AUTOMATED TIME SERIES BASED GRID EXTENSIONS PLANNING USING A COUPLED AGENT BASED SIMULATION AND GENETIC ALGORITHM APPROACH

Johannes HIRY, Chris KITTL,
Christian RÖMER, Christian REHTANZ
TU Dortmund University – Germany
johannes.hiry@tu-dortmund.de

Lars WILLMES,
Sebastian SCHIMMEYER
intulion solutions GmbH – Germany
lars.willmes@intulion.de

ABSTRACT

In recent years, the distribution grid planning process has faced the big challenge to integrate renewable energy sources in its planning methodology while preserving a secure and stable provision of electricity. With the currently observable efforts to electrify human mobility all around the world, another new challenge arises for the planning and operation of distribution grids. To address these challenges and to leverage the opportunities that are accompanied by them, new methods for the planning of distribution grids as well as planning decision-supportive approaches and algorithms are needed. The presented approach contributes to the described demands by means of a coupled approach, using both distribution grid time series as well as a genetic algorithm to support decision making in the planning process considering not only new assets for grid reinforcements and extensions but also smart-grid and operational opportunities.

INTRODUCTION

Besides the operation of their grid, one of the main tasks of distribution grid operators (DSOs) is the suitable and cost-efficient planning of the distribution grid under their control. The transformation of the existing energy system from a conventional power plant based into a renewable energy sources based system has challenged the DSOs planning process in recent years. [1] With the currently globally observable efforts to electrify human mobility, the next dare for planning and operating distribution grids is already conceivable. Besides the conventional approach to install new grid assets, operational flexibility provided through battery storages, on load tap changing (OLTC) transformers or smart market mechanisms like "traffic light concepts" may be additionally considered in the planning process as less cost intensive options for grid extensions and maintenance. [2] The consideration of all possible alternatives, including operational flexibility and external factors, highly increases the planning effort. Therefore, new planning methods as well as decision-supportive approaches, considering multidimensional dependencies as well as smart-grid mechanisms, need to be developed. [3]
This contribution gives first insights on how an automated decision-supportive tool for a future proof distribution grid planning may look like. As part of the research project "Agent.GridPlan" the functionality of the agent-based simulation *SIMONA* has been extended to interact with an existing automated genetic grid extension planning algorithm, that uses the results from *SIMONA* to propose grid extensions in case of a congestion. [4],[5]
In the first part of the paper the overall concept of the developed coupled simulation framework and its functionality are outlined. Afterwards, the extensions and adjustments of the used simulation approaches are described in detail, subsequently followed by a small application example. In the last part, a conclusion of the developed methodology is drawn and an outlook to further work is given.

DEVELOPMENT OF A COUPLED SIMULATION FRAMEWORK

To allow for the assessment of the current distribution grid state, *SIMONA* has to provide specific data to a genetic algorithm (GA). Furthermore, besides providing the data through a specific interface and format that is understandable by the GA, its results – proposed measures to overcome the congestions – have to be processed and considered for the subsequent simulation run. On the GA part a suitable representation of the optimisation problem needs to be found to encode the possible changes of the power grid on one side and to allow for simple application of genetic operators on the other side. In particular, this involves applying said changes to the grid at an optimised point of simulation time at which the change is actually applied. Strictly spoken, this means that variables of the optimisation problem consist of two components: the actual change to the grid ("what") and the time of application ("when").
Currently, the actual timing of change application is not used since simulation time is restricted to a single year such that timing of measures is of rather low significance. Further development will extend simulations to multi-year time frames where the actual placement of measures in time will be constrained by capacity limits on financial volume and available man power in a given time interval.

Overall coupling concept

On a high abstractive level, the overall concept of the integrated grid evaluation and extension approach can be structured in four main steps, depicted in **Figure 1.**

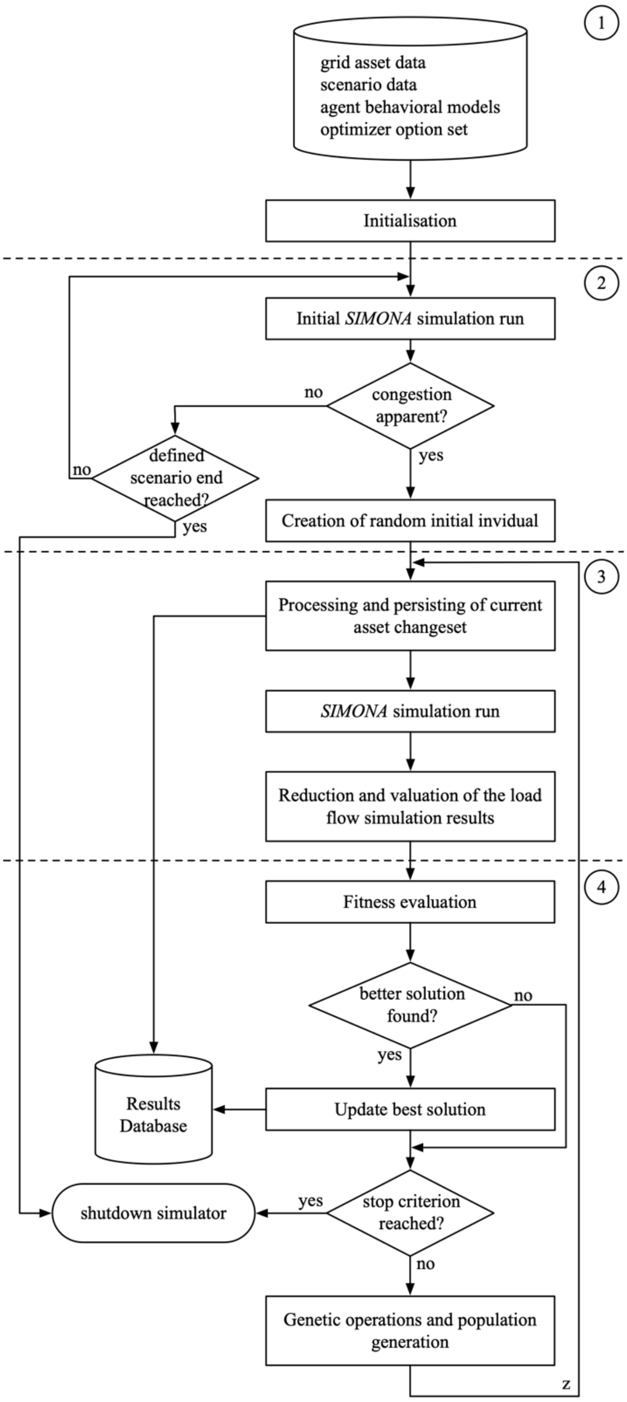

Figure 1: Conceptual flow chart overview of the coupled simulation framework

1. **Simulation Initialisation process**
During the initialisation, both, *SIMONA* as well as the GA, are provided with their input data that is held in an object relational database. It consists of the grid asset data, scenario data, the specific agent behavioural models as well as a set of available options for genetic operations. Furthermore, a stop criterion which can either be a specific amount of simulation runs $z \in \mathcal{Z}$, a convergence criterion ε or another, population specific measurement, is set. *SIMONA*'s bottom-up structure allows for the consideration of the whole grid in detail for all power flow simulations without any aggregation or simplifications.

2. **Initial simulation**
After the initialisation has been completed, a first initial power flow simulation including all agent's behaviour is executed to check if there are any violations in the initial grid configuration. As several scenarios might be investigated e. g. different penetration levels of electric vehicles (EV) or distributed energy resources (DER) over several years, even the initial grid configuration might not be sufficient anymore. After the initial power flow simulation, the calculation results are evaluated to identify possible thermal overloads of assets or violations of the allowed voltage range. If none exist, the simulation will head over to the next time frame or terminate. Otherwise, the GA grid extension process is triggered and creates a random initial individual to solve the grid congestion.

3. **Population processing and simulation run**
The third step starts with processing the optimiser adjustments and persisting its changeset into the results' database. This process includes financial valuation of the proposed individual by calculating and persisting capital expenditures (CAPEX) and operational expenditures (OPEX) as part of the change set. Furthermore, resulting cost data is hand over to the GA for being considered in the fitness function. For performance reasons, *SIMONA* result data as well as overload information are discarded here and only the optimiser change set is persisted. Following the changeset persisting, a steady state power flow analysis for the predefined simulation time window (currently one year) is performed and corresponding time series are generated. Based on them, the current state of the grid is valuated for each time step by calculating potentially occurring violations of grid constraints. The results of this step are handed over to the GA for further processing and application of genetic operations.

4. **Fitness evaluation, genetic operations and best solution update**
The GA converts the results supplied by the simulation to fitness values that allow for numerical comparison of alternative solutions. Using the fitness values tournament selection is applied to update populations and for mating selection. New individuals are generated by uniform crossover and a simple mutation operator based on uniform sampling for grid assets and on Gaussian Mutation for timing information. The GA status is updated accordingly and new data are exported to the

runtime environment for evaluation by the simulation.

While the steps 1 and 2 are executed only once, step 3 and 4 are repeated until the stop criterion is reached.

Upgrades and extensions of the existing time series generation simulation

While the initial concept and design of SIMONA was to provide the distribution grid operator with abilities to simulate their grid, generate time series and store them into a database for further, manual investigation, the immediate processing of results, grid modifications on runtime and skipping of invalid time steps hasn't been part of the concept yet. To allow for the aforementioned coupled simulation several new features have to be developed. Furthermore, as simulation runtime and performance play a major role when simulations are carried out multiple times, several improvements to increase the overall performance have been addressed. The following section provides insights into recent development and improvements that have been made to set up the coupled simulation. For a more detailed description of *SIMONA* and its functionalities, please refer to [6].

Initialisation of the grid data, on-the-fly grid adaptions during runtime and persisting proposed extensions

One requirement is to modify the grid model during runtime or at least after each iteration of the genetic algorithm. As reloading of the grid model from database turned out to be costly in terms of simulation time, a deep copy of the grid data is made before the initial simulation run. This copy is hold in memory and reloaded when adaptions proposed by the optimiser in each subsequent simulation have to be processed.
Considering the number of iterations, the GA is executed until its solution converges, storing and evaluating all result data would be non-feasible. To address this challenge, the authors reduced the amount of data to be stored in each iteration to only the provided extension proposals. This allows for a fast persistence, while preserving the ability to investigate specific iterations in detail after the coupled simulation terminates without producing a unnecessarily big amount of unused data.

Data reduction, constraints calculation and valuation

Thanks to its modular bottom-up architecture, *SIMONA* is able to provide detailed time series for each grid asset as well as some other grid operation information. While in the past, the overall investigation of grid asset loading data has been the primary goal, the provided raw data is not usable by the genetic algorithm without pre-processing, data reduction and results valuation.
By design, the grid model in *SIMONA* is divided into multiple subnets representing galvanically disjoint areas of the grid. These subnets are represented by unique *NetAgents* which carry out continuous power flow calculations for their underlying grid. Beside this, they are responsible for processing and persisting the corresponding generated time series. [6] Based on the concept of a finite state machine (FSM) the *NetAgent*'s current state cannot only be observed at any time in the simulation but also be extended by new states. To pre-process the calculation results of each subnet, the existing *NetAgentBehaviour* FSM has been extended by an additional state that carries out the following tasks after each time step:

1. Configuration dependent selection of relevant calculation results
2. Computation of assets' loading and assigning an overload dependent penalty cost to each asset
3. Storing results in memory for further processing
4. Hand over a result vector to the genetic algorithm

In a more formal way, this new FSM state can be described as follows.

$$\begin{aligned} k_j &\in \mathcal{K} = \{k_1, k_2, \ldots, k_j\} \\ t &\in \mathcal{T} = \{1,2, \ldots, T\} \\ a_i &\in \mathcal{A} = \{a_1, a_2, \ldots, a_i\} \\ z &\in \mathcal{Z} = \{1,2, \ldots, Z\} \end{aligned} \tag{1}$$

Let $\mathcal{K}$ denote the set of available *NetAgents*, $\mathcal{T}$ the set of pre-defined simulation timesteps, $\mathcal{A}$ the set of grid assets under investigation and $\mathcal{Z}$ the predefined number of simulations. The calculation of the penalty costs (step 1. and 2.) can then be written as

$$c_{pen,k,t}(x_a) = \begin{cases} (x_a - 100\,\%) \cdot c, & \forall x_a > 100\,\% \\ 0, & \forall x_a \leq 100\,\% \end{cases} \tag{2}$$

where c represents relative overload costs in €/% and x_a the loading of the valuated asset i. These values are aggregated to the $(n+1) \times 1$ result vector $\mathbf{R}_n$ which is handed over to the genetic algorithm after each run. It can be described a

$$\mathbf{R}_n = \begin{pmatrix} \sum_{a=1}^{A} c_{inv,a,z} \\ \sum_{k=1}^{K} c_{pen,k,1} \\ \vdots \\ \sum_{k=1}^{K} c_{pen,k,T} \end{pmatrix}. \tag{3}$$

The simulation result vector $\mathbf{R}$ that is handed over to the genetic algorithm consists of the aggregated value of the investment costs of the optimiser changeset, as well as the aggregated penalty costs values for each simulated time step.

Extension of the NetAgents behaviour to allow invalid time step valuation

In *SIMONA*, the implemented power flow calculations are based on a numeric Newton Raphson algorithm. While executing this algorithm over multiple voltage levels in a large electrical grid, non-convergent behaviour might be observable in one or multiple time steps. In the initial design of the simulation, the case of non-convergent power flow calculations has been considered as an invalid state. The simulation status changed to failure instead of just marking the corresponding time step as failed and proceeding with the simulation. The amount of infeasible time steps in a given period allows for a qualitative comparison of two infeasible grid asset bases – one is "more feasible" over another. To not only allow for a valuation of a grid extension proposal, even with invalid time steps, but also to send a corresponding feedback to the genetic algorithm the existing way of executing power flow calculations had to be extended and a second new FSM state was introduced.

Assuming the case that a time step does not converge, the corresponding *NetAgent* now informs its superior subnet about the failure including a request to skip this time step. As the subnet is a child of the superior net, this leads consequently to a "fail and skip" state in the superior net and is propagated until the highest *NetAgent* is informed about the skip. At this point, the corresponding time step is marked as failed and the corresponding cost values in equation (*2*) are set to infinity. Depending on the configuration of the GA, a predefined number of time steps marked with infinity costs can be allowed to relax the optimisation constraints.

Automated distribution grid extension using a genetic algorithm

The choice to use a GA is motivated by the nonlinear and combinatorial nature of the problem at hand. The variables to be optimised, i.e. basically grid assets, are categorial values drawn from a fixed set of possible values. The showcase example is the type of an overhead transmission line that can be used to connect two substations of the power grid. This transmission line type defines the electrical properties of the transmission line and its actual cost. For the optimisation a fixed set of predefined line types is assumed, reflecting current grid planning practice. In GA terms the line type is used directly as a phenotypic representation, i.e. variables are operated on directly without further encoding. In general, standard operators and population structures are applied to a genome built from all line type variables.

Since the power grid contains many more potential variables than the example line types, further assets are added in a similar fashion. The conceptual design of the method allows the extension to further asset classes as long as they are supported by the grid model. This allows optimisation of arbitrary asset classes, especially rather exotic or special purpose ones expected to be useful in future grid designs.

The time series nature of the simulation approach adds an additional nonstandard feature to the optimisation problem: any action taken on the simulated power grid, like the exchange of a line type, is supposed to happen at a given time within the simulation time frame. Therefore, all optimisation variables need to have an additional time stamp that indicates, at which simulation time the change is applied. This time stamp is implemented as a real valued parameter that is associated to any optimisation variable and adapted by the optimiser. By that, the optimisation adapts both the optimal asset type and the optimal simulation time to apply the change. The Genetic Algorithm uses a phenotypic representation of the time parameter with uniform crossover and Gaussian Mutation with fixed step size. The time stamp information is rounded to the closest feasible value in simulation time to match simulation requirements.

More formally, given the set of grid asset classes $\mathcal{A}$ (see equation (1)), where each asset class has an associated set of feasible type values $T(a_i) = \{T(a_i)_1, \ldots, T(a_i)_m\}$ for a power grid with n components the genome of an individual of the Genetic Optimiser is a vector of value pairs $x = (T_1, t_1, \ldots, T_n, t_n)$, where $T_i \in \{T(a_1), \ldots, T(a_k)\}$ denotes the type to be set for the i-th grid asset and $t_i \in \{1, \ldots, S\}$ denotes the point of time in the simulation, where the measure is taken.

The fitness function is composed of the cost associated with the actions suggested to the simulation on the one hand and a number of constraints that need to be met on the other hand. Both cost and constraint values are computed by the simulation and suitably aggregated in the fitness function. The optimiser is capable of handling constraints and objectives separately and does not require constraint violations to be encoded in penalty terms added to the objective function value. This opens the way to optimize for strict feasibility without compromising penalty term definitions if required.

Additional data generated by the simulation is used to heuristically steer the search towards good solutions. Due to rather long computation times the optimiser implements parallel evaluation of the fitness function to speed up total optimisation progress.

APPLICATION EXAMPLE

To investigate to correct functionality of the developed approach a small application example has been simulated as a first proof of concept. The grid for this case has been selected from a large real world distribution grid [7] with focus on traceability to ensure an easy understanding of the GA optimisation process.

Simulation setup

The investigated grid is a real world low voltage grid consisting of approx. 900 nodes and lines. Loads are modelled as standard load profiles [8], DER are modelled based on the descriptions in [6]. Simulations are carried out for one day in January with an hourly resolution. Six different options for line extensions and eight different options to extend the transformer capacity were available to the GA. The chosen scenario leads to a high share of additional PV penetration in the grid.

Simulation Results

As a first result, **Figure 2** demonstrates the fundamental operating principle of the developed approach. The abscissa shows the wall clock time from the computation on a single processor machine, the right ordinate and red graph denotes aggregated congestion and asset investments costs in Euro, while the left ordinate and blue graph denotes the valuated grid congestion. The congestion being a constraint is required to be less than or equal to 0 for feasible solutions, and costs are intended to be minimised.

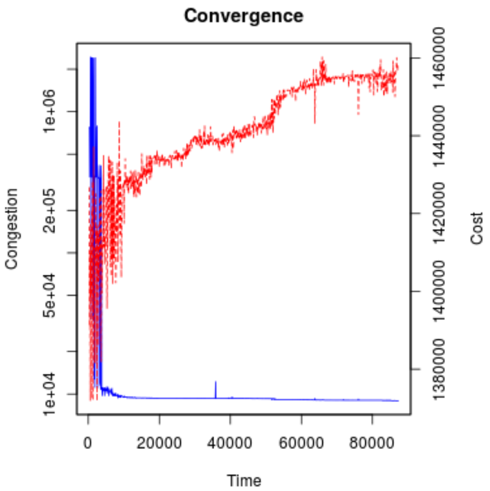

Figure 2: Optimiser convergence in the application example

As apparent in **Figure 2**, a convergence of grid congestion and grid extension costs in the course of an optimisation run takes place. On a first glimpse the convergence implies a failure of the optimisation method, because it neither achieves feasible nor optimal values, since the constraint value is strictly greater than 0 and the cost is actually increasing where it should be minimised. On closer inspection it turns out though, that after an initial drop of the congestion value there remains one transformer in the simulated grid that is consistently overloaded by a small amount. A further review of the available asset options and the simulation setup reviled that, with the provided asset changeset, the overloading on one transformer can't be fully removed, but is in an acceptable range, because a slight overloading only occurs in a few hours of the simulated time series. Still the optimiser is capable of reducing the actual overload during optimisation, although it applies actions that, from an engineering standpoint, would be a bad idea. In fact, the optimiser replaces a couple of other transformers and lines by alternatives with different impedances. Consequentially, the congestion indicator value is decreased slightly at the expense of rather big increases in total cost. Although the technical result achieved by the optimiser might be of non-optimal engineering value, it still shows that both the method and its prototypical implementation work as supposed.

CONCLUSION AND OUTLOOK

In this contribution, a methodology to couple an agent-based grid simulation with a genetic algorithm for grid extension planning has been presented. Its fundamental functionality for has been demonstrated in a very small application example. As a next step, the developed methodology will be refined and extended in a way to supply additional measures for the optimiser to improve the results from an engineering point of view. Furthermore, as the demonstrated application example represents a comparably small application example from a large real world distribution grid, the developed approach is going to be applied to the whole grid over multiple voltage levels.

Acknowledgments

The authors gratefully thank Westnetz GmbH for supporting the presented research by granting access to the mentioned real-life grid model during their participation in the research project Agent.GridPlan. It has been funded by the European Fund for Regional Development under grant agreement number EU-1-1-006.

REFERENCES

[1] Deutsche Energie-Agentur GmbH (editor), 2012, "*Ausbau- und Innovationsbedarf der Stromverteilnetze in Deutschland bis 2030.*", Berlin, available: https://www.dena.de/fileadmin/dena/Dokumente/Pdf/9100_dena-Verteilnetzstudie_Abschlussbericht.pdf

[2] J. Kays, A. Seack, U. Häger, 2016, "Consideration of innovative distribution grid operation concepts in the planning process", *Proceedings of 2016 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, Ljubljana, DOI: 10.1109/ISGTEurope.2016.7856331.

[3] L. Jendernalik, D. Giavarra, C. Engels, A. Maier, 2016, „Resilient and comprehensive networks planning is key to a successful integration of renewable energy resources", *CIRED Workshop 2016*, Helsinki

[4] Thomas Bäck, D.B Fogel, Z Michalewicz, 1997 *"Handbook of Evolutionary Computation"*, CRC Press

[5] L. Jendernalik, C. Engels, J. Hiry, C. Kittl and C. Rehtanz, 2017, A Holistic Planning Approach: Enhacement of the grid expansion using the flexibility of network participants, *CIRED - Open Access Proceedings Journal, vol. 2017, no. 1, pp. 2312-2315.*

[6] J. Kays, C. Rehtanz, 2016, "Planning process for distribution grids based on flexibly generated time series considering RES, DSM and storages", *IET Generation, Transmission & Distribution (IET GTD)*, vol. 10, 3405-3412. DOI: 10.1049/iet-gtd.2015.0825.

[7] C. Kittl, J. Hiry, C. Wagner, C. Rehtanz, C. Engels, 2019, "Large scale agent based simulation of distribution grid loading and its practical application", *Proceedings of the 25th International Conference on Electricity Distribution (CIRED),* Madrid

[8] H. Meier, C. Fünfgeld, T. Adam, B. Schlieferdecker, 1999, "Repräsentative VDEW-Lastprofile", available: https://www.bdew.de/media/documents/1999_Repraesentative-VDEW-Lastprofile.pdf