Received: 30 August 2025 | Revised: 24 March 2026 | Accepted: 19 May 2026 | Published online: 3 June 2026

RESEARCH ARTICLE

Artificial Intelligence and Applications
2026, Vol. 00(00) 1–14
DOI: 10.47852/bonviewAIA62027499

BON VIEW PUBLISHING

Hybrid Method for Optimizing the Structure of Power Generation Capacities Using An Evolutionary Algorithm and An MILP Solver

Sergii Saukh[1,*] **and Taras Puchko**[1]

[1] *Department of Mathematical and Econometric Modelling, G.E. Pukhov Institute for Modelling in Energy Engineering of the National Academy of Sciences of Ukraine, Ukraine*

Abstract: A hybrid method is proposed for solving large-scale mixed-integer linear programming (MILP) problems that arise in the optimization of the generation capacity structure of electric power systems combining conventional and renewable energy sources. The novelty of the proposed approach lies in combining the decomposition of the original generation expansion planning problem into investment-level search and operational-level evaluation with evolutionary exploration of the discrete space of generation capacity decisions and exact parallel solution of operational MILP subproblems on high-performance computing (HPC) resources. This combination enables detailed operational evaluation without replacing the operational model with a surrogate approximation. Numerical experiments using the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) and the Solving Constraint Integer Programs (SCIP) solver showed that the global optimum was achieved in 98% of runs for a population size of 80 and that, compared with the single-threaded SCIP baseline, the 128-thread configuration reduced the average runtime for successful runs by 3.77×. At the same time, scaling with respect to computation time and memory usage exhibited nearly linear behavior. A comparison with the alternative Evolutionary Centers Algorithm demonstrated the superiority of CMA-ES in convergence reliability. The obtained results indicate that the proposed method is well-suited for deployment in HPC environments and can serve as an effective tool for strategic planning of electric power system development. Future research should focus on extending the applicability of the approach to multi-period planning problems and analyzing the impact of algorithm parameters on convergence and solution accuracy.

Keywords: generation expansion planning, evolutionary algorithm, parallel optimization, mixed-integer linear programming

1. Introduction

Global efforts to mitigate climate change require an accelerated transition from fossil fuel-based electric power systems to renewable and low-carbon sources. In China, the world's largest energy consumer, the rapid growth of wind, solar, and hydropower illustrates both the scale of the challenge and the potential for emission reduction [1]. At the same time, Scandinavian countries demonstrate progress in carbon management technologies, which are considered an additional tool for achieving climate goals [2]. This highlights the necessity of simultaneously advancing renewable energy development and critically reassessing the role of transitional technologies in the future energy landscape.

Solar and wind power plants (SPP and WPP) are characterized by high variability of electricity generation and limited regulation capabilities, since their output depends on weather conditions [3]. With the growing share of renewable energy sources (RES) in the generation structure, balancing demand and supply becomes more complex, necessitating the integration of energy storage systems (ESS) and dispatchable capacities, in particular thermal power plants (TPPs). This underscores the need for advanced modeling tools for development planning and optimization of electric power system operation.

The rapid increase in the share of solar and wind generation changes the computational requirements of generation expansion planning (GEP). In systems with a large contribution from variable renewable sources, planning decisions must account not only for capital costs but also for chronological operating constraints, balancing needs, storage behavior, reserve capabilities, and the interaction between demand variability and renewable production patterns. As a result, GEP evolves from a relatively aggregated investment problem into a computationally intensive integrated planning and operation problem that often requires high-resolution mixed-integer linear programming (MILP) formulations [4, 5].

***Corresponding author:** Sergii Saukh, Department of Mathematical and Econometric Modelling, G.E. Pukhov Institute for Modelling in Energy Engineering of the National Academy of Sciences of Ukraine, Ukraine. Email: sergii.saukh@pimee.ua

Global energy models such as MARKAL, TIMES, and LEAP are widely applied for long-term planning of electric power system development. However, their limitation lies in low operational granularity: they do not capture the operation of individual generating units and often neglect technical constraints [6]. This may lead to overestimation of the potential for RES integration and underestimation of the need for balancing capacities. In contrast, unit commitment (UC) models account for the technical characteristics of power plants, ensuring higher forecast accuracy [7].

For modeling wind and solar power plant (WPP and SPP) generation, averaged graphs or probability distributions are often used [8], which only partially capture the stochastic nature of renewable resources. A suitable approach is the use of representative models that reproduce realistic generation graphs based on data clustering, thereby improving the accuracy of assessing the need for balancing capacities and ESS [9].

As a result of combining representative models of RES and ESS with UC models, complex optimization problems arise, usually in the form of large-scale MILP problems [9]. These involve a significant number of discrete variables and provide high temporal resolution. Even with the application of representative models, their solution often requires modern optimization methods that include decomposition and parallel computing.

One of the common approaches is Benders decomposition, which divides the problem into a master problem and subproblems and iteratively refines the solution [10]. The method has demonstrated effectiveness in generation capacity planning, particularly in mixed electric power systems [11]. At the same time, the classical algorithm often requires many iterations and leads to increased computational costs [4]. Therefore, its modifications and acceleration strategies remain relevant.

Classical Benders decomposition is applicable when investment decisions, including integer variables, are assigned to the master problem and fixing them yields a continuous linear operational subproblem [4, 12]. In realistic GEP models, however, the operational level often includes integer variables associated with UC and other detailed technical constraints [4, 7, 13]. In such cases, the standard form of classical Benders decomposition is not directly applicable because the derivation of classical feasibility and optimality cuts relies on linear programming duality for the subproblem [4]. This motivates the search for alternative decomposition strategies and hybrid methods that preserve detailed operational modeling while remaining computationally tractable [4, 12, 14].

An alternative is metaheuristic algorithms, which are increasingly applied for optimization in the energy sector. For instance, the Honey Badger Algorithm has been employed for generation capacity expansion planning in systems with a high share of RES [15], while binary genetic algorithms (GAs) have demonstrated effectiveness in planning the structure of electric power systems combining hydropower with solar generation [16]. Metaheuristics do not require information about gradients or problem structure, which makes them suitable for "black-box" problems [17]. Recent developments, such as LB+IC-CMA-ES, demonstrate the capability of metaheuristic algorithms to solve problems with discrete variables [18].

The combination of metaheuristics with exact methods within a decomposition framework enhances efficiency: discrete variables are handled by evolutionary algorithms, whereas continuous ones are solved by exact solvers [19]. Hybrid approaches ensure a balance between solution quality and speed, while parallel processing of subproblems contributes to scalability [20]. This establishes the foundation for high-performance hybrid methods of energy planning optimization.

GEP has long been recognized as a computationally difficult optimization problem, especially when investment decisions must be coordinated with detailed operational constraints [4]. Early studies demonstrated the applicability of metaheuristic methods such as GAs, simulated annealing, and particle swarm optimization to GEP, while also showing that the relative performance of these methods depends strongly on the structure of a particular problem instance [21]. As power systems incorporate larger shares of solar and wind generation, the planning problem becomes more difficult because variability, uncertainty, balancing requirements, storage operation, and chronological operating constraints must be represented more accurately [4, 14].

One important research direction is the use of decomposition methods. Benders decomposition and its modern extensions have been widely studied for capacity expansion and energy system planning [4]. Recent works have proposed regularized Benders methods, logic-based variants, and parallel decomposition strategies in order to improve convergence and computational scalability in large-scale models with detailed operations and time coupling constraints [4, 14]. These studies show that decomposition remains one of the key methodological directions for high-resolution planning models.

Another important direction is the use of metaheuristic optimization. Evolutionary and other population-based methods are attractive because they do not require derivative information and can naturally explore discrete and nonconvex investment spaces [21]. They have been applied to GEP, microgrid optimization, and related energy system design problems [21]. At the same time, their practical effectiveness depends heavily on how candidate solutions are evaluated, since realistic operational assessment is computationally expensive [20, 21].

Hybrid approaches that combine exact optimization and evolutionary search provide a promising compromise. Prior work has shown that the combination of evolutionary algorithms with decomposition or exact mathematical programming can improve search efficiency and solution quality [20]. In the context of GEP, hybrid approaches such as GA with Benders decomposition and metamodel-assisted evolutionary algorithms have already been proposed [12, 22]. However, approaches based on surrogate or metamodel evaluation may introduce approximation errors in the estimation of operational cost and system feasibility [20, 22].

Parallel computing opens another path to scalability. Recent studies in energy system optimization and capacity expansion have shown that decomposition combined with parallel computing can substantially reduce solution time and enable the treatment of much larger planning problems [4, 23, 24]. Nevertheless, the literature still provides limited evidence on how massively parallel exact evaluation of operational MILP subproblems can be combined with population-based evolutionary search for GEP with large shares of variable renewable generation [4, 12, 20].

Against this background, the present study investigates a hybrid decomposition-based method in which the investment structure is searched by an evolutionary algorithm, and each candidate solution is evaluated through exact operational MILP models solved in parallel. In contrast to metamodel-assisted approaches, the proposed method does not replace the operational model with a surrogate approximation. This is important for planning problems in which simplified evaluation may lead to decisions that are economically nonoptimal or technically infeasible under realistic operating conditions [4, 22].

Compared with prior studies, the proposed framework differs in several key respects. Unlike classical Benders decomposition, it does not depend on cut generation from a continuous operational subproblem, which is restrictive when system operation is represented by MILP models with integer variables [4]. Unlike metamodel-assisted evolutionary approaches, it uses direct operational MILP evaluation and therefore does not introduce additional surrogate-model approximation error into objective assessment [22]. In addition, the paper compares Evolutionary Centers Algorithm (ECA) and Covariance Matrix Adaptation Evolution Strategy (CMA-ES) under the same problem formulation and computational setting and treats parallel high-performance computing (HPC) execution as a core methodological component of the framework [23, 24].

In this work, we investigate a hybrid method for optimizing the generation capacity structure of an electric power system that includes solar, wind, thermal, and storage technologies. The optimization is carried out over a five-year planning horizon with the objective of satisfying forecasted demand while minimizing the total cost of investment and operation. From the perspective of artificial intelligence (AI) applications, the proposed hybrid method combines evolutionary search in the investment space with exact MILP-based operational verification of candidate solutions, which is consistent with the broader use of AI and metaheuristic approaches in power system operation, control, and planning, as well as with established GEP formulations in the literature [5, 25].

To this end, a comparative analysis is conducted to assess the performance of two modern evolutionary algorithms, the ECA [26] and the CMA-ES [18]. Both approaches are implemented together with decomposition of the original problem and parallel solution of the resulting subproblems using the Solving Constraint Integer Programs (SCIP) solver [27], enabling high computational performance and scalability. In the broader context of intelligent power systems, this positioning is aligned with the growing use of AI not only for forecasting and control but also for energy management, optimization, and planning in systems with high shares of renewable generation, storage, and flexibility requirements [25, 28].

The main innovation of this study is the development and evaluation of an AI-enabled hybrid method for GEP in power systems with high shares of variable RES. More specifically, the method combines evolutionary search as an AI component at the investment-planning level with exact MILP solution of operational subproblems at the evaluation level, thereby preserving detailed operational feasibility without replacing the operational model by a surrogate approximation, in contrast to metamodel-assisted approaches proposed in the literature [22]. In this sense, the proposed method may be viewed as a decision-support core for future intelligent energy systems that integrate renewable generation, energy storage, and advanced digital management layers. Operational adequacy is assessed under representative scenarios of electricity demand and renewable-generation variability, so that expansion decisions reflect flexibility requirements under realistic high-renewable conditions.

2. Methods

2.1. Mathematical model

In this study, a more comprehensive version of the local electric grid model, relative to that described in [29], was employed. All parameters and variables of the model are presented in Tables 1 and 2.

The objective function to be minimized is the total cost of electricity generation, that is, the sum of capital investments and current operational costs:

$$\sum_{i\in S\cup R\cup F} \bar{P}_i^{\text{tech}} C_i^{\text{con}} N_i + \sum_{z\in Z} \tau_z \sum_{k\in K} \frac{\Lambda_{k,z}}{\Lambda_z} \sum_{t\in W_{k,z}} \left(\sum_{i\in S\cup R\cup F} \text{Cost}_{i,t} + \sum_{i\in F} (\text{Cost}_{i,t}^{\text{SU}} + \text{Cost}_{i,t}^{\text{SD}}) + \text{Cost}_t^{\text{Ext}} - \text{Income}_t \right) \to \min. \quad (1)$$

Capital investments are constrained by the established limit:

$$\sum_{i\in S\cup R\cup F} \bar{P}_i^{\text{tech}} C_i^{\text{con}} N_i \le \text{Invest}. \quad (2)$$

The balance between electricity generation and consumption in the local electric grid is represented by the following equation:

$$\sum_{i\in R\cup F} p_{i,t} + \sum_{i\in S} p_{i,t}^{\text{G}} + p_t^{\text{Ext}} = l_t + \sum_{i\in S} p_{i,t}^{\text{P}} + p_t^{\text{Int}}, \ \forall t \in T = W_{k,z}, \ \forall z \in Z, \ \forall k \in K. \quad (3)$$

The load of transmission lines connecting the local electric grid with the national electric power system is constrained as follows:

$$0 \le p_t^{\text{Ext}} \le h_t H, \ \forall t \in T = W_{k,z}, \ \forall z \in Z, \ \forall k \in K, \quad (4)$$

Table 1
GEP model parameters

Parameter	Description
S, R, F	Sets of available types of ESS, RES, and TPP
$\bar{P}_i^{\text{tech}}$	Installed capacity of a power plant of type i
C_i^{con}	Capital costs for constructing one unit of equipment of type i
$N_i^{\max}$	Maximum number of ESS units of type i
Z	Set of forecast periods z
τ_z	Duration of forecast period z in weeks
Λ_z	Total number of weekly functions within period z
K	Set of clusters of weekly functions
$\Lambda_{k,z}$	Number of weekly functions of cluster k in period z

(*Continued*)

Table 1
(*Continued*)

Parameter	Description
$W_{k,z}$	Set of time intervals of weekly functions of cluster k in forecast period z
$E_{i,t}$	Value of the representative weekly graph of the capacity factor of an RES power plant of type i at time $t \in W_{k,z}$
l_t	Electricity consumption load at time $t \in W_{k,z}$
Invest	Maximum volume of capital investments
H	Maximum permissible load of transmission lines connecting the local grid with the national electric power system
c_i	Specific operational costs of electricity generation by an RES unit
$c_i^{\mathrm{P}}, c_i^{\mathrm{G}}$	Specific operational costs for an ESS of type i in charging and discharging modes
$\eta_i^{\mathrm{P}}, \eta_i^{\mathrm{G}}$	Efficiency of an ESS of type i in charging and discharging modes
$\bar{p}_i^{\mathrm{P}}, \bar{p}_i^{\mathrm{G}}$	Maximum load of an ESS of type i in charging and discharging modes
$\bar{q}_i, \underline{q_i}$	Maximum and minimum amount of energy that can be stored by an ESS of type i
$t_0,\ T_0$	Indices of the leftmost and rightmost time intervals of a weekly period
$\mathrm{Pr}_t^{\mathrm{TL}}$	Electricity price in the wholesale market at time t
$\mathrm{Pr}_t^{\mathrm{TSO}}$	Transmission service tariff at time t
$c_i^{\mathrm{SU}}, c_i^{\mathrm{SD}}$	Startup and shutdown costs, respectively, for a unit of type $i \in F$
$\Delta P_i^{\mathrm{up}},\ \Delta P_i^{\mathrm{down}}$	Maximum permissible ramp-up and ramp-down of load, respectively, for a unit of type $i \in F$
$\overline{P}_i, \underline{P}_i$	Minimum and maximum load, respectively, for a unit of type $i \in F$
$a_i,\ b_i,\ c_i$	Coefficients of the polynomial $a_i p_{i,t}^2 + b_i p_{i,t} + c_i$ that approximates the dependence of operational costs of a unit of type $i \in F$ on its load $p_{i,t}$
$\mathscr{L}_i$	Set of ordinal indices of segments forming the piecewise-linear approximation of the dependence of operational costs of a unit of type $i \in F$ on its load $p_{i,t}$

Table 2
GEP model variables

Variable	Description
N_i	Number of units of type i considered for commissioning
$\mathrm{Cost}_{i,t}$	Operational costs of electricity generation by a unit of type i in time interval t
$\mathrm{Cost}_t^{\mathrm{Ext}}$	Cost of electricity purchase in the wholesale market in time interval t
Income_t	Revenue from electricity sales in the wholesale market in time interval t
$p_{i,t}$	Load of a unit of type $i \in R \cup F$ at time t
$p_{i,t}^{\mathrm{P}}, p_{i,t}^{\mathrm{G}}$	Load of an ESS of type i at time t in charging and discharging modes, respectively.
$p_t^{\mathrm{Int}}, p_t^{\mathrm{Ext}}$	Load for electricity sales and purchases, respectively, in the wholesale market at time t
h_t	Binary function equal to 0 when there is load for electricity sales in the wholesale market, and equal to 1 when there is load for electricity purchases
$q_{i,t}$	Amount of electricity stored in an ESS of type i at time t
$u_{i,t}^{\mathrm{PG}}$	Binary function equal to 0 when an ESS of type i operates in charging mode, and equal to 1 when it operates in discharging mode
$u_{i,t}, y_{i,t}, x_{i,t}$	Integer functions equal to the number of units of type $i \in F$ in generation, startup, and shutdown modes, respectively, at time t
$\tilde{p}_{i,t}^{\,l}$	Variable component of the load of units of type $i \in F$ at time t
$\mathrm{Cost}_{i,t}^{\mathrm{SU}},\ \mathrm{Cost}_{i,t}^{\mathrm{SD}}$	Startup and shutdown costs, respectively, of units of type $i \in F$ in time interval t

$$0 \le p_t^{\mathrm{Int}} \le (1-h_t)\,H,\ \forall t \in T = W_{k,z},\ \forall z \in Z,\ \forall k \in K. \quad (5)$$

The cost of electricity purchase in the wholesale market is calculated as follows:

$$\begin{aligned}\mathrm{Cost}_t^{\mathrm{Ext}} &= \left(\mathrm{Pr}_t^{\mathrm{TL}} + \mathrm{Pr}_t^{\mathrm{TSO}}\right) p_t^{\mathrm{Ext}},\\ &\forall t \in T = W_{k,z},\ \forall z \in Z,\ \forall k \in K.\end{aligned} \quad (6)$$

The revenue from selling surplus electricity in the wholesale market is determined as follows:

$$\mathrm{Income}_t = \mathrm{Pr}_t^{\mathrm{TL}} p_t^{\mathrm{Int}},\ \forall t \in T = W_{\mathrm{k,z}},\ \forall z \in Z,\ \forall k \in K. \quad (7)$$

2.1.1. ESS description

Operational costs of ESS are defined as:

$$\begin{aligned}\mathrm{Cost}_{i,t} &= c_i^{\mathrm{P}}\left(1-\eta_i^{\mathrm{P}}\right) p_{i,t}^{\mathrm{P}} - c_i^{\mathrm{G}}\left(1-\eta_i^{\mathrm{G}}\right) p_{i,t}^{\mathrm{G}},\\ &\forall t \in T = W_{k,z},\ \forall z \in Z,\ \forall k \in K,\ \forall i \in S.\end{aligned} \quad (8)$$

Energy balances of ESS are expressed as:

$$\begin{aligned}q_{i,t} - q_{i,t-1} &= \eta_i^{\mathrm{P}} p_{i,t}^{\mathrm{P}} - \frac{p_{i,t}^{\mathrm{G}}}{\eta_i^{\mathrm{G}}},\ \forall t \in T = W_{k,z},\ t \ne t_0,\\ &\forall z \in Z,\ \forall k \in K,\ \forall i \in S.\end{aligned} \quad (9)$$

Weekly cyclicity conditions of energy storage processes are given by:

$$q_{i,t_0} = q_{i,T_0},\ \forall t \in T = W_{k,z}, \forall z \in Z,\ \forall k \in K,\ \forall i \in S. \quad (10)$$

The volumes of stored energy are constrained as follows:

$$\begin{aligned}N_i\,\underline{q}_i \le q_{i,t} \le N_i \bar{q}_i,\ &\forall t \in T = W_{k,z},\\ &\forall z \in Z,\ \forall k \in K,\ \forall i \in S.\end{aligned} \quad (11)$$

The available number of ESS units is limited by:

$$\begin{aligned}N_i \le \mathrm{N_i^{max}},\ &\forall t \in T = W_{k,z},\ \forall z \in Z,\\ &\forall k \in K,\ \forall i \in S.\end{aligned} \quad (12)$$

The ESS load in discharging mode is constrained as follows:

$$\begin{aligned}p_{i,t}^{\mathrm{G}} \le u_{i,t}^{\mathrm{PG}} N_i^{\max} \bar{p}_i^{\mathrm{G}},\ &\forall t \in T = W_{k,z},\\ &\forall z \in Z,\ \forall k \in K,\ \forall i \in S,\end{aligned} \quad (13)$$

$$p_{i,t}^{\mathrm{G}} \le N_i \bar{p}_i^{\mathrm{G}},\ \forall t \in T = W_{k,\,z},\ \forall z \in Z,\ \forall k \in K,\ \forall i \in S, \quad (14)$$

$$p_{i,t}^{\mathrm{G}} \ge 0,\ \forall t \in T = W_{k,z},\ \forall z \in Z,\ \forall k \in K,\ \forall i \in S. \quad (15)$$

The ESS load in charging mode is also constrained as follows:

$$\begin{aligned}p_{i,t}^{\mathrm{P}} \le \left(1-u_{i,t}^{\mathrm{PG}}\right) N_i^{\max} \bar{p}_i^{\mathrm{P}},\ &\forall t \in T = W_{k,z},\\ &\forall z \in Z,\ \forall k \in K,\ \forall i \in S,\end{aligned} \quad (16)$$

$$p_{i,\,t}^{\mathrm{P}} \le N_i \bar{p}_i^{\mathrm{P}},\ \forall t \in T = W_{k,z},\ \forall z \in Z,\ \forall k \in K,\ \forall i \in S, \quad (17)$$

$$p_{i,\,t}^{\mathrm{P}} \ge 0,\ \forall t \in T = W_{k,z},\ \forall z \in Z,\ \forall k \in K,\ \forall i \in S. \quad (18)$$

2.1.2. RES description

The operational costs of electricity generation from RES are calculated by the following formula:

$$\begin{aligned}\mathrm{Cost}_{i,t} &= c_i p_{i,t},\ \forall t \in T = W_{k,z},\ \forall z \in Z,\\ &\forall k \in K,\ \forall i \in R.\end{aligned} \quad (19)$$

The load of RES units is determined by the following relation:

$$\begin{aligned}p_{i,t} &= N_i E_{i,t} \bar{P}_i^{\mathrm{tech}},\ \forall t \in T = W_{k,z},\ \forall z \in Z,\\ &\forall k \in K,\ \forall i \in R.\end{aligned} \quad (20)$$

2.1.3. TPP description

The operational costs of electricity generation by TPP generating units are calculated as follows:

$$\begin{aligned}\mathrm{Cost}_{i,t} &= \breve{C}_i u_{i,t} + \textstyle\sum_{l \in \mathscr{L}_i} \tilde{C}_i^l \tilde{p}_{i,t}^l,\ \forall t \in T = W_{k,z},\\ &\forall z \in Z,\ \forall k \in K,\ \forall i \in F,\end{aligned} \quad (21)$$

Where

$$\breve{C}_i = \left(a_i \underline{P}_i + b_i\right) \underline{P}_i + c_i, \quad (22)$$

$$\tilde{C}_i^l = a_i \left[2\underline{P}_i + (2l-1)\,\Delta P_i\right] + b_i, \quad (23)$$

$$\Delta P_i = \left(\overline{P}_i - \underline{P}_i\right) / \left|\mathscr{L}_i\right|. \quad (24)$$

The variable components of the load are constrained as follows:

$$\begin{aligned}\tilde{p}_{i,t}^l \le \Delta P_i u_{i,t},\ &\forall t \in T = W_{k,z},\ \forall z \in Z,\\ &\forall k \in K,\ \forall l \in \mathscr{L}_i,\ \forall i \in F.\end{aligned} \quad (25)$$

The values of total load are computed by the following expression:

$$\begin{aligned}p_{i,t} &= \underline{P}_i u_{i,t} + \textstyle\sum_{l \in \mathscr{L}_i} \tilde{p}_{i,t}^l,\ \forall t \in T = W_{k,z},\\ &\forall z \in Z,\ \forall k \in K, \forall i \in F.\end{aligned} \quad (26)$$

The values of total load are subject to the following constraints:

$$\begin{aligned}\underline{P}_i u_{i,t} \le p_{i,t} \le \overline{P}_i u_{i,t},\ &\forall t \in T = W_{k,z},\\ &\forall z \in Z,\ \forall k \in K, \forall i \in F.\end{aligned} \quad (27)$$

The state balance equations for the sets of generating units operating in startup, load, and shutdown modes are given by:

$$\begin{aligned}&y_{i,t} - x_{i,t}\\ &= \begin{cases} u_{i,t} - u_{i,t-1},\ t \ne t_0, & \forall t \in T = W_{k,z},\ \forall z \in Z,\\ u_{i,t_0} - u_{i,T_0},\ t = t_0, & \forall k \in K, \forall i \in F.\end{cases}\end{aligned} \quad (28)$$

The total number of generating units in startup and shutdown modes is constrained by:

$$y_{i,t} + x_{i,t} \le N_i,\ \forall t \in T = W_{k,z},\ \forall z \in Z,\ \forall k \in K, \forall i \in F. \quad (29)$$

The set of generating units in load mode is also limited by:

$$u_{i,t} \le N_i,\ \forall t \in T = W_{k,z},\ \forall z \in Z,\ \forall k \in K, \forall i \in F. \quad (30)$$

The ramp-up rate of generating units is constrained as follows:

$$\begin{cases} p_{i,t} - p_{i,t-1} \le \Delta P_i^{\text{up}} u_{i,t-1}, & t \ne t_0, \\ p_{i,t_0} - p_{i,T_0} \le \Delta P_i^{\text{up}} u_{i,T_0}, & t = t_0, \end{cases} \quad \begin{matrix} \forall t \in T = W_{k,z}, \ \forall z \in Z, \\ \forall k \in K, \forall i \in F. \end{matrix} \tag{31}$$

The ramp-down rate of generating units is also constrained by:

$$\begin{cases} p_{i,t} - p_{i,t-1} \le -\Delta P_i^{\text{down}} u_{i,t-1}, & t \ne t_0, \\ p_{i,t_0} - p_{i,T_0} \le -\Delta P_i^{\text{down}} u_{i,T_0}, & t = t_0, \end{cases} \quad \begin{matrix} \forall t \in T = W_{k,z}, \ \forall z \in Z, \\ \forall k \in K, \forall i \in F. \end{matrix} \tag{32}$$

The startup operational costs of generating units are computed by:

$$\text{Cost}_{i,t}^{\text{SU}} = c_i^{\text{SU}} y_{i,t}, \ \forall t \in T = W_{k,z}, \ \forall z \in Z, \ \forall k \in K, \forall i \in F. \tag{33}$$

The shutdown operational costs of generating units are determined by:

$$\text{Cost}_{i,t}^{\text{SD}} = c_i^{\text{SD}} x_{i,t}, \ \forall t \in T = W_{k,z}, \ \forall z \in Z, \ \forall k \in K, \forall i \in F. \tag{34}$$

2.2. Model parameters

For each of the four seasons, representative cluster models were constructed for electricity consumption and for SPP and WPP generation, reflecting variations in consumption and generation volumes. Each model is represented by four weekly (168-h) graphs. All electricity consumption graphs have equal weight coefficients, while for SPP and WPP units, the first two graphs correspond to the minimum and maximum efficiency levels of installed capacity utilization, and the other two represent average efficiency levels. The relative weights of the summer and winter graphs of capacity utilization factors for SPP and WPP are given in Table 3, while the graphs themselves are shown in Figures 1, 2, and 3.

Table 3
Weights of representative RES graphs for summer and winter

Equipment type	Summer	Winter
SPP	1, 1, 28, 7	1, 1, 22, 7
WPP	1, 1, 8, 27	1, 1, 10, 19

Since the mathematical model assumes the use of a single set of clusters, 96 graph combinations were generated, integrating SPP and WPP capacity utilization factors with electricity consumption graphs. For each season, six RES efficiency graph combinations were considered: the first corresponds to the minimum efficiency scenario, the second to the maximum efficiency scenario, while the remaining four were formed by combining the third and fourth graphs of SPP and WPP. Each of these six combinations was combined with four consumption graphs, resulting in 24 graph combinations per season and a total of 96 combinations over the entire planning horizon.

The total investment budget was fixed at 1.5 billion USD, which determined the upper limits for the installation of generating equipment, as shown in Table 4. For RES, operational costs were set at 5 USD/MWh. The permissible load of transmission lines was limited to 80 MW. The electricity transmission tariff was 10 USD/MWh. The characteristics of ESS are presented in Table 5, while the parameters of TPP generating units are given in Table 6. The electricity price in the wholesale market was defined by a linear dependence on demand:

$$\text{Pr}_t^{\text{TL}} = 0.163794 \, l_t + 12.24138. \tag{35}$$

Figure 1
Representative weekly graphs of SPP capacity utilization factors, color-coded by clusters, for summer (a) and winter (b)

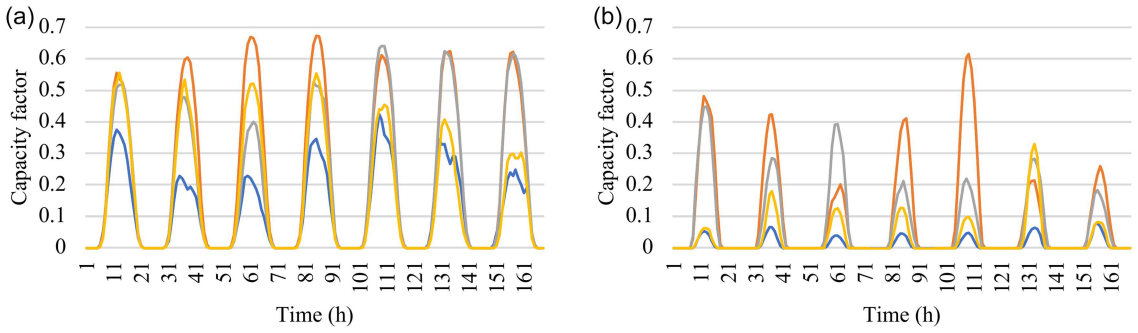


Figure 2
Representative weekly graphs of WPP capacity utilization factors, color-coded by clusters, for summer (a) and winter (b)

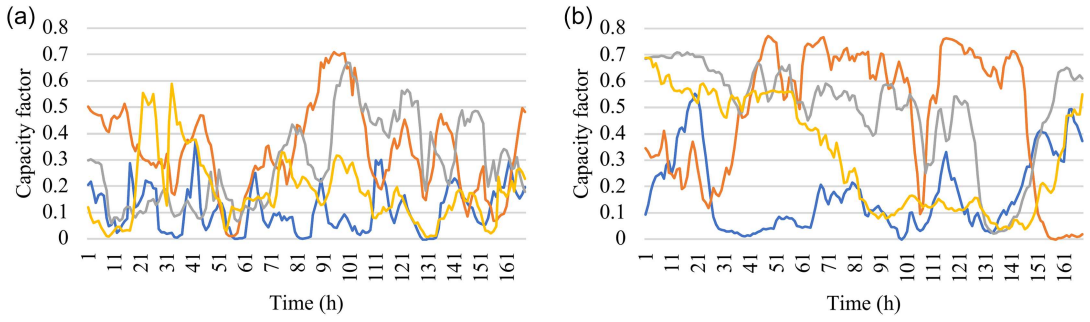


Figure 3
Representative weekly graphs of electricity consumption, color-coded by clusters, for summer (a) and winter (b)

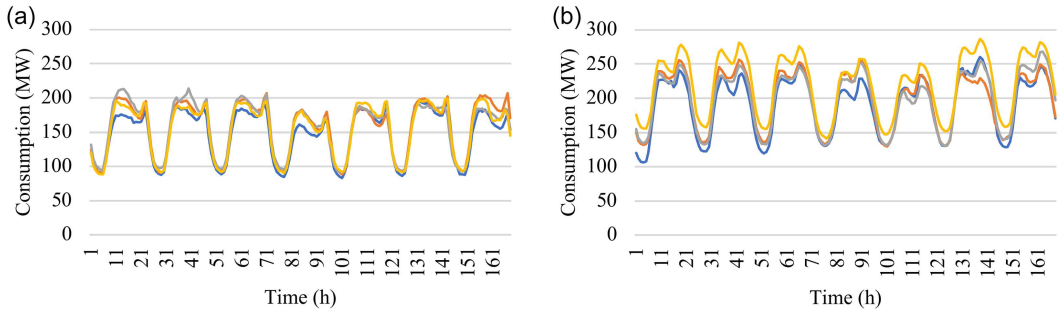


Table 4
General characteristics of candidate equipment

Equipment type	Unit capacity (MW)	Capital costs (USD/kW)	Maximum number of units
ESS	25.00	845.00	71
SPP	20.00	284.81	263
WPP	20.00	1417.72	52
TPP	77.00	5078.00	3

Table 5
ESS parameters used in the model

Parameter	Value
η^P	0.82
η^G	0.85
c^P	30 USD/MWh
c^G	30 USD/MWh
$\bar{q}$	200 MWh
$\underline{q}$	0 MWh
$\bar{p}^P$	25 MW
$\bar{p}^G$	25 MW

Table 6
TPP unit parameters used in the model

Parameter	Value
c^{SU}	1337.49 USD
c^{SD}	1337.49 USD
ΔP^{up}	57.75 MW
ΔP^{down}	57.75 MW
$\overline{P}$	77.00 MW
$\underline{P}$	57.75 MW
a	0.00 USD/(MW²·h)
b	3.00 USD/(MW·h)
c	835.05 USD/h
$\lvert\mathscr{L}\rvert$	2

2.3. Decomposition-based hybrid method

Given the large dimensionality of the generation capacity structure optimization problem, a decomposition method was applied, dividing it into a master problem and a set of independent subproblems subordinated to it.

Within the iterative cycle of the master problem, the optimal quantitative structure of generating equipment is determined, along with the corresponding investment costs for its installation and commissioning. At the subproblem level, operational costs are minimized by searching for the optimal load modes of power equipment. A generalized workflow of the proposed decomposition method of the original problem is shown in Figure 4.

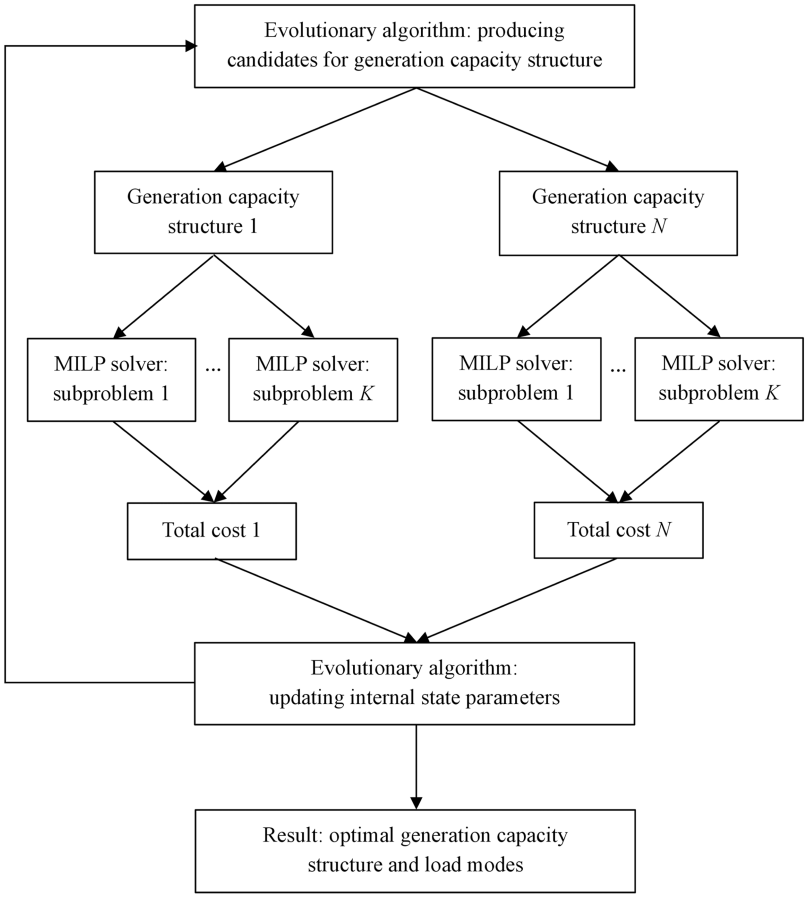
The master problem is solved by an evolutionary algorithm, which at each iteration generates a set of N alternative variants of the number of units of each type. For each of these variants, K subproblems corresponding to different combinations of electricity consumption graphs and capacity utilization factor graphs for SPP and WPP are solved independently and in parallel using the SCIP solver. Based on the results of the subproblem solutions, the total cost is calculated for each equipment structure, including both investment and operational components. These results are then used by the evolutionary algorithm to update the parameters of its internal state and adjust the direction of the subsequent search. The iterative process continues until the defined stopping criteria are met, ensuring the convergence of the algorithm to an approximate global optimum.

2.4. Numerical experiments

The computations were carried out on dedicated nodes of the HPC cluster Vulcan at the High-Performance Computing Center Stuttgart (HLRS) [30]. Each node was equipped with 1024 GB of RAM and two 64-core AMD Zen4 processors running at 2.0 GHz. The algorithms were implemented in the Julia programming language [31] using the JuMP library [32].

At the first stage, baseline measurements of computation time and memory consumption were carried out while solving the

Figure 4
Workflow of the proposed hybrid decomposition-based method



full problem without decomposition using the SCIP solver version 9.2.2 in single-threaded mode. To reduce the impact of random factors, the experiment was repeated ten times.

After the baseline single-threaded SCIP runs, the hybrid method was evaluated in three stages. First, a population-size sensitivity study was performed at 128 threads with caching enabled, using 100 runs for each tested population size. Second, the effect of caching was evaluated only for the default population sizes selected from the sensitivity study again using 100 runs per setting. Third, scalability with respect to the number of threads was analyzed for these default configurations on 16, 32, 64, and 128 threads using the matched set of seeds 1–10 for all thread counts, so that the compared averages were based on the same number of runs. In all experiments, the random number generator was initialized according to the run number (seed), infeasible equipment structures were assigned an objective value of 10^{100}, and the time limit was fixed at 10 h.

2.4.1. ECA algorithm

To evaluate the effectiveness of the ECA algorithm, the Metaheuristics library version 3.3.5 [33] was used, modified by adding support for integer variables in parallel mode through rounding of real values. For variables describing the number of units, the lower bound was set to zero, while the upper bound was determined by investment constraints (Table 4).

For ECA, the number of vectors used to construct the center of mass was fixed at $K = 3$ in all reported experiments. The population size was analyzed separately for $N \in \{60, 80, 100, 120, 140\}$, and the detailed comparison is reported in Section 3. Based on that sensitivity study, $N = 100$ was used as the default configuration in the cache-ablation and thread-scaling experiments. The remaining parameters, including stopping criteria, were kept at their default values or determined automatically.

2.4.2. CMA-ES algorithm

To evaluate the effectiveness of the CMA-ES algorithm, the pycma library version 4.0.0 [34] was used. Variables describing the number of units were defined as integers with a lower bound of zero, while the upper bound was set according to the maximum number of units specified in Table 4. To scale the search space, the parameter CMA_stds was assigned values equal to these maximum quantities. The initial point corresponded to a configuration with zero units.

For CMA-ES, the initial step size was set to $\sigma_0 = 1.0$. Negative covariance matrix updates were disabled (CMA_active=false), trace normalization was enabled (CMA_const_trace=true), and the final reevaluation of the objective function was turned off (eval_final_mean=false). The population sizes of 40, 60, 80, 100, and 120 were analyzed separately, and the detailed comparison is reported in Section 3. Based on that sensitivity study, popsize = 80 was used as the default configuration in the cache-ablation and thread-scaling experiments. The stopping criteria were determined by the automatic settings of the pycma library.

3. Results

According to the results obtained using the SCIP solver without applying the decomposition method, the minimum total cost

amounted to 1.02×10^9 USD for an electric power system consisting of three ESS units, six SPP units, and two TPP units, with no WPP installations. The average solution time was 6644.21 ± 28.37 s, while memory consumption reached 4.51 ± 0.03 GB.

The experimental results are reported in four steps. First, we analyze the influence of population size because this parameter defines the default configurations used in the subsequent cache-ablation and thread-scaling experiments. Second, we quantify the effect of caching for the selected defaults. Third, we analyze scalability with respect to the number of threads using a matched 10-run subset for all thread counts. Finally, we examine the failure modes of unsuccessful runs.

Table 7 summarizes the population-size sensitivity study for both algorithms at 128 threads with caching enabled. Each configuration was evaluated in 100 independent runs, and the reported wall-clock time and peak memory values were averaged over successful runs, that is, runs that reached the reference global optimum.

For CMA-ES, popsize = 80 is the smallest population size that exceeds the 90% reliability threshold, with 98% successful runs and a 3.77× speedup relative to the single-threaded SCIP baseline, whereas larger populations further increase runtime and memory. For ECA, $N = 100$ is the first population size with success above 90%, and larger populations do not improve reliability but increase runtime and memory. These two settings were therefore selected as default configurations for the secondary experiments.

Table 8 reports the cache-ablation results for the default population sizes at 128 threads. Success rate is reported over all 100 runs, whereas wall-clock time, peak memory, and cache-hit statistics are averaged over all runs with recorded values. Timeout runs with missing wall-clock measurements are excluded from the corresponding averages.

Caching has a much stronger effect for CMA-ES than for ECA. For CMA-ES at popsize = 80, disabling the cache increases average runtime from 1746.20 s to 9246.73 s and peak memory from 148.20 GB to 223.18 GB, which is consistent with the high mean cache hit rate of 86.17%. For ECA at $N = 100$, the mean hit rate is only 30.01%, and the effect of disabling the cache is correspondingly smaller.

Table 9 and Figure 5 present the scalability results for the selected default configurations across thread counts. For consistency, the analysis was based on the matched subset of the first 10 runs for each thread count, including 128 threads. Table 9 reports the absolute performance values, whereas Figure 5 shows the corresponding normalized trends.

The speedup values shown in Figure 5 describe normalized thread-scaling behavior within the matched-sample experiment and should not be confused with the 3.77× speedup relative to the single-threaded SCIP baseline reported for CMA-ES with popsize = 80 in Table 7. Under this matched-sample protocol, both default configurations achieved 100% success across all tested thread counts, while runtime decreased and memory usage increased with the number of threads.

Table 10 reports the failure modes of unsuccessful runs for the 128-thread experiments with caching enabled, based on 100 runs for each algorithm. Unsuccessful runs are classified into timeout failures and convergence failures, with infeasible convergence failures representing those converged runs that produced penalized infeasible equipment structures. To preserve interpretability, the average and maximum objective gaps are

Table 7
Effect of population size on convergence and resource usage at 128 threads with caching enabled

Algorithm	Population size	Success rate (%)	Wall-clock time for successful runs (s)	Peak memory for successful runs (GB)
CMA-ES	40	65	1107.78 ± 313.92	120.18 ± 8.97
CMA-ES	60	89	1389.07 ± 290.19	135.86 ± 8.54
CMA-ES	80	98	1761.39 ± 1044.17	148.48 ± 10.76
CMA-ES	100	100	2005.63 ± 396.13	161.28 ± 9.27
CMA-ES	120	100	2120.39 ± 435.09	165.24 ± 11.01
ECA	60	75	1149.43 ± 437.56	145.98 ± 6.53
ECA	80	84	1746.26 ± 2391.36	161.00 ± 5.82
ECA	100	94	1760.91 ± 1883.10	173.54 ± 7.61
ECA	120	92	2329.63 ± 3250.19	183.27 ± 7.34
ECA	140	93	2587.11 ± 3834.36	192.28 ± 7.55

Table 8
Effect of caching for the default population sizes at 128 threads

Algorithm	Population size	Cache mode	Success rate (%)	Mean cache hit rate (%)	Wall-clock time (s)	Peak memory (GB)
CMA-ES	80	ON	98	86.17	1746.20 ± 1039.77	148.20 ± 10.86
CMA-ES	80	OFF	98	–	9246.73 ± 1852.95	223.18 ± 11.34
ECA	100	ON	94	30.01	1657.70 ± 1870.76	164.26 ± 37.67
ECA	100	OFF	93	–	2017.34 ± 1931.81	200.40 ± 47.12

Table 9
Scalability across thread counts for the selected default configurations

Algorithm	Threads	Success rate (%)	Wall-clock time (s)	Peak memory (GB)
CMA-ES	16	100	3145.43 ± 691.05	58.15 ± 3.43
CMA-ES	32	100	2426.09 ± 542.30	69.04 ± 4.43
CMA-ES	64	100	1934.13 ± 424.94	96.96 ± 5.94
CMA-ES	128	100	1557.11 ± 320.06	147.87 ± 8.15
ECA	16	100	3206.34 ± 357.64	65.47 ± 2.33
ECA	32	100	2430.19 ± 260.74	78.67 ± 6.62
ECA	64	100	1986.68 ± 229.13	103.47 ± 5.01
ECA	128	100	1593.66 ± 198.85	173.80 ± 6.77

Figure 5
Speedup (a) and memory consumption ratio (b) versus the number of threads

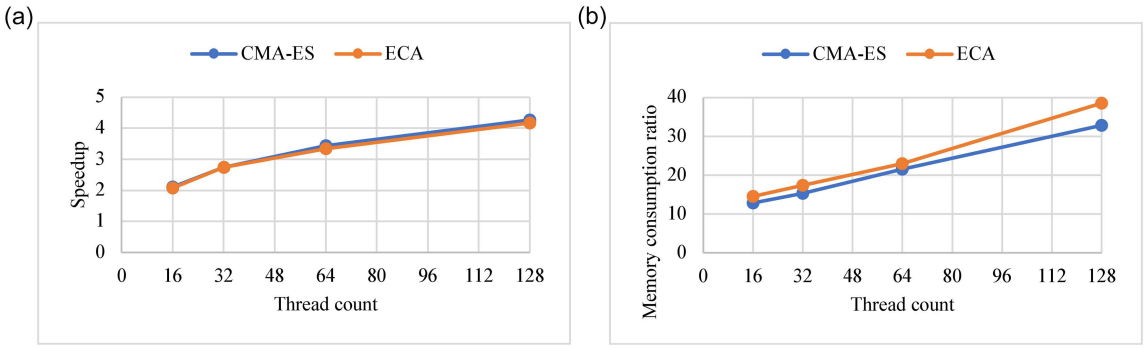


Table 10
Failure modes of unsuccessful runs at 128 threads with caching enabled

Algorithm	Population size	Failed runs	Timeout failures	Convergence failures	Infeasible convergence failures	Avg. gap (%)	Max. gap (%)
CMA-ES	40	35	0	35	0	1.1301	1.2979
CMA-ES	60	11	0	11	0	1.0803	1.2979
CMA-ES	80	2	0	2	0	0.6997	1.2979
CMA-ES	100	0	0	0	0	–	–
CMA-ES	120	0	0	0	0	–	–
ECA	60	25	0	25	25	–	–
ECA	80	16	2	14	14	–	–
ECA	100	6	0	6	6	–	–
ECA	120	8	4	4	4	–	–
ECA	140	7	4	3	3	–	–

reported only for feasible failed runs and are measured relative to the reference solution obtained in the baseline experiment.

The failure patterns of the two algorithms differ qualitatively. For CMA-ES, all unsuccessful runs are convergence failures, with no timeout or infeasible cases, and the selected default configuration, popsize = 80, yields only two failures, with an average gap of 0.70% and a maximum gap of 1.30%. In contrast, ECA failures are dominated by infeasible convergence, while timeout failures occur at three of the five selected population sizes and are absent at the selected default $N = 100$, suggesting that timeout occurrence is irregular across population-size settings.

The modeling also showed that the optimal strategy involves continuously operating TPP generating units at the maximum permissible level, while balancing electricity generation and consumption is achieved through the use of ESS and electricity import/export mechanisms (Figures 6, 7, 8, and 9).

4. Discussion

The results of the computational experiments demonstrated that combining the evolutionary algorithms ECA and CMA-ES with the parallel solution of subproblems using the SCIP solver provides, in most cases, a significant reduction in the time to reach the global optimum compared to using only the SCIP solver in single-threaded mode. Despite comparable average solution times for both algorithms, CMA-ES exhibited a higher probability of convergence to the global optimum. At the same time, the ECA

Figure 6
Weekly RES load graphs, color-coded by cluster combinations, for summer (a) and winter (b)

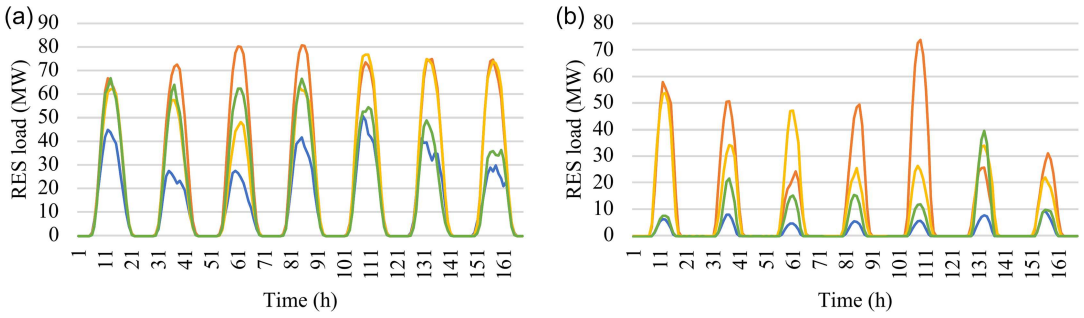


Figure 7
Weekly graphs of stored energy in ESS, color-coded by cluster combinations, for summer (a) and winter (b)

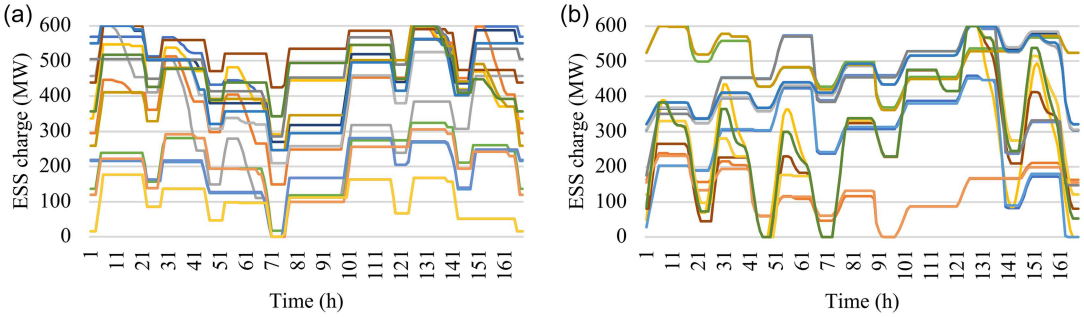


Figure 8
Weekly electricity sales graphs, color-coded by cluster combinations, for summer (a) and winter (b)

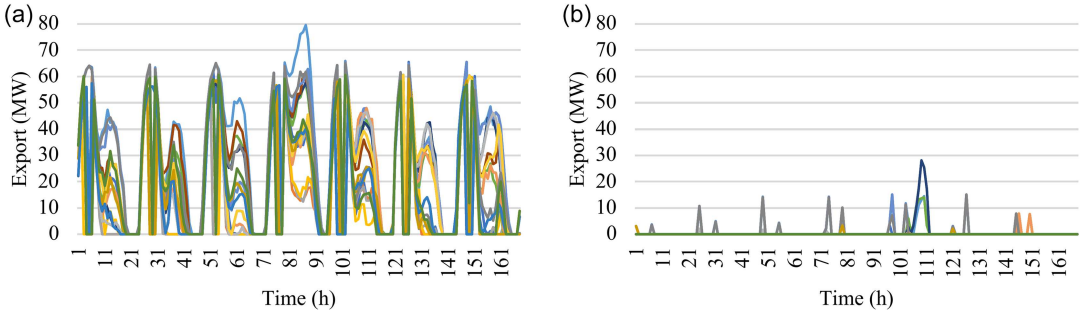
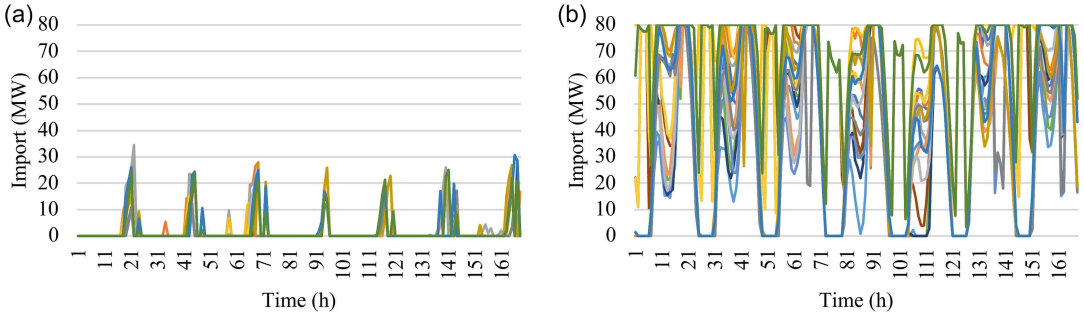


Figure 9
Weekly electricity purchase graphs, color-coded by cluster combinations, for summer (a) and winter (b)



algorithm was characterized by high variability of solution time: the standard deviation exceeded the mean value, and in some cases, the computations did not finish within the allocated time limit (10 hours), which limits its suitability for problems where guaranteed convergence is critical. A likely reason for this is the lack of specific adaptation of the ECA algorithm to work with integer variables, in contrast to CMA-ES. In addition, both solution time and convergence depend on the chosen parameters of the evolutionary algorithm.

The population-size sensitivity analysis shows that the best-performing configuration is not the one with the largest population. For CMA-ES, popsize = 80 provides the smallest population size with reliability above 90%, whereas larger values improve reliability only marginally at the cost of additional runtime and memory. Relative to the single-threaded SCIP baseline, this CMA-ES configuration achieved a 3.77× speedup in average time to the global optimum, where this value refers to the 128-thread population-size experiment with caching enabled and to averages computed over successful runs only. For ECA, $N = 100$ is the first setting to cross the 90% success threshold, while further increases in population size do not improve robustness and instead raise computational cost.

The cache-ablation results indicate that caching is a critical component of the CMA-ES implementation and a secondary acceleration mechanism for ECA. This difference is explained by the much higher reuse of previously evaluated integer structures in CMA-ES, reflected by an average cache hit rate of 86.2% versus 30.0% for ECA. Consequently, disabling caching causes a dramatic slowdown for CMA-ES and only a moderate slowdown for ECA.

The analysis of unsuccessful runs clarifies that the two algorithms fail for different reasons. CMA-ES fails through local stagnation and still returns near-optimal solutions, whereas ECA failures are dominated by infeasible convergence and, at some larger populations, by timeouts. This supports the interpretation that CMA-ES is not only more reliable overall but also more graceful in failure.

The nearly linear growth of performance and memory requirements with the increase in the number of threads is consistent with the typical properties of parallel computing. The high scalability is facilitated both by the use of population-based algorithms and by the application of the problem decomposition method. In particular, in each generation, the CMA-ES algorithm independently evaluates $N = 80$ solutions, for each of which $K = 96$ MILP subproblems are solved, which is equivalent to solving up to 7680 subproblems simultaneously. This ensures flexible utilization of computational resources and allows the proposed parallel method to be applied both on workstations and on multi-node HPC platforms, opening prospects for reducing the solution time of even larger-scale problems.

At the same time, the developed approach to organizing parallel computations has several limitations. First, its application requires a substantial number of processor cores and a large amount of memory. Second, as the number of threads increases, no proportional decrease in solution time is observed, which may be caused by non-uniform solution times across subproblems as well as by the overhead associated with parallelization. Third, the stochastic nature of evolutionary algorithms leads to solution time variability and does not guarantee reaching the global optimum or completing the process within the specified time for each individual run.

Further research is also required to assess the applicability of the proposed approach to multi-period planning problems of electric power system development, in which optimization must account for the dynamics of investment decisions over time. Another challenge is the scalability of the problem decomposition method on multi-node HPC architectures with limited memory per node, which can significantly affect its practical effectiveness.

5. Conclusion

The proposed decomposition-based method for determining the optimal generation capacity structure of an electric power system shows strong potential for scalable implementation on multicore computing systems. Its architecture, which combines an evolutionary algorithm for optimizing the generation capacity structure with a deterministic MILP solver for modeling operating modes, achieves a balanced trade-off between computational efficiency and solution accuracy.

The results of the computational experiments indicate that, for the 128-thread CMA-ES configuration with a population size of 80, the proposed approach achieved a 3.77× speedup relative to the single-threaded SCIP baseline and reached the global optimum in 98 of 100 runs. This validates the effectiveness of the proposed method for solving large-scale MILP problems related to the optimization of the generation capacity structure in electric power systems. In contrast, the use of the ECA algorithm demonstrates a comparable level of speedup but is characterized by lower reliability in reaching the global optimum due to inferior convergence properties.

At the same time, the method also has certain limitations: its practical implementation requires substantial amounts of RAM and powerful computing resources, and it is subject to solution time variability. In addition, the stochastic nature of evolutionary algorithms cannot guarantee convergence to the global optimum.

Further research should focus on two directions: evaluating the effectiveness of the method in multi-period electric power system development problems and assessing its applicability in distributed computing environments, along with analyzing its sensitivity to the selection of key algorithm parameters.

Overall, the combination of evolutionary algorithms with classical MILP solvers can be regarded as a promising, fast, flexible, and sufficiently accurate tool for solving strategic planning problems of the generation capacity structure in electric power systems.

Funding Support

The research was conducted within the framework of project No. 2025.07/0204, "Parallel Methods and Algorithms for Solving Mixed-Integer Linear Programming Problems in the Planning of Structurally Flexible and Resilient Power Systems Development in Ukraine," funded by the National Research Foundation of Ukraine.

Ethical Statement

This study did not contain any studies with human or animal subjects performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

The data that support the findings of this study are openly available in Zenodo at https://doi.org/10.5281/zenodo.20214170.

Author Contribution Statement

Sergii Saukh: Conceptualization, Methodology, Validation, Formal analysis, Resources, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration, Funding acquisition. **Taras Puchko:** Methodology, Software, Investigation, Data curation, Writing – review & editing, Visualization.

References

[1] Pedraza, J. M. (2024). China toward a green economy: Current situation and perspective in the use of different energy sources for electricity generation. *Academia Green Energy*, *1*(1), 1–28. https://www.doi.org/10.20935/AcadEnergy6236

[2] Lefstad, L., Allesson, J., Busch, H., & Carton, W. (2024). Burying problems? Imaginaries of carbon capture and storage in Scandinavia. *Energy Research & Social Science*, *113*, 1–13. https://doi.org/10.1016/j.erss.2024.103564

[3] Pan, J., & Liu, T. (2022). Optimal scheduling for unit commitment with electric vehicles and uncertainty of renewable energy sources. *Energy Reports*, *8*, 13023–13036. https://doi.org/10.1016/j.egyr.2022.09.087

[4] Jacobson, A., Pecci, F., Sepulveda, N., Xu, Q., & Jenkins, J. (2024). A computationally efficient Benders decomposition for energy systems planning problems with detailed operations and time-coupling constraints. *Informs Journal on Optimization*, *6*(1), 32–45. https://doi.org/10.1287/ijoo.2023.0005

[5] Li, C., Conejo, A. J., Liu, P., Omell, B. P., Siirola, J. D., & Grossmann, I. E. (2022). Mixed-integer linear programming models and algorithms for generation and transmission expansion planning of power systems. *European Journal of Operational Research*, *297*(3), 1071–1082. https://doi.org/10.1016/j.ejor.2021.06.024

[6] Morales-España, G., Ramírez-Elizondo, L., & Hobbs, B. F. (2017). Hidden power system inflexibilities imposed by traditional unit commitment formulations. *Applied Energy*, *191*, 223–238. https://doi.org/10.1016/j.apenergy.2017.01.089

[7] Bacci, T., Frangioni, A., Gentile, C., & Tavlaridis-Gyparakis, K. (2024). New mixed-integer nonlinear programming formulations for the unit commitment problems with ramping constraints. *Operations Research*, *72*(5), 2153–2167. https://doi.org/10.1287/opre.2023.2435

[8] Liu, H., Li, H., Liu, H., Gu, C., Li, Q., & Ren, Q. (2024). A closed–loop representative day selection framework for generation and transmission expansion planning with demand response. *Energy Conversion and Economics*, *5*(2), 93–109. https://doi.org/10.1049/enc2.12114

[9] Saukh, S., & Borysenko, A. (2024). Cluster and representative models for generation units of flexible grids with small modular reactors. *Nuclear and Radiation Safety*, *1*(101), 49–58. https://doi.org/10.32918/nrs.2024.1(101).05

[10] Vojvodic, G., Novoa, L. J., & Jarrah, A. I. (2023). Experimentation with Benders decomposition for solving the two-timescale stochastic generation capacity expansion problem. *EURO Journal on Computational Optimization*, *11*, 1–52. https://doi.org/10.1016/j.ejco.2023.100059

[11] Yagi, K., & Sioshansi, R. (2024). Nested Benders's decomposition of capacity-planning problems for electricity systems with hydroelectric and renewable generation. *Computational Management Science*, *21*(1), 16. https://doi.org/10.1007/s10287-023-00469-9

[12] Sirikum, J., Techanitisawad, A., & Kachitvichyanukul, V. (2007). A new efficient GA-benders' decomposition method: For power generation expansion planning with emission controls. *IEEE Transactions on Power Systems*, *22*(3), 1092–1100. https://doi.org/10.1109/TPWRS.2007.901092

[13] Guo, C., Bodur, M., & Papageorgiou, D. J. (2022). Generation expansion planning with revenue adequacy constraints. *Computers & Operations Research*, *142*, 105736. https://doi.org/10.1016/j.cor.2022.105736

[14] Pecci, F., & Jenkins, J. D. (2025). Regularized Benders decomposition for high performance capacity expansion models. *IEEE Transactions on Power Systems*, *40*(4), 3105–3116. https://doi.org/10.1109/TPWRS.2025.3526413

[15] El-Ela, Abou., A, A., El-Sehiemy, R. A., Shaheen, A. M., Shalaby, A. S., & Mouwafi, M. T. (2024). Robust generation expansion planning in power grids under renewable energy penetration via honey badger algorithm. *Neural Computing and Applications*, *36*(14), 7923–7952. https://doi.org/10.1007/s00521-024-09485-5

[16] Pathak, K., Chaudhary, S., Bhandari, M. P., Sharma, P., Poudel, N., & Neupane, D. (2024). Optimal generation expansion planning model for solar PV generation on run of river-based hydro-based power system using binary genetic algorithm. *International Journal of Low-Carbon Technologies*, *19*, 1315–1322. https://doi.org/10.1093/ijlct/ctae079

[17] Ploskas, N., & Sahinidis, N. V. (2022). Review and comparison of algorithms and software for mixed-integer derivative-free optimization. *Journal of Global Optimization*, *82*(3), 433–462. https://doi.org/10.1007/s10898-021-01085-0

[18] Marty, T., Hansen, N., Auger, A., Semet, Y., & Héron, S. (2024). LB+ IC-CMA-ES: Two simple modifications of CMA-ES to handle mixed-integer problems. In *International Conference on Parallel Problem Solving from Nature*, 284–299. https://doi.org/10.1007/978-3-031-70068-2_18

[19] Gonzalez, M., López-Espín, J. J., Aparicio, J., & Talbi, E. G. (2022). A hyper-matheuristic approach for solving mixed integer linear optimization models in the context of data envelopment analysis. *PeerJ Computer Science*, *8*, e828. https://doi.org/10.7717/peerj-cs.828

[20] Duan, S., Jiang, S., Dai, H., Wang, L., & He, Z. (2023). The applications of hybrid approach combining exact method and evolutionary algorithm in combinatorial optimization. *Journal of Computational Design and Engineering*, *10*(3), 934–946. https://doi.org/10.1093/jcde/qwad029

[21] Akter, A., Zafir, E. I., Dana, N. H., Joysoyal, R., Sarker, S. K., Li, L., . . . , & Kamwa, I. (2024). A review on microgrid optimization with meta-heuristic techniques: Scopes, trends and recommendation. *Energy Strategy Reviews*, *51*, 1–27. https://doi.org/10.1016/j.esr.2024.101298

[22] Vrionis, C., Tsalavoutis, V., & Tolis, A. (2020). A generation expansion planning model for integrating high shares of renewable energy: A meta-model assisted evolutionary algorithm approach. *Applied Energy*, *259*, 114085. https://doi.org/10.1016/j.apenergy.2019.114085

[23] Zuluaga, T. V., Musselman, A., Watson, J. P., & Oren, S. S. (2024). Parallel computing for power system climate resiliency: Solving a large-scale stochastic capacity expansion problem with mpi-sppy. *Electric Power Systems Research*, *235*, 110720. https://doi.org/10.1016/j.epsr.2024.110720

[24] Wetzel, M., Cao, K. K., & Sasanpour, S. (2025). Understanding the performance impact of a massively parallel solver for energy system optimization models-a computational experiment using the PIPS-IPM++ solver for REMix instances. *Sustainable Energy, Grids and Networks*, 1–16. https://doi.org/10.1016/j.segan.2025.101893

[25] Pandey, U., Pathak, A., Kumar, A., & Mondal, S. (2023). Applications of artificial intelligence in power system operation, control and planning: A review. *Clean Energy*, *7*(6), 1199–1218. https://doi.org/10.1093/ce/zkad061

[26] Mejía-de-Dios, J. A., & Mezura-Montes, E. (2018). A new evolutionary optimization method based on center of mass. In K. Deep, M. Jain, & S. Salhi (Eds.), *Decision science in action: Theory and applications of modern decision analytic optimisation* (pp. 65–74). Springer. https://doi.org/10.1007/978-981-13-0860-4_6

[27] Bolusani, S., Besançon, M., Bestuzheva, K., Chmiela, A., Dionísio, J., Donkiewicz, T., . . . , & Xu, L. (2024). The SCIP optimization suite 9.0. arXiv. https://doi.org/10.48550/arXiv.2402.17702

[28] Banad, Y. M., Sharif, S. S., & Rezaei, Z. (2025). Artificial intelligence and machine learning for smart grids: From foundational paradigms to emerging technologies with digital twin and large language model-driven intelligence. *Energy Conversion and Management: X*, *28*, 1–35. https://doi.org/10.1016/j.ecmx.2025.101329

[29] Saukh, S., & Puchko, T. (2024). Parallel optimization of generation capacity structure in a local grid under military threats. In *2024 14th International Conference on Dependable Systems, Services and Technologies*, 1–8. https://doi.org/10.1109/DESSERT65323.2024.11122148

[30] Platforms, HLRS. (2026). *Batch System PBSPro (vulcan)*. https://kb.hlrs.de/platforms/index.php/Batch_System_PBSPro_(vulcan)

[31] Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. B. (2017). Julia: A fresh approach to numerical computing. *SIAM Review*, *59*(1), 65–98. https://doi.org/10.1137/141000671

[32] Lubin, M., Dowson, O., Garcia, J. D., Huchette, J., Legat, B., & Vielma, J. P. (2023). JuMP 1.0: Recent improvements to a modeling language for mathematical optimization. *Mathematical Programming Computation*, *15*(3), 581–589. https://doi.org/10.1007/s12532-023-00239-3

[33] Mejía-de-Dios, J. A., & Mezura-Montes, E. (2022). Metaheuristics: A Julia package for single-and multi-objective optimization. *Journal of Open Source Software*, *7*(78), 4723. https://doi.org/10.21105/joss.04723

[34] Hansen, N., Cakmak, S., Kadlecová, G., Abad López, G., Nozawa, K., Rolshoven, L., & Brockhoff, D. (2024). *CMA-ES/pycma (r4.0.0)* [Software]. Zenodo. https://doi.org/10.5281/zenodo.13640131

How to Cite: Saukh, S., & Puchko, T. (2026). Hybrid Method for Optimizing the Structure of Power Generation Capacities Using An Evolutionary Algorithm and An MILP Solver. *Artificial Intelligence and Applications*. https://doi.org/10.47852/bonviewAIA62027499