quad . \tag{1}$$

2. Annual operation cost. This is the sum of grid electricity cost (C_{GridE}), gas consumption (C_{Gas}) from CHP and conventional boilers, peak electricity load (from the grid) fee, CHP maintenance cost C_{CHP} (which is proportional to the total operation hours h_{CHP}: $C_{CHP} = 4.3$ Euro $\cdot h_{CHP}$) minus CHP subsidies S_{CHP} of 4.35 ct per produced kWh (assuming very high self-consumption rates) and reimbursements for feeding excess electricity back to grid (C_{FeedIn}). Total annual cost is then (in Euro):

$$C_{Annual} = C_{Grid} - C_{FeedIn} + C_{Gas} + C_{Peak} + C_{CHP} - S_{CHP} \quad . \tag{2}$$

   For gas we assume a price of $c_{Gas} = 2.5$ ct/kWh (thermal), grid electricity costs of $c_{El} = 13.0$ ct/kWh, and a feed-in tariff of $c_{FeedIn} = 7$ ct/kWh. The peak load cost is 76 Euros for each kW at the highest energy load (averaged over a 15 min interval) within the whole calendar year. All consumption values are provided by the simulation tool.
3. Annual CO_2 emissions. Reducing CO_2 emissions is a high-priority task. We compute the combined CO_2 from purchasing grid electricity $emis_{grid}$ (at 500 g/ kWh, approximated data for Germany) and gas $emis_{gas}$ (for CHP and boilers, at 185 g/kWh thermal), based on simulated electricity and gas consumption. Total CO_2 emission is then:

$$emis_{total} = emis_{grid} + emis_{gas} \quad . \tag{3}$$

4. Resilience. The availability of local energy production and electric storage capacity would allow a system operation even in case of severe grid malfunctions. This ability is termed resilience and in our specific case refers to the duration the company could operate if no grid power is available. Specifically we are using the minimum of battery charging level $BatLevel$ (in kWh) divided by current grid load E_{Load} (in kW), over the complete simulation period. This represents a worst case scenario (grid failure occurs at

worst possible time). Obviously, one could also consider other formulations, like mean instead of min, or using a fixed "emergency" power demand. The resulting resilience (in seconds) in our case was computed as:

$$resi = \min\left(\frac{BatLevel}{E_{Load}}\right) \cdot 3600 \tag{4}$$

5. Battery lifetime. We are using a coarse model of battery aging based on calendaric and cyclic aging. Please note that the quantitative results should be viewed with care since battery aging strongly depends on the underlying battery chemistry. In this work a generic Li-Ion battery was used. The simulation tool will compute the final battery State-Of-Health (SOH_{1m}) after a one month of operation, which represents the remaining battery capacity (relative to the initial capacity). Battery lifetime (in years) will then be computed as the number of years before state of health falls to $\theta_{SOH} = 90\%$:

$$\text{lifetime} = \frac{1}{12}\frac{\log(\theta_{SOH})}{\log(SOH_{1m})} \tag{5}$$

The first three objectives (investment cost, annual cost, CO_2 emissions) are minimized, while resilience and battery lifetime are maximized. Results for yearly costs and emissions are approximated by simply scaling up all values from the monthly simulation by a factor 12.

1.5 Desirabilities

With 5 different objectives that vary within different ranges, visualization of solutions becomes a challenge. In order to avoid various scaling operations at different stages of the optimization process, we decided to employ the concept of desirability functions [1, 9]. The desirability $d(Y)$ for an objective Y is given by:

$$d(Y) = \exp\left(-\exp\left(-\left(b_0 + b_1 Y\right)\right)\right) \tag{6}$$

The closer the desirability score to one, the more satisfying the quality of the objective value. For controlling the shape of the desirability function, we define two pairs of values to compute the parameters b_0 and b_1. These are two example objective values ($Y^{(1)}$ and $Y^{(2)}$) and corresponding desirability scores ($d^{(1)}$ and $d^{(2)}$), for details see e.g. [1].

$$b_0 = \quad -\log(-\log(d^{(1)})) - b_1 Y^{(1)} \tag{7}$$

$$b_1 = \frac{-\log(-\log(d^{(2)})) + \log(-\log(d^{(1)}))}{Y^{(2)} - Y^{(1)}} \tag{8}$$

Specific values for (d, Y) pairs are given in Tab. 2. In practice one would first discuss with the actual decision maker about her preferences and then adapt the

desirability functions accordingly. Here the selection was based on the baseline configurations described in the next section. To formulate the optimization as a minimization problem we internally used $1 - d(Y)$ as objective values, i.e. 0.0 represents the optimal value and 1.0 the worst possible solution, which is additionally used as the reference point for the hypervolume computation with an $\epsilon = 10^{-5}$ added.

Objective	$d^{(1)}$	$Y^{(1)}$	$d^{(2)}$	$Y^{(2)}$)
C_{Invest} (Euro)	0.9	100,000	0.1	600,000
C_{Annual} (Euro)	0.9	350,000	0.1	400,000
$emis_{total}$ (t)	0.9	2,100	0.1	2,200
$resi$ (s)	0.1	1	0.9	900
$Lifetime$ (y)	0.1	10	0.9	30

Table 2. Control points for desirability function for different objectives.

1.6 Simulation System

Since we want to investigate a variety of different aspects of the energy system we need a simulation tool that can handle various physical domains (electricity, heat, cold, e-mobility among others). We therefore decided to use the Modelica [6] simulation language that is based on physical equations and well suited for simulation of complex systems including non-linear effects. The model was built based on an analysis of the real building and smart meter measurements of energy consumption over several years. The simulator provides all required output information like energy consumption or battery charge levels. A typical simulation now runs for approximately 10s, but some configurations require far more than this. We therefore decided to stop any simulation that runs longer than a threshold value of 20 s and assign worst possible values for all objectives. See [8] for a discussion of this issue.

2 Optimization Task

Our application instance is a many-objective optimization problem (MaOP) with 10 parameters and 5 objectives. Prior work (e.g. [10]) investigated several algorithms for the optimization of a hybrid car controller and found that they exhibit different performance qualities. We now want to compare several optimization algorithms of different types on our test application. Due to the long simulation (fitness computation) times, we performed all tests on a computing cluster, allowing us to run all solutions of a single generation in parallel. For all algorithms we use a population size of $\mu = 35$ individuals. The total number of objective function calls was limited at 5250. All experiments were repeated 10

times with different random number seeds. Design parameters were hard limited to the range specified above. The software we employ for the optimizers is PlatEMO V1.5 [11]. All hyperparameters were chosen as the default values from the above software, since an extensive hyperparameter scan (as for example via irace [12]) is way too time-consuming. Please also consult [13] for some thoughts on selecting the proper optimizer for compute-intensive application problems.

We used the following algorithms to cover a broad range of approaches, which span the range from 2001 - 2016 as the year of publication: NSGA-II [14], NSGA-III [15], RVEA [16], IBEA [17], SPEA [18], KnEA [19], GDE3 [20], MOPSO [21], PICEAg [22], TwoArch2 [23].

Note that there are 77 algorithms within PlatEMO V1.5, so that any selection is to a substantial degree arbitrary. It is not our target to benchmark these algorithms but to evaluate if there is a practical difference in performance between different optimizers. As far as possible, evaluation of solutions was parallelized. As this is not trivially possible for steady-state algorithms (like [24, 25]), those were not tested. Average run-times for all algorithms (a single (out of 10) runs for one specific optimizer) are around two hours but would be around 24 hours for steady state algorithms.

3 Results

Due to the large run-times we only used default settings for all optimizers and kept population sizes for all optimizers at the same value (35 individuals). Since we do not know the true Pareto front we are using the hypervolume as the main performance indicator.

3.1 Baseline Solutions

For a realistic performance assessment in addition to hypervolume values we compare results for our optimized configurations with some manually chosen baseline configurations. The results (objectives) of these basic configurations were also used to determine reasonable values for the parameters of the desirability function. Please note that actual results for the explicit simulation of a complete year are substantially different (mostly better) compared to the values shown here, which were extrapolated from a single month.

Current Configuration The baseline configuration of the building has a 10 kW peak PV system, no stationary battery, a moderate heat storage of 1.3 m^3 and a CHP threshold setting of 17 $°C$. The corresponding parameter values and the resulting objectives are found in Tab. 3 under the row *Current*. Investment cost is minimal but yearly costs and CO_2 emissions are high, and the system does not have any resilience capacity.

Moderate Expansion A moderate expansion configuration is simulated by setting PV size, battery capacity, and heat storage volume to intermediate values ((Max-Min)/2). The battery is used at a range from 20 - 80% capacity, it is charged when energy production exceeds demand ($u_{Batt}^{charge} = 0$ kW) and discharges when energy demand is above 600 kW ($u_{Batt}^{discharge} = 600$ kW). Inclination and orientation angle of the PV system and CHP threshold are the same as for the current configuration. Parameters and objectives can be found in row *Moderate*. We see that with an investment of 280,000 Euros, a reduction of annual costs by around 9,500 Euro, and CO_2 emissions by 34 tons / year is possible compared to the current solution. In addition the system now has a resilience, being able to operate on its own for 450 s in case of a grid failure. With the current desirability settings this value is considered sufficient.

Full Set-up The third configuration we analyze represents a maximum investment with the largest possible size of PV system, battery, and heat storage. The remaining parameters are kept as before, except for battery SOC settings which we set to the range 5 - 95% and a more ambitious battery discharge threshold of 500 kW. It is interesting to see that the effects are almost linear (roughly double investment costs and double savings on annual costs and emissions). This indicates that the larger PV system can be fully used.

Parameter	$PPeak_{PV}$	E_{Batt}	u_{Batt}^{charge}	$u_{Batt}^{discharge}$	SOC_{min}	SOC_{max}	V_{stor}
Current	10 kWp	0	—	—	—	—	1.3 m^3
Moderate	230 kWp	200 kWh	0 kW	600 kW	20 %	80 %	3 m^3
Full	450 kWp	400 kWh	0 kW	500 kW	5%	95%	5 m^3

Objective	C_{Invest}	C_{Annual}	$emis_{total}$	$resi$	lifetime
Current	0.0 (1.0)	382285 Euro (0.46)	2202 t (0.08)	0.0 s (0.0)	—
Moderate	281185 Euro (0.72)	372735 Euro (0.65)	2168 t (0.42)	449 s (0.61)	26.6 y (0.75)
Full	550665 Euro (0.18)	363517 Euro (0.78)	2140 t (0.69)	1223 s (0.96)	26.5 y (0.73)

Table 3. Parameters (*top*) and objective values (*bottom*) of different alternative configurations. For all configurations $\alpha_{PV} = 35°$, $\beta_{PV} = 180°$,$u_{high} = 17°C$ is used. Desirabilities are shown in brackets.

3.2 Optimizer Results

Now we take a look at the results of the various optimization runs. Our main quality indicator is the maximum hypervolume of the parent population in the final generation. All hypervolume values are computed in desirability space. For a better comparison of hypervolume values we present final hypervolume results in a boxplot, see Fig. 4.

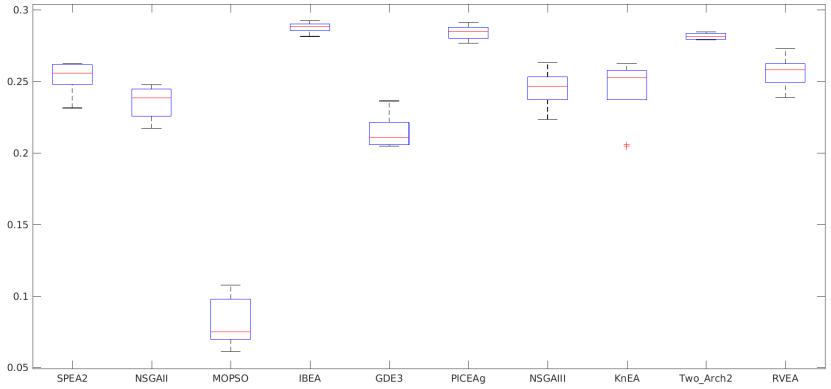

Fig. 4. Boxplot of hypervolume results for all runs and optimizers. Note that optimizers are sorted by year of publication - from oldest (2001, left) to newest (2016, right).

Our findings show that there is a substantial variation in both the mean HV values of different optimizers as well as variations over different runs of the same optimizer. The MOPSO method obviously struggles substantially with a lowest HV of 0.06. This might be due to sub-optimal hyperparameter settings, an implementation problem or some feature of our application problem. The remaining optimizers all stay in a range of HV values of [0.20, 0.29]. Still this means that some runs produced 50% better hypervolume values than other runs. Another interesting observation is that we can't see a direct trend for increasing performance over time of publication. The oldest (SPEA2) and the newest (RVEA) algorithm are very similar in their HV values.

Results from the optimization runs could be used in a number of ways, some of which will be shown exemplary below. First we plot desirabilities for all final solutions from all runs in a parallel coordinate plot (Fig. 5). This type of representation allows us to understand the relations between the objectives for all populations. We see a large spread of solutions in desirability space. A clear correlation is that investment cost is inversely proportional to annual cost and emissions, while the latter two are strongly correlated. But for example for resilience and battery lifetime the relations are more diverse. A very interesting finding is shown in Fig. 6, where we plot desirability values for objective 2 (annual cost) vs objective 1 (investment cost) for all runs of 5 optimizers. Each symbol represents one final solution. Individual runs show patterns different in detail but overall rather similar. We can see that solutions from different optimizers can cover different parts of objective space. In our example solutions from IBEA and RVEA strongly overlap, while MOPSO, SPEA, and especially PICEAg occupy different areas (note that some of the shown solutions might be dominated).

To better understand the solutions we also studied the ranges of found parameters. Due to space limitations we can only show three exemplary parameter

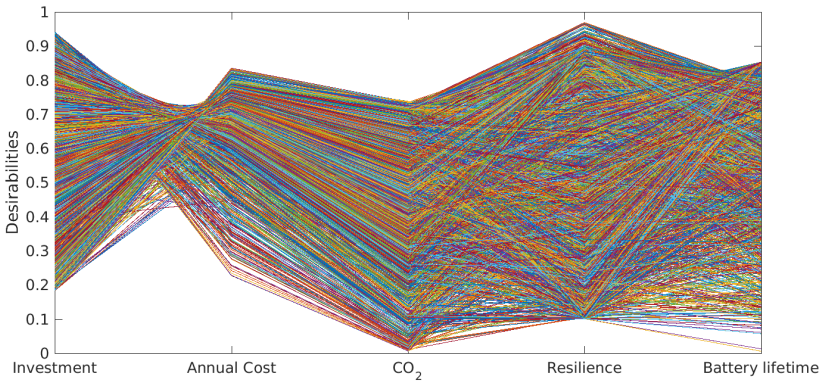

Fig. 5. Parallel coordinate plot of desirabilities

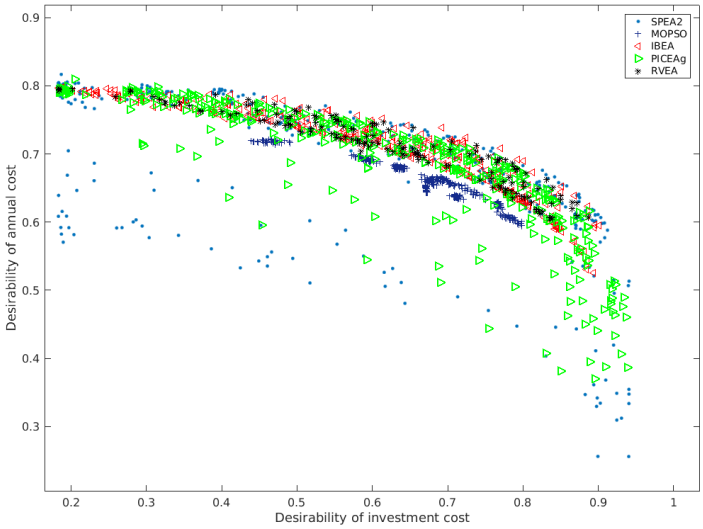

Fig. 6. Desirabilities for emissions vs. initial investment for different optimizers

histograms (Fig. 7). We see that almost all solutions select a charging threshold of 0, which means that the battery is charged as soon as surplus power is available (*top*). This makes sense as feeding energy back to the grid is less desirable than self-consumption. In the middle panel we see a large variety in discharging thresholds which is probably due to a low impact of the battery for the overall energy flows (the battery is mainly used for resilience and peak shaving, but the latter is a very tricky business). The bottom row finally demonstrates that in all cases a larger heat storage is beneficial.

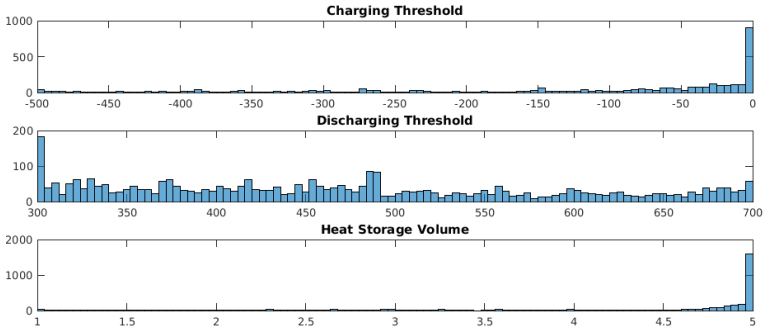

Fig. 7. Histograms of parameter values of all final solutions from all runs. *Top*: battery charging threshold u_{Batt}^{Charge} , *middle* battery discharging threshold $U_{Batt}^{discharge}$, *bottom* heat storage volume V_{stor}

A look at the minimum and maximum desirability values for all objectives over all final solutions from all runs (Tab. 4) shows that we can find a large variety of solutions with higher peak desirabilities (except for investment cost) compared to the three baseline configurations (see Tab. 3).

Objective	Max. desirability	Min. desirability
C_{Invest}	11,600 Euros (0.94)	550,000 Euros (0.18)
C_{Annual}	359,000 Euros (0.84)	393,000 Euros (0.22)
$emis_{total}$	2134 t (0.74)	2225 t (0.01)
$resi$	1252 s (0.97)	1.7 s (0.10)
$lifetime$	28.7 a (0.85)	17.4 a (0.01)

Table 4. Best and worst objective values (desirabilities in brackets) for all found solutions, please compare these values to results from Tab. 3.

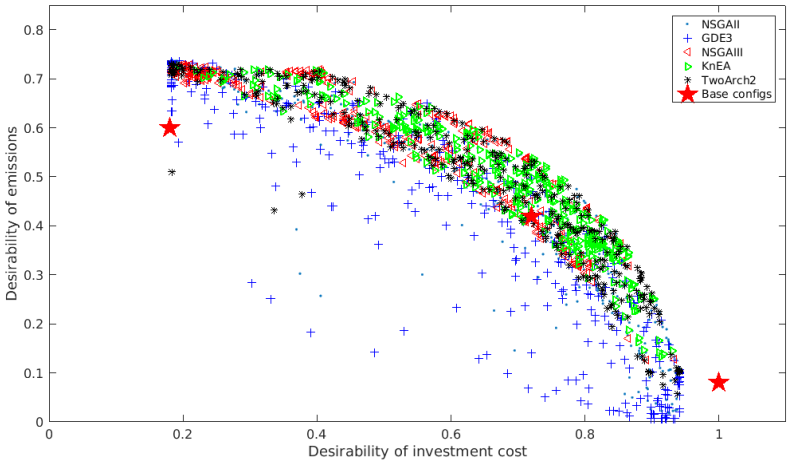

Fig. 8. Results (yearly emissions vs. investment cost) for 5 selected optimizer (all runs) plus initial configurations.

3.3 Comparison to baseline solutions

In Fig. 8 we show for 5 selected optimizers and all runs the desirabilities for objective 3 vs objective 1. We also added the three reference solutions. It is obvious that coverage in desirability space is not the same for all optimizers. In relation to the baseline solutions we see that most found solutions are clear improvements and both middle and full extension baseline configuration are clearly dominated by the majority of solutions (this is also true for other objectives).

4 Summary and Outlook

For the application of building energy system optimization we compared 10 different many-objective optimization algorithms using the hypervolume indicator. We found that different optimizers exhibit substantial differences in their performance as indicated by the hypervolume of the final population. Interestingly, we don't see a clear trend for increasing performance with more recent algorithms compared to older work. This means that at least when using the default hyperparameters we don't see any clear benefits from using more recently published algorithms compared to older ones. Among the 10 test optimizers, and without any tuning of hyperparameters, IBEA was the best performing algorithm in terms of hypervolume. It also exhibited, along with TwoArch2, a very stable performance (low performance variations between runs).

When looking at solutions from individual runs the different optimizers show very different patterns of solutions. We conclude that the choice of the optimizer is not arbitrary, and other optimizers might further improve the results.

We used desirabilities to confine all objectives to the range of [0,1]. We found desirabilities to ease the handling of results for visualization and analysis of solutions, but in several cases (test) decision makers requested that the real objective values are shown.

As future work we plan to extend the system under study by e-mobility modules (electric vehicles and charging stations). As optimization runs still require too much time, we plan to investigate different options for speeding the process up like meta-modeling [26]. Finally, we need to investigate preference-based methods [10] and Multi-Criteria Decision Making (MCDM) on this application to actually identify a promising upgrade option in cooperation with a human decision maker.

References

1. Wagner, T., Trautmann, H.: Integration of preferences in hypervolume-based multiobjective evolutionary algorithms by means of desirability functions. Evolutionary Computation, IEEE Transactions on 14(5), 688–701 (2010)
2. Yang, R., Wang, L.: Multi-objective optimization for decision-making of energy and comfort management in building automation and control. Sustainable Cities and Society 2(1), 1 – 7 (2012), http://www.sciencedirect.com/science/article/pii/S221067071100059X
3. Fadaee, M., Radzi, M.: Multi-objective optimization of a stand-alone hybrid renewable energy system by using evolutionary algorithms: A review. Renewable and Sustainable Energy Reviews 16(5), 3364 – 3369 (2012), http://www.sciencedirect.com/science/article/pii/S1364032112001669
4. Khodr, H.M., Vale, Z.A., Ramos, C., Soares, J.P., Morais, H., Kdr, P.: Optimal methodology for renewable energy dispatching in islanded operation. In: IEEE PES T D 2010. pp. 1–7 (April 2010)
5. Naharudinsyah, I., Limmer, S.: Optimal charging of electric vehicles with trading on the intraday electricity market. Energies 11(6) (2018)
6. Fritzson, P., Bunus, P.: Modelica – a general object-oriented language for continuous and discrete-event system modeling. In: Proceedings of the 35th annual simulation symposium. pp. 14–18 (2002)
7. Unger, R., Mikoleit, B., Schwan, T., Bäker, B., Kehrer, C., Rodemann, T.: Green building - modeling renewable building energy systems with emobility using Modelica. In: Proceedings of Modelica 2012 conference. Modelica Association, Munich, Germany (2012)
8. Rodemann, T.: A many-objective configuration optimization for building energy management. In: Proceedings of IEEE WCCI (CEC) (2018)
9. Ogino, Y., Iida, R., Rodemann, T.: Using desirability functions for many-objective optimization of a hybrid car controller. In: GECCO 2017 conference companion (2017)
10. Cheng, R., Rodemann, T., Fischer, M., Olhofer, M., Jin, Y.: Evolutionary many-objective optimization of hybrid electric vehicle control: from general optimization to preference articulation. IEEE Transactions on Emerging Topics in Computational Intelligence 1(2), 97–111 (February 2017)
11. Tian, Y., Cheng, R., Zhang, X., Jin, Y.: PlatEMO: A MATLAB platform for evolutionary multi-objective optimization. CoRR abs/1701.00879 (2017), http://arxiv.org/abs/1701.00879

12. López-Ibáñez, M., Dubois-Lacoste, J., Cáceres, L.P., Birattari, M., Stützle, T.: The irace package: Iterated racing for automatic algorithm configuration. Operations Research Perspectives 3, 43–58 (2016)
13. Rodemann, T.: Industrial portfolio management for many-objective optimization algorithms. In: Proceedings of IEEE WCCI 2018 (CEC) (2018)
14. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6, 182 – 197 (2002)
15. Deb, K., Jain, H.: An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. IEEE Transactions on Evolutionary Computation 18(4), 577–601 (2014)
16. Cheng, R., Jin, Y., Olhofer, M., Sendhoff, B.: A reference vector guided evolutionary algorithm for many-objective optimization. IEEE Transactions on Evolutionary Computation 20(5), 773 – 791 (2016)
17. Zitzler, E., Künzli, S.: Indicator-based selection in multiobjective search. In: Yao, X., Burke, E., Lozano, J., Smith, J., Guervós, J., Bullinaria, J., Rowe, J., Tiño, P., Kabán, A., Schwefel, H. (eds.) Parallel Problem Solving from Nature: PPSN VIII. Lecture Notes in Computer Science, vol. 3242, pp. 832–842. Springer (2004)
18. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the strength pareto evolutionary algorithm for multiobjective optimization. In: Proc. of Evolutionary Methods for Design, Optimisation and Control with Application to Industrial Problems. pp. 95–100 (2001)
19. Zhang, X., Tian, Y., Jin, Y.: A knee point driven evolutionary algorithm for many-objective optimization. IEEE Transactions on Evolutionary Computation 19(6), 761–776 (2015)
20. Kukkonen, S., Lampine, J.: GDE3: The third evolution step of generalized differential evolution. In: Proceedings of the 2005 IEEE Congress on Evolutionary Computation. pp. 443–450 (2005)
21. Coello, C.C., Lechuga, M.S.: MOPSO: A proposal for multiple objective particle swarm optimization. In: Proceedings of the 2002 IEEE Congress on Evolutionary Computation. pp. 1051–1056 (2002)
22. R. Wang, R.C.P., Fleming, P.J.: Preference-inspired coevolutionary algorithms for many-objective optimization. IEEE Transactions on Evolutionary Computation 17(4), 474–494 (2013)
23. H. Wang, L.J., Yao, X.: Two Arch2: An improved two-archive algorithm for many-objective optimization. IEEE Transactions on Evolutionary Computation 19, 524–541 (2015)
24. Zhang, Q., Li, H.: MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Trans. on Evolutionary Computation 11(6), 712–731 (2007)
25. Bader, J., Zitzler, E.: HypE: An algorithm for fast hypervolume-based many-objective optimization. Evolutionary Computation 19(1), 45–76 (2011)
26. Chugh, T., Jin, Y., Miettinen, K., Hakanen, J., Sindhya, K.: A surrogate-assisted reference vector guided evolutionary algorithm for computationally expensive many-objective optimization. IEEE Transactions on Evolutionary Computation 22(1), 129–142 (2018)