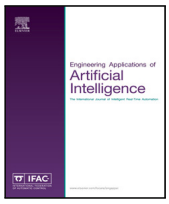
Engineering Applications of Artificial Intelligence 149 (2025) 110350

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



Research paper

Goal-oriented graph generation for transmission expansion planning

Anna Varbella [a], Blazhe Gjorgiev [a], Federico Sartore [a], Enrico Zio [b,c], Giovanni Sansavini [a,*]

[a] Reliability and Risk Engineering Laboratory, Institute of Energy and Process Engineering, Department of Mechanical and Process Engineering, ETH Zurich, Switzerland

[b] Dipartimento di Energia, Politecnico di Milano, Italy

[c] Center for research on Risk and Crises, CRC, Mines Paris, PSL, France

ARTICLE INFO

Keywords:
Transmission expansion planning
Power grid
Cascading failures
Multi-objective optimization
Deep reinforcement learning
Neural networks
Graph representation learning

ABSTRACT

The electrification strategies that are being designed to meet sustainability objectives and rising energy demands pose significant challenges for power systems worldwide and require Transmission Expansion Planning (TEP). This study adopts a risk-informed approach to TEP, formulated as a multi-objective optimization problem that concurrently minimizes systemic risks and expansion costs. Given the intractability of this problem with conventional solvers, we turn to artificial intelligence techniques. In particular, we conceptualize power grids as graphs and introduce a goal-oriented graph generation methodology using deep reinforcement learning. We extend welfare-Q learning, a modified variant of Q-learning tailored to yield high rewards across multiple dimensions, by incorporating geometric deep learning for function approximation. This allows us to account for system security while minimizing grid expansion costs. Notably, system risk is evaluated by incorporating a Graph Neural Network (GNN) cascading failure meta-model into the proposed approach. The TEP method is applied to the IEEE 118-bus system, and the efficacy of this novel technique is compared against the state of the art. We conclude that the deep reinforcement learning method can compete with established methods for multi-objective optimization, identifying expansion strategies that improve system security at reduced costs. Furthermore, we test the robustness of the meta-model against topology changes in the transmission network, demonstrating its applicability to novel grid configurations.

1. Introduction

Energy demand is continually growing due to the electrification of many sectors (IEA, 2023). Correspondingly, the power grid must evolve and grow in a way that guarantees reliability and supply security, also against extreme demand scenarios that could lead to catastrophic events like blackouts. Transmission Expansion Planning (TEP) is the systematic decision-making process focused on expanding the existing power grid and building new assets to enhance its performance. In this context, Transmission System Operators (TSOs) aim to identify the most pertinent and cost-effective upgrades to the network for an efficient TEP. The decision-making process considers the investment costs and the power system security, with adherence to regulatory standards for system security being crucial. Typically, the N-1 security criterion (Vrakopoulou et al., 2013) guarantees secure power grid operation, including an unplanned single-element outage. Recent standards from the North American Electric Reliability Corporation (NERC) (Standard TPL-001-4 NERC, 2015) require going beyond the N-1 criterion, i.e., considering simultaneous outages and addressing cascading failures. The latter is recognized as a primary cause of blackout events (Andersson et al., 2005; Stankovski et al., 2023). Thus, it is essential to identify TEP that balances expansion costs and system security. This renders the decision-making process a multi-objective optimization problem.

Traditional methods for TEP (Gomes and Saraiva, 2019) encompass deterministic models, providing a single solution for a given demand scenario, and probabilistic models, which consider multiple demand scenarios. In Golestani et al. (2010), the authors introduce a TEP approach that simultaneously considers unit commitment in the planning. Subsequently, the research community has broadened the scope of the TEP problem (Lumbreras and Ramos, 2016), addressing dynamic planning across multiple time steps (Dodu and Merlin, 1981) and enhancing network modeling by integrating realistic power flow models (Alhamrouni et al., 2014). Security assessment in TEP often incorporates the N-1 security criterion (Vrakopoulou et al., 2013). In studies such as Akbari et al. (2011), authors apply the N-1 criterion to a multi-stage stochastic TEP model, and in Choi et al. (2007), the N-k contingency criterion is introduced. However, assessing power system security beyond the N-1 criterion entails analyzing the risk of

* Corresponding author.
E-mail address: sansavig@ethz.ch (G. Sansavini).

https://doi.org/10.1016/j.engappai.2025.110350
Received 1 October 2024; Received in revised form 12 January 2025; Accepted 16 February 2025
Available online 11 March 2025

Nomenclature

A_t	Action at time t
B	Total number of branches
C	Total cost of transmission expansion planning
C_j	Cost associated with building branch j
E	Set of edges
G	Cumulative reward
K	Batch size
L	Loss
M	Maximum number of upgrades per corridor
N	Maximum number of branches allowed to be upgraded
Q_{targ}	Target network
RI	Risk Improvement
S_t	State at time t
V	Set of vertices
α	Learning rate
ϵ	probability of choosing to explore
γ	Discount factor
$\mathbf{r_t}$	Reward at time t
$\mathbf{v}$	Vector-valued reward
$\mathcal{A}$	Set of candidate actions
$\mathcal{P}$	Transition dynamics
$\mathcal{R}$	Set of rewards
$\mathcal{S}$	Set of states
$\mathcal{W}$	Feasible set of TEP solutions
π	Policy
τ	Trajectory
$\mathbf{A}$	Weighted adjacency matrix
$\mathbf{H}$	Node embeddings
$\mathbf{X}$	Node feature matrix
θ	Neural network parameters
b_j	Number of upgrades in corridor j
d	hidden dimension
q_π	Q-value function
r_{acc}	Accumulated reward
$risk_{\text{ref}}$	Risk profile before transmission expansion
v_π	Value function
w	Single TEP solution, vector of feasible branch upgrades.

cascading failures, for which selecting an adequate cascading failure model is necessary. Existing literature often employs quasi-steady state models like OPA (Carreras et al., 2002) and Manchester (Nedic et al., 2006) models. Many studies also use a simplified DC power flow model despite the well-known limitations (Kile et al., 2014). A risk-informed combined transmission and generation expansion planning is introduced in Raycheva et al. (2023). Furthermore, in Gjorgiev et al. (2022a), the authors present a cascades-risk-informed TEP that utilizes the Cascades (Gjorgiev et al., 2019) model. The latter expands upon quasi-steady state approaches based on the AC power flow model. The Non-dominated Sorting Genetic Algorithm II (NSGA-II) (Deb et al., 2002) algorithm tackles the cascade-risk informed TEP optimization. The choice of this optimization algorithm is driven by the inherent nature of the TEP optimization problem, which is nonconvex, nonlinear, and involves discrete decision variables. Nevertheless, metaheuristic methods like Deb et al. (2002) are computationally intensive and lack optimality guarantees.

Here, we utilize an innovative approach that involves graph generation techniques (Zhu et al., 2022). These techniques are designed to discover novel graph structures using deep learning. Notably, they have demonstrated success in diverse applications, e.g., molecule graph generation (Mercado et al., 2021; Cao and Kipf, 2018), wearable sensors (Yu et al., 2024). A relevant contribution in this domain comes from You et al. (2018), which couples geometric deep learning and reinforcement learning to generate new graph structures optimized for desired molecular properties. Similarly, Zhang et al. (2024) highlights various combinations of reinforcement learning and geometric deep learning explored in the literature. Moreover, recent progress in reinforcement learning for multi-objective decision-making has been highlighted in Mehta et al. (2022). The paper develops a multi-objective reinforcement learning framework to address the multi-objective traveling salesman problem. The research underscores the efficiency of reinforcement learning techniques in exploring the decision variables space and adeptly covering the Pareto front.

The utilization of Deep Reinforcement Learning (DRL) for power systems remains relatively limited compared to other optimization techniques, as highlighted in the comprehensive review (Massaoudi et al., 2021). Most DRL methods in the power systems domain focus on power systems stability (Yin et al., 2023). Yet, recent works (MingKui et al., 2020) employ DRL for transmission expansion planning. Notably, the problem is framed as a multi-objective optimization task, minimizing costs while ensuring the grid's secure operation. However, the analysis in this study does not extend beyond the conventional N-1 security criterion. Similarly, in Wang et al. (2021), the authors develop a deep reinforcement learning model for multistage and multi-objective TEP on the IEEE 24-bus system. The model uses the N-k security criterion but only applies the DC power flow model. None of these works, however, consider the risk of cascading failures and they are tested only on small-size test systems.

The above literature review shows a gap in the exploitation of Deep Reinforcement Learning (DRL) for TEP that incorporates AC power flows and in-depth system security. This work tackles the research gap by coupling DRL and AC-based cascading failure analysis. We adopt the multi-objective Deep Q-Network (DQN) method, presented in Fan et al. (2023), within a goal-oriented graph generation approach. Specifically, we employ a geometric deep learning network, which we introduce as a Graph Neural Q-network (GNQN), to identify the optimal set of power grid upgrades by maximizing cumulative rewards. The reward encompasses upgrade costs and systemic risks, computed using the cascading failure analysis model Cascades (Gjorgiev et al., 2022b). We assess risks with Cascades, as it relies on an accurate AC power flow model and can produce results that align with recorded data (Li et al., 2018). To improve the computational efficiency of Cascades, we introduce a geometric deep-learning meta-model. We leverage Graph Neural Networks (GNN) to train a computationally efficient regression meta-model of Cascades that predicts the magnitude of demand not supplied (DNS). This is an advancement over prior studies (Varbella et al., 2023), where the output of the cascading failure simulations is typically represented as a binary value, categorizing power grid states as safe if there is no DNS and unsafe otherwise.

This work addresses the following research questions: i) Is it feasible to construct a graph generation model utilizing Reinforcement Learning (RL) to optimize a power grid's structure? Specifically, can the model be tailored to balance investment costs and cascading failure risks? ii) Can we employ a geometric deep learning-based meta-model to assess the risk of cascading failures? Does the meta-model generalize for novel power grid topologies? iii) How effective are reinforcement learning techniques in solving multi-objective optimization problems?

The contributions of this work are as follows: (i) we introduce a model for power grid expansion that leverages goal-oriented graph generation, enabling the creation of new grids with optimized properties; (ii) we showcase the capability of a geometric deep learning meta-model to generalize to new topologies, predicting the output of

a cascading failure model; (iii) we extend Welfare Q-learning by incorporating geometric deep learning for function approximation, proposing GNQN, a novel geometric deep learning architecture for deep Q-learning, (iv) we demonstrate that the presented multi-objective Deep Reinforcement Learning (DRL) technique competes with state-of-the-art methods for multi-objective optimization; and (v) we show that in the online use, the trained agent finds the best compromise solution and it can adapt to power grid topology variations.

The rest of the paper is organized as follows: Section 2 outlines the methodology employed for addressing the TEP using reinforcement learning; Section 3 introduces the case study; Section 4 presents and examines the achieved results, highlighting the diverse contributions of this study; and Section 5 wraps up the article with concluding remarks and summarizes the prospects stemming from this research.

2. Methodology

2.1. Transmission expansion planning: problem definition

We represent Transmission Expansion Planning (TEP) as a multi-objective optimization problem that comprises two objectives (Gjorgiev et al., 2022a), namely, expansion cost and risk improvement (RI). The expansion cost objective serves as an indicator of the financial viability and feasibility of a proposed design, ensuring that the investment remains practical within budgetary constraints. Conversely, the RI objective aims to maximize system security by improving the system's risk profile. Risk improvement is computed as the Euclidean distance between the risk profiles of the system before and after the transmission expansion. This metric reflects how the system's vulnerability to cascading failures and other risks has been mitigated. Together, these two objectives enable a balanced evaluation of the trade-offs between cost-effectiveness and system reliability in the optimization process.

The TEP involves strengthening the existing power transmission corridors.[1] Therefore, branches (transmission lines or transformers) are only added to already existing corridors. We define the feasible set of solutions as $\mathcal{W}$, and we denote a feasible solution by $\mathbf{w}$, which represents a vector of a feasible combination of added branches. We indicate with b_j the number of branches upgraded in corridor j. Considering a generic solution $\mathbf{w} = \{b_1, \dots, b_j, \dots, b_n\}$, the expansion cost objective is computed as the sum of the costs of all individual branch upgrades:

$$c(\mathbf{w}) = \mathbf{C}(\mathbf{w}) \cdot \mathbf{w} = \sum_{j=1}^{n} C_j \cdot b_j \tag{1}$$

where $c(\mathbf{w})$ represents the total cost of TEP, $\mathbf{C}(\mathbf{w})$ is the vector of branch upgrade costs for the branches considered in solution $\mathbf{w}$, C_j is the cost associated with upgrading one branch in corridor j and b_j indicates the number of branches upgraded in corridor j. The risk improvement (RI) objective, which aims to maximize the system security, is expressed as:

$$RI(\mathbf{w}) = \sqrt{\left|risk_{\text{ref}} - risk(\mathbf{w})\right|^2} \tag{2}$$

The RI is computed as the Euclidean distance between the risk profiles of the system before transmission expansion $risk_{\text{ref}}$ (reference system) and $risk(\mathbf{w})$ after grid expansion with the branches in solution $\mathbf{w}$. The expectation is that $\mathbf{w}$ lies below $risk_{\text{ref}}$, indicating a reduced probability of experiencing DNS because of the branch upgrades. We compute the risk profiles using the meta-model of Cascades as detailed in Section 2.1.1. The complete multi-objective formulation is represented as:

$$\underset{\mathbf{w}\in\mathbb{N}^n}{\text{maximize}} \quad -c(\mathbf{w}), RI(\mathbf{w}) \tag{3}$$

[1] Corridor refers to a designated path that connects two nodes (buses) in a power grid with one or multiple parallel branches (transmission lines or transformers).

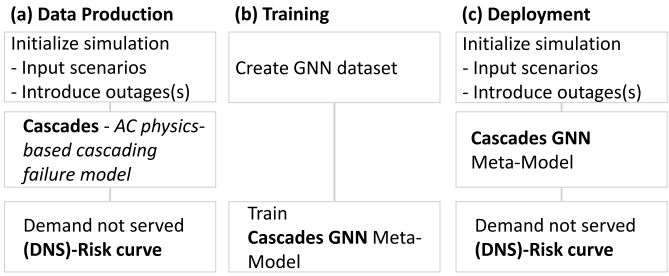


Fig. 1. Complete workflow for developing and utilizing the Cascades meta-model. **(a)** Physics-based Cascades generates the training data. **(b)** The data undergoes processing to make it suitable for training the GNN Cascades meta-model. **(c)** Deployment of the Cascades meta-model to estimate the DNS.

subject to

$$\sum_{j=1}^{n} b_j \leq N \tag{4}$$

$$b_i \leq M \tag{5}$$

where N denotes the maximum number of branches allowed to be upgraded and M is the maximum number of upgrades per corridor, set at eight and one, respectively. The TEP utilizes integer decision variables to expand the system, choosing from a candidate list of branches. Overall, this is a mixed-integer, nonlinear, and non-convex optimization problem and, therefore, demands the appropriate solvers. Here, we utilize DRL to solve the TEP problem by training an agent to identify the new branches to be built. The cascading failure model, Cascades (Gjorgiev et al., 2022b), serves as the DRL environment evaluating the impact of the action space (branch upgrades) on system security. Since the cascading failure analyses are computationally demanding, we develop a model surrogate of Cascades.

2.1.1. Risk improvement via cascading failures meta-model

Cascades is a power system security platform that assesses the power grid's stability under outages (Gjorgiev et al., 2022b). The model uses the pre-outage operating conditions (demand, dispatch, power flows) and a list of contingencies (each comprising single or multiple simultaneous outages). Cascades executes a complete simulation of the post-contingency evolution of the power grid, returning the amount of demand not served (DNS). The DNS, computed for a large set of scenarios (demand and contingencies) is aggregated into the complementary cumulative distribution function (CCDF) of the DNS (Fig. 1(a)). This result is referred to as the risk curve, as it provides the exceedance probability of experiencing DNS greater than the one observed. Running Cascades, like other physics-based cascading failure models (Carreras et al., 2002; Nedic et al., 2006), requires significant computational power (Gupta et al., 2015) and, thus, is unsuitable for online analysis. It was experimentally observed that by coupling traditional cascading failure models for TEP to inform the decision-making about systemic risks, the computational time could reach up to several days on high-performance computers (Gjorgiev et al., 2022a). To address the computational issue, we train a GNN model that can predict the magnitude of DNS given the pre-outage operating conditions and contingency list. The problem is modeled as a graph regression task: each system's initial state is a single graph to which we associate a graph-level label, which is the computed DNS. We employ Cascades (Gjorgiev et al., 2022b) to obtain synthetic data, which we use to train the GNN model (Fig. 1(b)). More details on the meta-model dataset and training are in Appendix B. The meta-model is deployed to infer the risk curve after upgrades (Fig. 1(c)) using the operational data of power grids in its raw form. This allows us to evaluate the risk improvement (Eq. (2)) in a computationally tractable manner. Then, the Cascades meta-model can be used in a computationally intensive DLR framework.

2.2. Transmission expansion planning as a multi-objective Markov decision process

Reinforcement Learning (RL) is a machine learning approach where an agent learns decision-making by interacting with an environment. Modeled as a multi-objective Markov Decision Process (MOMDP), the RL involves an agent observing a state, taking an action, receiving a vectorial reward, and transitioning to a new state. The key elements are the state (current condition), actions (possible decisions), rewards (feedback on the actions, which are vectors), and the environment (external system). The agent aims to learn a policy that maximizes cumulative rewards over time through iterative interactions with the environment.

We adopt the MOMDP modeling of the TEP problem defined in Section 2.1. To this aim, the MOMDP is defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$, where $\mathcal{S}$ is the set of power grid states, $\mathcal{A}$ is the set of candidate branches representing viable actions, $\mathcal{R}$ is the set of possible vectorial rewards after reaching a state, γ is the discount factor tuned to prioritize immediate rewards over long-term rewards, and $\mathcal{P}$ is the transition dynamic (i.e., the outcome after the execution of an action). In MOMDP, a trajectory τ is defined as a sequence of states s_t, actions a_t, and vectorial rewards $\mathbf{r_t}$, i.e., $\tau = (s_0, a_0, \mathbf{r_0}, s_1, a_1, \mathbf{r_1}, \dots, s_T, a_T, \mathbf{r_T})$ for all steps or iterations t inside an episode[2] finishing at T. The behavior of the decision-making agent is modeled via a policy, denoted as $\pi(a_t|s_t) = P(a_t|s_t)$, which dictates the choice of action given the state. The paradigm of reinforcement learning involves finding:

$$\pi^* = \underset{\pi}{\operatorname{argmax}}\, \mathbb{E}[\mathbf{G}(\tau)] \tag{6}$$

The objective is to find the policy that maximizes the discounted cumulative return of the trajectory:

$$\mathbf{G}(\boldsymbol{\tau}) = \sum_{t=1}^{T} \gamma^{t-1} \mathbf{r_t} \tag{7}$$

We define the action-value function or q-value function $\mathbf{q}_\pi(s_t, a_t)$ that is often used to find π^*. Q-value is the expected outcome of taking a specific action in a given state, by estimating the expected future rewards following an optimal policy. Notably, the vector-valued reward translates into vector-valued q-values:

$$\mathbf{q}_\pi(s, a) = \mathbb{E}_\pi[\mathbf{G}(\tau)|s_t, a_t] = \mathbb{E}_\pi[\sum_{t=1}^{T} \gamma^{t-1} \mathbf{r_t}|s_t, a_t] \tag{8}$$

In practice, Q-learning (Watkins and Dayan, 1992) is the algorithmic paradigm used to estimate q-values recursively using the Bellman equation. It is a model-free reinforcement learning method designed to learn the q-value of an action from a specific state. Our problem requires finding the policy π that maximizes $\mathbb{E}_\pi[\mathbf{G}(\tau)]$ in multiple dimensions. Therefore, we adopt the approach from Mossalam et al. (2016), which extends the Q-learning (Watkins, 1989) to Welfare Q-learning that aims to maximize the vector-valued reward ($\mathbf{r_t}$) in multiple dimensions. In Welfare Q-learning, nonlinear scalarized learning updates and non-stationary action selection are leveraged to optimize the multi-dimensional reward. The problem is cast as a fair resource allocation problem, where the resources are the elements of the vector-valued reward. Therefore, the q-values undergo scalarization using a nonlinear fair welfare function, i.e., the Nash Social Welfare function or, simply, geometrical mean, which is chosen for its balance between efficiency and fairness (Mossalam et al., 2016):

$$NSW(\mathbf{q}) = \left(\prod_{i=1}^{n} q_i\right)^{\frac{1}{n}}. \tag{9}$$

A key feature of Welfare Q-learning is the nonstationary action selection, where the agent selects the action that maximizes the total discounted reward, considering past and future states. The agent aims to find the policy $\pi(a|s, \tau)$ that maximizes the expected value, which depends on the episode's trajectory τ and not only on the previous state $s_t = s$. Therefore, action selection is not driven by the immediate reward $\mathbf{r_{t+1}}$, but by the accumulated reward, $\mathbf{r_{acc}} = \mathbf{r_{acc}} + \gamma \mathbf{r_{t+1}}$.

Here, we extend Welfare Q-learning by incorporating geometric deep learning for function approximation. The neural networks for function approximation in Q-learning were first introduced by Mnih et al. (2013), known as deep q-networks (DQN). In this method, the mapping between input states, action, and the q-value is parameterized by a neural network, Q. A target network Q_{targ} stabilizes training by periodically updating its weights θ with those of the original network Q, reducing the potential for divergence during Q-value estimation. Our extension involves using a graph neural network that maps a state to the q-value of each action, namely a Graph Neural Q Network (GNQN). GNQN returns the bi-dimensional q-value approximation associated with each action (upgraded branch) to account for the vectorial reward, thus vectorial $\mathbf{q}$:

$$a = \underset{NSW(\mathbf{q})}{\operatorname{argmax}}\, NSW(\mathbf{q}) \tag{10}$$

We utilize experience replay, which involves storing experiences (consisting of state, action, reward, and next state) in a replay buffer (Mnih et al., 2013). The experience replay randomly samples K transitions $(s_t, a, \mathbf{r_{acc}}, s_{t+1})$ from this buffer during training every 10 episodes. This is done to break the temporal correlation between consecutive experiences in the same trajectories and improve learning efficiency. We balance exploration and exploitation using a $\epsilon - greedy$ policy with linear decay over episodes (see Algorithm 1). The episode always ends when any of the constraints (Eq. (4), Eq. (5)) are violated. The episode can also end if a Pareto non-dominated solution (Debreu, 1954) is found. When the agent discovers a non-dominated solution, the agent is rewarded with a bonus. The value of the bonus is empirically set to 40, which is the order of magnitude of summing the elements of $\mathbf{r_{acc}}$. The value of the bonus makes the algorithm converge faster, but convergence is reached despite the bonus. The full pseudo-algorithm is presented in Algorithm 2.

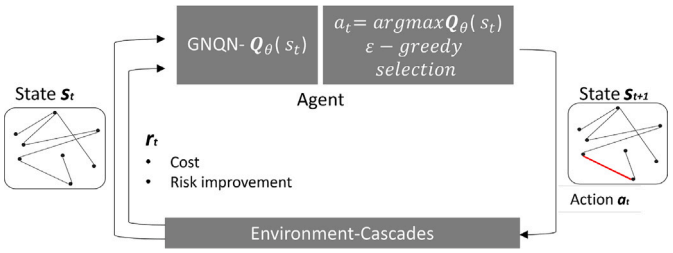


Fig. 2. Overview of the proposed goal-oriented graph generation for TEP. The agent, which is the graph neural q-network (GNQN) detailed in Section 2.2.1, chooses the action, which is the branch highlighted in red; in the environment, we evaluate the reward of taking that action, which is a vector including the RI and cost.

Fig. 2 shows a simplified scheme of the reinforcement learning model used for TEP. We represent power grids as graphs to capture their configuration in the state space $\mathcal{S}$. The nodes and edges correspond to the buses and branches of the power grid, respectively, and are characterized by the attributed graph $G = (V, E, \mathbf{X}, \mathbf{A})$. Here, V is the set of vertices (nodes), E is the set of edges (links), $\mathbf{X} \in \mathbb{R}^{|V| \times f}$ is the node feature matrix (see Appendix A for more details), $|V|$ is the number of nodes, f is the number of features per node, and $\mathbf{A} \in \mathbb{R}^{|V| \times |V|}$ is the weighted adjacency matrix. The a_{ij} elements of the weighted adjacency matrix $\mathbf{A}$ contain the number of branches in the corridor connecting node i and j. This structured approach enables efficient exploration of various power grid configurations. For this purpose, we model the problem as graph generation, which generates new graphs from a distribution similar to the observed graphs. The

[2] An episode refers to a sequence of states, actions, and rewards that starts from an initial state and ends in a terminal state T.

TEP approach considers only branch upgrades, which are the feasible actions available. This translates into a graph generation process that involves only generating new edges. As a result, the action space $\mathcal{A}$ is discrete, with each discrete action involving the addition of a single branch at each step t. Practically, this requires iteratively updating the entries of the weighted adjacency matrix to account for branch upgrades. The node feature matrix $\mathbf{X}$ remains unchanged during the graph generation procedure. We leverage graph representation learning techniques to address the complexities of our state and action spaces. The procedure governing the state transition dynamics is designed to map the input graph structure and select a branch expansion action, as detailed in Section 2.2.1.

Algorithm 1: $\epsilon - greedy$ with linear decay

input : episode, No. episodes, $\mathbf{q}(s_t)$
output: action
ϵ_{final}=0, $\epsilon_{initial} = 1$
$\epsilon = max(\epsilon_{final}, \epsilon_{initial} - (\epsilon_{initial} - \epsilon_{final}) * \frac{episode}{No.episodes})$
if $\epsilon > rand(1)$ **then**
  Select random action
else
  Select action with the maximum q-value- equation (10)
end

Algorithm 2: Goal-oriented TEP pseudo-code

input : α learning rate, γ discount factor, No. episodes, batch size K
Initialize parameters θ for $\mathbf{Q}$ and $\theta_{targ} \leftarrow \theta$ for $\mathbf{Q_{targ}}$;
Initialize empty memory replay buffer B;
output: Pareto front
for *episode* = 1 *to* $No.$ *episodes* **do**
  reset environment:
  Power grid without upgrades (s_0);
  done = False
  $\mathbf{r_{acc}} = \mathbf{0}$
  while *not* ***done*** **do**
    Select action a_t with alg. 1: ϵ-greedy policy with linear decay w.r.t. $\mathbf{Q}$
    $s_{t+1}, \mathbf{r_{t+1}}$, **done** = step into environment(a_t)
    if *Pareto non dominance* **then**
      done = True
      $\mathbf{r_{t+1}} = \mathbf{r_{t+1}}$ + bonus
    else if *constraints violation* **then**
      done = True
    else
      done = False
    end
    $\mathbf{r_{acc}} = \mathbf{r_{acc}} + \gamma \mathbf{r_{t+1}}$
    Insert transition $(s_t, a_t, \mathbf{r_{acc}}, s_{t+1})$ in the buffer B
    Sample K transitions $(s_t, a, \mathbf{r_{acc}}, s_{t+1})$
    Compute loss function over the batch of experience:

$$L(\theta) = \frac{1}{|K|} \sum_{i=1}^{K} [r_{acc,i} + \gamma \mathbf{Q_{targ}}(s_{i,t+1}|\theta_{targ}) - \mathbf{Q}(s_{i,t}|\theta)]^2$$

    Update GNQN parameters $\theta \sim \theta - \alpha \nabla L(\theta)$
    Every 10 episodes update $\theta_{targ} \leftarrow \theta$
  end
end

2.2.1. Graph neural Q-network

Geometric deep learning has significantly advanced non-Euclidean domain applications, with Graph Neural Networks (GNNs) playing a central role in this progress. GNNs are employed in diverse fields such as bioinformatics (Zhang et al., 2021), chemistry (Godwin et al., 2022) and social networks (Guo and Wang, 2021). They learn encoded node representations through message passing in graph structures (Hamilton, 2020). The GNQN adapts this approach to the Q-learning paradigm, utilizing a message-passing layer (MPL) for node embedding. The MPL iteratively aggregates features of neighboring nodes to learn encoded node representations. In our formulation, GNQN is employed for link prediction, aiming to assign a q-value to each branch. From the q-values, we identify the optimal power grid upgrade, which is the branch with maximal q-value. Link prediction is a prominent GNN application, applied in various domains such as citation networks (Subramonian et al., 2023), recommender systems (Koren et al., 2009) and molecular generation (You et al., 2018). The GNQN architecture for link prediction is illustrated in Fig. 3.

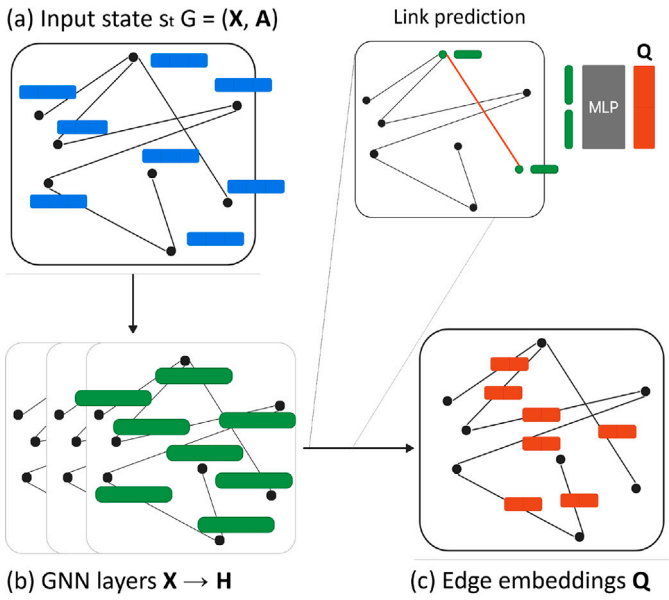


Fig. 3. The GNQN architecture for link prediction consists of Message Passing Layers with ReLU activation on the graph input G, generating node embeddings $\mathbf{H}^{|V|\times d}$, shown in **(a)** and **(b)**. The final edge embeddings $\mathbf{Q}$ are depicted in **(c)**. The zoomed-in section illustrates the link prediction process for a specific edge marked in red. Although the zoom focuses on this single edge for clarity, the link prediction process is simultaneously performed for all edges in the graph.

The GNQN processes the input state (Fig. 3**(a)**), represented as a graph G with node features (in blue), to produce a Q-value matrix (Fig. 3**(c)**). To achieve this, it employs three Message Passing Layers (MPLs) [47], each followed by a Rectified Linear Unit (ReLU) activation function, resulting in the node embedding $\mathbf{H}^{|V|\times d}$, where d is the number of neurons set to 128 (Fig. 3**(b)**). The hyperparameters, such as the number of MPLs and the embedding dimensionality d, are selected using a grid search over 1, 2, and 3 MPLs and d values 32, 64, and 128. The final configuration of three MPLs and $d = 128$ is chosen because it balances performance and computational efficiency. Specifically, GNNs with more than three layers tend to oversmooth the node embeddings resulting in diminished model performance (Chen et al., 2020). Furthermore, increasing the dimensionality d beyond 128 did not yield significant improvements. The edge embeddings $\mathbf{Q}$ (Fig. 3**(c)** in red) are generated by pairwise concatenating adjacent node embeddings and passing them through a 3-layer feed-forward neural network with ReLU activation functions. Our experiments show that increasing the size and width of the network did not improve the performance. Thus, $\mathbf{Q} = GNQN(s_t; \theta)$, where $\mathbf{Q}^{|E|\times 2}$ is the Q-value matrix, with each entry representing a bi-dimensional vector of edge embeddings, reflecting the dimension of our objective function. The GNQN, parameterized by θ, operates on the input graph s_t to compute these values.

3. Experimental setup and test system

We test the developed TEP methodology on the IEEE 118-bus benchmark system (Texas A&M University Engineering, 2022). This benchmark system comprises 186 branches, 118 nodes, 54 generators and a mean power demand of 3733 MW. The annual load curve, sourced from U.S. Energy Information Administration (2019), forms the basis of our simulations. All branches identify the candidate corridors for expansions and only one branch upgrade is allowed per corridor. We estimate the costs of upgrading existing branches using the methodology outlined in Zhang et al. (2013). To evaluate the risk of cascading failures, we simulate 1000 contingencies, including single and multiple simultaneous line failures, over a power grid with 186 lines. These scenarios were designed to cover a wide range of potential contingency scenarios effectively. From the yearly load curve of the IEEE 118-bus system (U.S. Energy Information Administration, 2019), we selected 18 representative hours, focusing on conditions with higher load, which are most likely to reflect worst-case grid stress. After removing duplicate contingencies, 4806 cascading failure simulations remained in total. While the number of loading conditions may appear limited, this choice allowed us to balance the computational complexity of the analysis. It enabled a direct comparison of our framework with computationally intensive metaheuristic algorithms, ensuring the feasibility and relevance of the results.

The reinforcement learning model is implemented in Python using the Gymnasium package (Towers et al., 2023), Pytorch-geometric (Fey and Lenssen, 2019), and Pytorch (Paszke et al., 2019). The experiments are conducted on a Windows desktop with an AMD Ryzen Threadripper 3960X 24-Core Processor and an NVIDIA RTX A6000 GPU with 48 GB VRAM. After a grid search over batch sizes (8, 16, 32, 64, 128), the batch size of 32 is selected for the memory buffer. The Adam optimizer is employed with a fixed learning rate of 10^{-4}. The training is carried out for 5000 episodes, where an episode ends if more than eight branches are built or a non-dominated solution is found, as detailed in Algorithm 2. This results in a total of 20292 iterations, each involving the evaluation of the objective function. Given that running the cascades meta-model for 4806 scenarios takes approximately 3 s and RL performs a total of $9.7 \cdot 10^7$ cascading failure simulations, the entire training lasts about 17 h.

4. Results

4.1. Goal-oriented graph generation

Fig. 4 shows the Pareto front achieved through the goal-oriented graph generation framework. It comprises 14 non-dominated solutions that balance two conflicting objectives: cost and risk improvement. Notably, the risk improvement metric is reported as a negative value. Each point on the Pareto front represents a set of upgrades. The marker types indicate the number of branches per point, whereas the color coding depicts the size of the worst (maximum) cascading failure scenario for that upgrade. The number of branch upgrades is directly linked to the cost of each proposed solution. As expected, more economical expansion leads to less secure solutions, indicating lower risk improvement. Conversely, higher expenditure results in a more robust power grid. The Pareto front ranges from the most cost-effective solution (0.91 M USD, −0.0854) to the most risk-averse yet expensive option (34.7 M USD, −1.6824). Table 1 shows a detailed breakdown of the cost and risk improvement values of the obtained solutions.

Fig. 4 shows two clusters of solutions. The first cluster observed for $RI > -0.6$ is associated with upgrading branch 2, which connects bus 4 to bus 5 (U.S. Energy Information Administration, 2019). This line is the second cheapest, after line 182, and provides a risk improvement of −0.4221. The second cluster, where $RI < -1.2$, is associated with upgrading branch 6. Branch 6 connects bus 8 to bus 9 and links the generator at bus 10 to the rest of the grid. We define the best

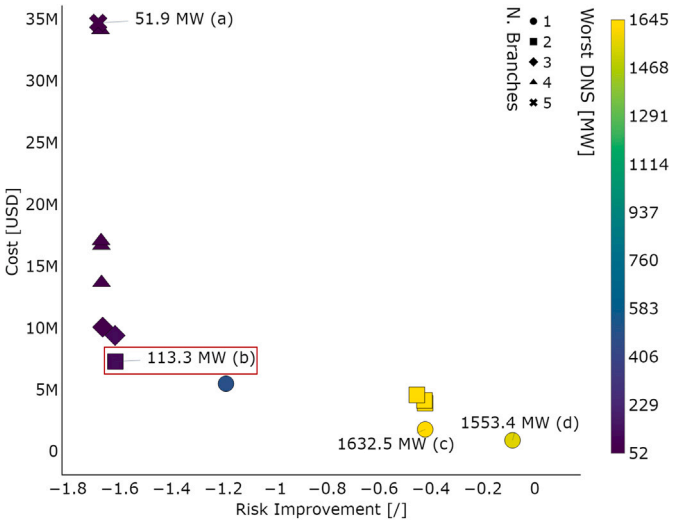


Fig. 4. Pareto front visualization using goal-oriented graph generation for TEP. The color bar indicates the number of branches associated with each solution. The best compromise solution is highlighted by the red box. The risk improvement metric is reported as a negative value.

compromise solution (BCS), i.e., the solution that achieves the highest risk improvement (RI) at the lowest cost, following the approach in Gjorgiev et al. (2022a). We find that the BCS, illustrated in Fig. 4(II), corresponds to the upgrade of branches 2 and 6. These upgrades play a major role in mitigating systemic risks at a cost of 7.28 million USD. Table 1 further quantifies these observations, demonstrating how consecutive improvements in RI stem from combining branch upgrades, such as branches 2 and 6. The marginal benefits of additional upgrades beyond the BCS become evident as costs increase significantly while yielding only minor improvements in risk. For instance, while additional upgrades may slightly improve grid security, the associated costs become increasingly prohibitive. This illustrates the framework's ability to identify critical branches, such as branches 2 and 6, which provide the most significant security gains at lower costs.

The risk profile across different risk improvements is presented in Fig. 5. Fig. 5**a** shows the most secure (most expensive) with a worst-case scenario of 51.9 MW of DNS. Fig. 5**d** shows the most cost-effective (least secure) solution with a worst-case scenario of 1553.4 MW of DNS. These extremes underscore the trade-offs inherent in prioritizing cost versus security. Fig. 5**b** focuses on the best compromise solution (BCS), which achieves a balanced trade-off between cost and risk. This solution, with a worst-case scenario loss of 113.3 MW of DNS, demonstrates the framework's ability to provide secure yet economically viable options. Compared to the most secure solution, the BCS offers a risk improvement (RI) that is only marginally lower with a significant cost reduction. In fact, the RI is reduced by only 4%, while the cost is reduced by 27.4 million USD (approximately a 79% cost savings). Despite this slight compromise in security, the BCS provides a practical option for decision-makers aiming to achieve significant risk mitigation without incurring disproportionately high costs. The solution in Fig. 5**c** achieves a higher risk improvement compared to the cheapest solution in Fig. 5**d**; however, it results in a larger worst-case scenario DNS of 1632.5 MW. This underscores the importance of evaluating the entire risk profile rather than focusing exclusively on extreme scenarios, such as the worst-case DNS or the highest cost.

4.2. Robustness of the cascades meta-model in generalizing to new topologies

The risk curves are generated through a meta-model of the cascading failure simulator (Cascades), developed using deep learning

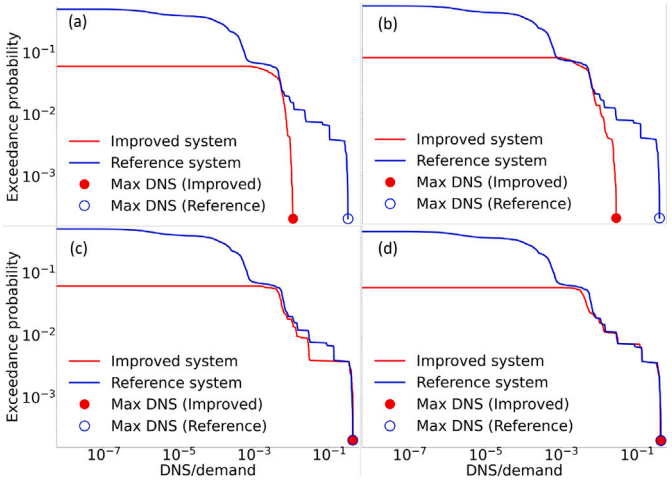


Fig. 5. Complementary Cumulative Distribution Functions (Risk Curves) for the four solutions (a), (b), (c), (d) in Fig. 4 and in Table 1. The four panels are labeled as the four solutions and represent decreasing costs.

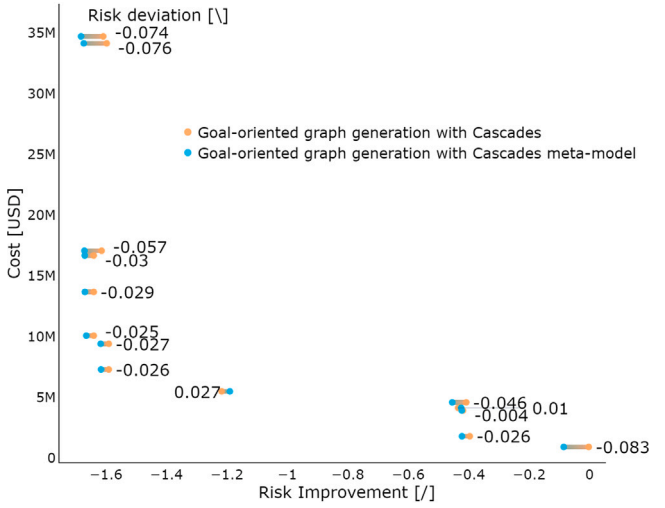


Fig. 6. Comparison of the Pareto fronts obtained using the Cascades meta-model and the original physics-based Cascades model.

Table 1
Goal Oriented graph generation Pareto front solutions. We assign labels to relevant solutions, as highlighted in Figs. 4, 7 and C.9. We indicate the branch indexes associated with each solution, β_i is the branch index for the element i of a solution vector.

Label	Risk Improvement	Cost (USD)	β_1	β_2	β_3	β_4	β_5
(a) (I)	−1.6824	3.472×10^7	0	2	6	106	120
	−1.6734	3.416×10^7	6	7	53	152	
	−1.6713	1.706×10^7	2	6	39	120	
	−1.6701	1.667×10^7	2	6	120	128	
	−1.6696	1.368×10^7	2	6	103	120	
	−1.6651	1.008×10^7	2	6	120		
	−1.6175	9.400×10^6	2	6	49		
(b) (II)	−1.6163	7.286×10^6	2	6			
	−1.1899	5.490×10^6	6				
	−0.4542	4.585×10^6	2	120			
	−0.4247	4.135×10^6	2	181			
	−0.4226	3.911×10^6	2	49			
(c)	−0.4221	1.796×10^6	2				
(d) (III)	−0.0854	9.113×10^5	182				

and trained with diverse topologies of the IEEE 118-bus system as elaborated in Appendix B and Section 2.1.1. However, uncertainties persist regarding the generalization capability of the GNN model to previously unseen topologies. In this study, the topology is modified as new branches are added parallel to the existing ones. To address this concern, we extracted the branches associated with each solution and calculated the risk improvement using the original Cascades model. Therefore, we evaluate the GNN model's capacity to infer a risk curve without specific training on the exact topology.

Fig. 6 shows the Pareto front obtained with the Cascades meta-model in light blue, alongside the Pareto front obtained with the physics-based Cascades in orange. The risk deviation—defined as the difference in estimating risk improvement between the meta-model and the Cascades model—is used to quantify the accuracy of the approximation. In most cases, the meta-model tends to overestimate the risk improvement, suggesting that it is not a conservative model. Nevertheless, the risk deviation remains consistently low, with a maximum of just 4.88% of the total range (0 to 1.7) observed for the risk improvement metric. Moreover, this deviation is consistently low, with no significant outliers, indicating that it could be systematically compensated for in future work. The most substantial overestimation occurs in scenarios involving single-branch upgrades where topological changes are minimal. While this demonstrates the potential of the meta-model to approximate risk improvements effectively, it also reveals a limitation: the model's tendency to overestimate improvements in scenarios with minimal topological changes. This overestimation may result from the reduced sensitivity of the meta-model to subtle variations in grid topology. Addressing this limitation would require enhancing the meta-model's training dataset or employing regularization techniques to improve its generalization capabilities. Despite this, the low deviation across the solution space underlines the meta-model's reliability in approximating risk improvements. This makes it particularly suitable for rapidly identifying promising configurations and evaluating their risk profiles. Indeed, the meta-model offers a significant computational advantage, completing evaluations in just 3 s compared to the 20 s required by the physics-based Cascades model (Gjorgiev et al., 2022a).

A practical application of this framework is to use the meta-model in a two-stage process. First, it can be employed to generate a wide array of potential solutions, quantifying risk profiles efficiently and identifying the most promising candidates. In the second stage, the selected design choices can be validated with the more accurate but computationally intensive physics-based Cascades model. This hybrid approach leverages the computational efficiency of the meta-model while ensuring the final designs adhere to the rigorous standards of the physics-based evaluation. Such a methodology not only accelerates the design process but also enhances its practicality in real-world applications.

4.3. Comparison with a meta-heuristic multi-objective algorithm

This section compares the proposed approach and the NSGA-III algorithm (Deb and Jain, 2014). Table 2 reports the branches to be upgraded associated with the Pareto optimal solutions found by the NSGA-III algorithm. We observe six solutions that are also obtained with our method (see Table 1 and in Fig. C.9). Our framework outperforms the NSGA-III, providing higher risk improvements, whereas the NSGA-III framework finds more solutions than ours. We use the hypervolume indicator (HV) (Auger et al., 2009) to compare the two Pareto fronts. The hypervolume indicator measures the region's volume in the objective space that is dominated by a set of solutions and bounded by a reference point. It is used to compare the performance of different algorithms in multi-objective optimization. A higher hypervolume indicates a better approximation of the Pareto front. The hypervolume from the reference point of zero risk improvement and 5M USD dollars dominated by our approach is greater by 106 units than the NSGA-III. HV is used for post-hoc comparison of the optimization

Table 2
NSGA-III Pareto optimal solutions. We assign labels to relevant solutions, as highlighted in Fig. 7 and C.9. We indicate the branch indexes associated with each solution, β_i is the branch index for the element i of a solution vector.

Label	Risk Improvement	Cost (USD)	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8
(I)	−1.6745	4.309×10^7	2	6	16	39	77	91	120	177
	−1.6743	4.095×10^7	2	6	16	39	91	120	177	
	−1.6740	3.714×10^7	2	6	16	77	91	120	177	
	−1.6738	3.499×10^7	2	6	16	91	120	177		
	−1.6736	3.037×10^7	2	6	39	77	91	120	177	
	−1.6736	2.822×10^7	2	6	39	91	120	177		
	−1.6733	2.441×10^7	2	6	77	91	120	177		
	−1.6731	2.226×10^7	2	6	91	120	177			
	−1.6666	1.764×10^7	2	6	77	120	177			
	−1.6660	1.684×10^7	2	6	91	120				
	−1.6659	1.549×10^7	2	6	120	177				
	−1.6656	1.222×10^7	2	6	77	120				
	−1.6651	1.008×10^7	2	6	120					
(II)	−1.6163	7.286×10^6	2	6						
	−1.1899	5.490×10^6	6							
	−0.4542	4.586×10^6	2	120						
	−0.4239	3.944×10^6	2	77						
	−0.4221	1.796×10^6	2							
(III)	−0.0854	9.113×10^5	182							

frameworks rather than as a guiding metric during optimization. The superior solutions identified by our goal-oriented framework result from the Welfare Q-learning method, which effectively combines the two objectives of minimizing risk and cost. Nevertheless, the Pareto fronts reveal that the branches chosen for upgrades often align between both frameworks. Similarly to our approach, with NSGA-III, the solution positioned above the best compromise solution arises from adding a few branches to a configuration that already yields a significant risk improvement. Fig. 7(I) shows the solution in the Pareto front, yielding the biggest risk improvement for both approaches. Here, the algorithms find different solutions, however the differences in risk improvement are marginal (Tables 1 and 2). Most importantly, both approaches identify branches 2 and 6 as strategic for TEP grid configuration. This emphasizes the robustness and consistency of the identified strategic branches across various scenarios and methods.

Appendix C offers a detailed overview of the NSGA-III experimental setup. The results show that NSGA-III produces a greater number of solutions in approximately seven times fewer iterations.[3], although our innovative method excels in discovering more secure grid configurations at a reduced cost. Additionally, unlike the NSGA-III framework, where no enhancements are observed with an increased number of iterations, our framework exhibits continuous learning throughout episodes.

4.4. Online use of the trained agent

The lower computational efficiency of RL compared to NSGA-III is compensated by several factors. First, RL benefits from continuous learning. Second, after training, the RL agent can serve as an expert evaluator with little to no retraining required, even if the grid configuration changes. For demonstration, here we present two computational experiments, namely, applying the trained agent (i.e., using the θ parameters learned at the end of training for Q_θ) to the original IEEE 118-bus system and to the modified IEEE 118-bus system with the addition of two extra branches.

[3] The comparison based on computational resources is not feasible because the RL algorithm primarily utilizes GPUs, whereas the NSGA-III algorithm runs on CPUs. This fundamental difference in hardware usage affects the efficiency and speed of each method, making direct comparisons based on computational time or resources unfair. In fact, we can compare the number of total iterations, which means steps into the environment for RL or objective function evaluations for NSGA-III.

Table 3
Setup for testing the trained agent on a variation of the IEEE 118 with two extra random branches. All the experiments converge to the same line upgrade, comprising line 120, which results in RI = 1.649, c = 2.79 M USD.

ϵ	Total No. iterations	No. Pareto optimal solutions
0.2	827	4
0.5	1148	7
1	1708	7

Upon applying the agent to the original IEEE 118-bus system, we observe its ability to identify two critical solutions, i.e. (7.286 M USD, −1.6163) and (5.49 M USD, −1.1899). The associated configurations correspond to the one depicted in Fig. 7(II) and the grid featuring the upgraded line 6. Notably, the agent converges to the optimal compromise solution. This demonstrates that the identified policy (which maps the best action from the initial state) can be just applied to optimization problems, decreasing the computational effort to find the best compromise solution. In real-world scenarios, however, the grid configuration changes due to planned or unplanned outages, operational switching, grid expansions, or the decommissioning of aging infrastructure. To test if the trained agent can function as an expert evaluator for various power grid configurations with minimal training, we introduce two random lines to the IEEE 118-bus system. The trained agent is applied to it for 500 episodes with different initial exploration rates (see Algorithm 1). Regardless of the initial exploration rate and number of iterations, the agent consistently converges to the same upgraded grid configuration. Specifically, the agent consistently selects line 120 in the TEP, which yields a risk improvement of −1.649 at a minimal cost of 2.79 M USD. The TEP problem is a black-box, multi-objective, non-convex optimization problem, meaning no paradigm guarantees finding the best global compromise solution. However, the agent's consistent identification of the same configuration, despite varying exploration criteria and with minimal retraining, underscores its robustness and computational efficiency. This contrasts with meta-heuristic algorithms, which require a completely new computationally expensive search. While we focused on grid topology changes, generalization to varying loading conditions beyond the 18 representative hours used in this study remains an area for future exploration.

To further evaluate the computational efficiency of the trained agent, Table 3 presents the number of iterations required to complete 500 episodes, along with the number of Pareto optimal solutions identified for different initial ϵ values. The number of iterations is a key metric for assessing computational expense, as it directly reflects the number of environment evaluations (or interactions), which is critical for determining the practicality of deploying the agent in real-time scenarios. Furthermore, it provides a measure of computational cost independent of device-specific factors. Employing the trained agent with a small initial ϵ, i.e. $\epsilon = 0.2$ significantly reduces the number of iterations required. Nevertheless, the agent finds more Pareto optimal solutions when the initial ϵ is higher, such as 0.5. Yet, transitioning to fully random initial exploration does not significantly increase the number of Pareto solutions found and erases any memory of previous training. In summary, the goal-oriented graph generation framework, coupled with the trained agent, facilitates rapid convergence to optimal solutions and offers robust adaptability to new grid configurations with minimal exploration and training overhead.

5. Conclusions

This paper introduces a goal-oriented graph generation method to address the risk-informed transmission expansion planning problem. The goal is to enhance system security by concurrently minimizing cascading failure propagation and grid expansion costs. The method utilizes reinforcement learning and a GNN-based cascading failure meta-model. The former trains an agent to learn a multi-objective

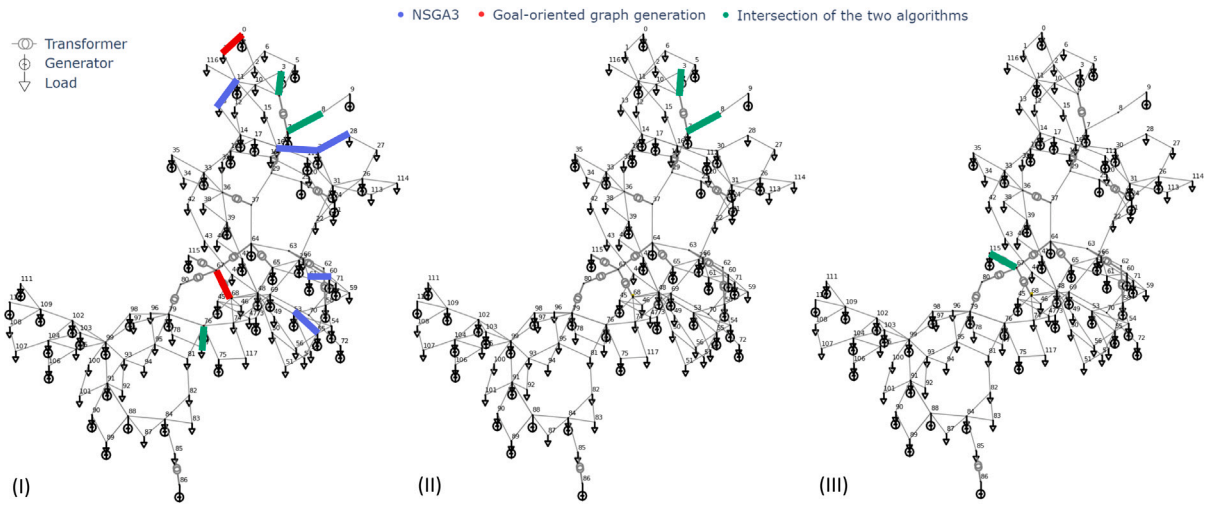


Fig. 7. Visualization of TEP for selected solutions: most secure solution I, best compromise solution II and cheapest solution III, as labeled in Fig. C.9.

expansion plan of the IEEE 118-bus system. The latter computes the risk of cascading failures in near real-time and, therefore, improves the computational efficiency of the method. We demonstrate the generalizability of the GNN meta-model on previously unseen topologies. In addition, we perform a comparative analysis against the state-of-the-art multi-objective optimization approach, i.e., the NSGA-III. We found that the reinforcement learning agent identifies more secure power grid configurations at a lower cost. Nevertheless, the obtained decisions are comparable, underscoring the significance of two specific lines within the best compromise solution shared by both approaches. Our results show that the trained agent demonstrates two key points. First, the trained agent autonomously converges to the optimal best compromise solution given cost and security objectives. Secondly, the agent demonstrates adaptability to novel grid configurations, requiring minimal exploration and training to identify optimal grid upgrades. In future work, we aim to assess how the trained GNQN can be utilized to identify new branches in non-existing corridors through link prediction modeling, a task well-suited for graph neural networks (GNNs). Link prediction in GNNs involves predicting the likelihood of an edge (branch) between two nodes (buses) in a graph, leveraging structural, feature-based, and topological information. Specifically, our objective is to extend the training to a fully connected graph that distinguishes between existing and potential new corridors. This approach would enable the model to identify plausible locations for new branches.

An evident challenge in this framework is assigning costs to lines in non-existing corridors, as we lack information regarding their geographical length, environmental constraints, and overall feasibility. Overcoming this limitation may require incorporating external geographical or planning data into the model.

Additionally, we plan to analyze how the meta-model performs when applied to new grid configurations. Previously, the meta-model demonstrated generalization capabilities to new topologies with additional branches on existing corridors. However, it remains uncertain how well the model will generalize when integrating entirely new corridors into the grid. This direction aims to improve the model's ability to predict and evaluate the integration of new branches into the grid.

CRediT authorship contribution statement

Anna Varbella: Writing – original draft, Visualization, Validation, Supervision, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Blazhe Gjorgiev:** Writing – original draft, Validation, Supervision, Methodology, Investigation, Formal analysis, Conceptualization. **Federico Sartore:** Writing – original draft, Software, Methodology, Investigation, Formal analysis. **Enrico Zio:** Writing – review & editing, Supervision, Investigation, Formal analysis, Conceptualization. **Giovanni Sansavini:** Writing – review & editing, Supervision, Resources, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

A.V. would like to express her heartfelt gratitude to the LASAR group at Politecnico di Milano for their invaluable support and for providing an inspiring environment that enabled the collaborative development of this research. A special thanks to Alfredo Oneto for his insightful guidance on the methodology underpinning the Welfare-Q procedure. His expertise and direction were instrumental in shaping the approach, and the authors deeply appreciate his contributions to this work.

Appendix A. State space specification

The state space S represents power grids as graphs. The power grid is denoted by $G = (V, E, \mathbf{X}, \mathbf{A})$. The node feature matrix $\mathbf{X}$ contains bus information, where $|V|$ is the number of buses, and f is the number of features per bus. In our case, there are 54 features per bus. These features are obtained by solving the optimal power flow for 18 randomly sampled hours from the yearly loading demand. The features include net active power, net apparent power, and voltage magnitude per bus. These features are concatenated to form a 118×54 node feature matrix, and mean normalization is applied. In PyTorch Geometric, the weighted adjacency matrix is represented by encoding the graph connectivity in Coordinate list (COO) format (Fey and Lenssen, 2019). An edge weight vector is assigned, and its elements are integer values representing the number of branches per corridor. The edge weight vector, with dimensions 179×1, is the only entity modified during the learning process. Importantly, the node features matrix remains unchanged.

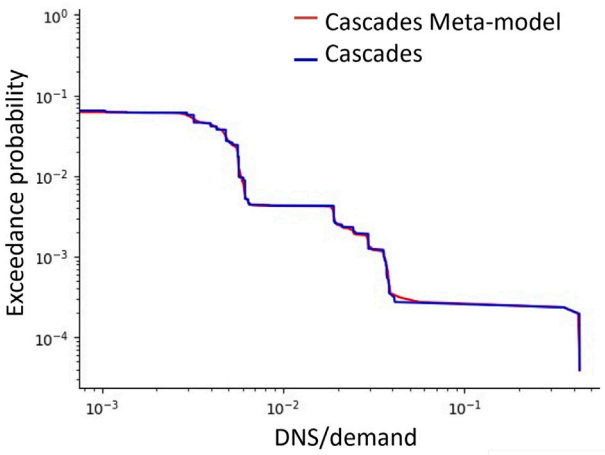


Fig. B.8. CCDF comparison: physics-based model (blue) vs trained GNN model (red).

Appendix B. Cascades GNN meta-model architecture and model performance

The meta-model of Cascades has been developed using geometric deep-learning techniques that involve training GNN. We extend the work of a prior study (Varbella et al., 2023) by presenting a regression model that leverages a Graph Neural Network (GNN) architecture for graph-level tasks. This model aims to predict the precise magnitude of Demand Not Served (DNS) resulting from a cascading failure event. It represents an advancement beyond the previous work, where the output of cascading failure models was encoded with a binary label, only indicating the presence or absence of DNS. The data is organized as graph-structured data, which is needed as input to the GNN, meaning that the dataset comprises a set of N graphs $\mathcal{G} = \{G_1, G_2, \dots, G_N\}$. Each graph represents the power grid state associated with a DNS in MW. Each input graph is assigned a node feature matrix ($\mathbf{N}$) and edge feature matrix ($\mathbf{E}$). Nodes and edges represent the buses and branches of the power grid, respectively. We provide three features per bus and four features per edge. The bus-level features are the net active power, net apparent power, and voltage magnitude. The branch-level features are the active power flow, reactive power flow, branch reactance, and branch rating. The initial contingencies are considered by removing the failed branches from the list of branches in the power grid and the respective branch features. All bus and branch features are normalized using mean normalization. In summary, each graph instance consists of the node feature matrix, the edge feature matrix, the branch list, and a graph-level label. The model uses two message-passing layers, specifically GraphTransformer layers (Shi et al., 2020), each combined with a parametric Relu (PRelu) activation, one graph pooling layer, and three fully connected layers (FC) with PRelu activation. Finally, mean squared error loss is applied to deal with the regression problem. We report the results of the GNN model trained on a dataset of 56 IEEE 118 topologies. Each of the topologies is created by randomly imposing grid upgrades. We included 4806 graphs, as many as the scenario considered, the same scenario described in Section 4. As a result, the entire dataset to train and test the metamodel consisted of 269136 graphs. We split the dataset into 80-15-5 train–test–validation sets. We trained the meta-model with a Windows desktop with an AMD Ryzen Threadripper 3960X 24-Core Processor and an NVIDIA RTX A6000 GPU with 48 GB VRAM.. Fig. B.8 shows the risk curve obtained with the physics-based model (blue) and the risk curve obtained with the trained GNN model (red). We observe that the risk curves are very well aligned, i.e., the GNN model can reproduce the results of Cascades. The mean squared error (MSE) loss and the R2 score on the test dataset are 8.1e−8 and 99.8%, respectively. The R2 score is $R2 = 1 - \frac{RSS}{TSS}$, where RSS is the sum of squares of the residual errors and TSS is the total sum of the errors.

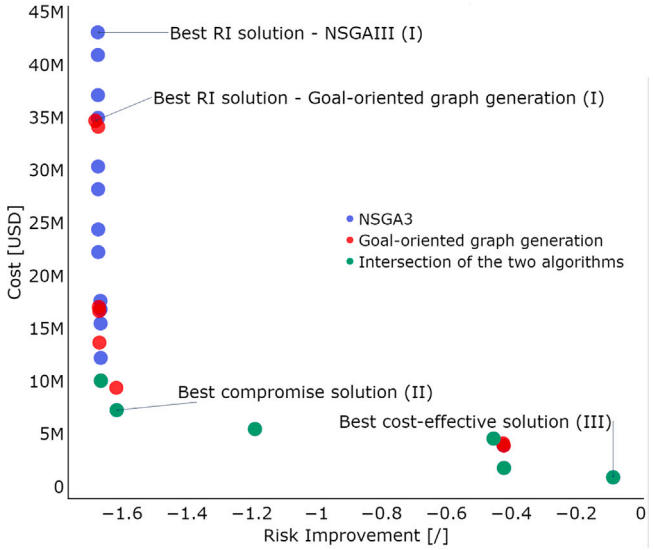


Fig. C.9. Goal-oriented graph generation vs NSGA-III.

Appendix C. NSGA-III framework

In this section, we comprehensively explore the results obtained from the NSGA-III algorithm (Deb and Jain, 2014), shown in Fig. C.9. NSGA-III is a Genetic Algorithm designed to solve optimization problems with multiple conflicting objectives, such as the TEP problem. The TEP is characterized by a nonlinear, non-convex nature and includes a black-box function associated with the second objective RI, which lacks an analytic form. The algorithm employs distinctive features to iteratively converge towards the desired Pareto-optimal front. In particular, it deploys widely distributed reference directions to efficiently guide the search through improved function evaluations, leading to a well-distributed Pareto front (Deb and Jain, 2014).

The experiment involves an initial population of 40 randomized individuals and 70 iterations. Each individual is generated by random boolean vectors with eight non-zero entries, representing the branches eligible for upgrade. If this results in all-zero entries, to avoid non-reproducibility of the population, a ”Repair” mechanism is implemented to randomly assign an entry equal to ”1” in the individual. This experimental setup provides a robust foundation for evaluating the algorithm’s performance. The number of iterations used with the NSGA-III is lower than those used by the goal-oriented graph generation framework. However, we monitor the hypervolume to examine the convergence of the NSGA-III. We find that in 30 iterations, the Pareto population converges to the reported solution, meaning that a higher number of iterations will likely not improve the results.

Fig. C.9 visualizes the NSGA-III outcomes, offering insights into its ability to generate diverse, high-quality solutions throughout the iterations. In Table 2, we present all the branches associated with the respective objective values. The algorithm is implemented using the PYMOO library (Blank and Deb, 2020). The experiments are conducted on a Windows machine with an Intel(R) Core(TM) i7-4770 4-Core Processor. To complete the iterations, approximately 8 h are necessary.

Appendix D. DQN versus welfare DQN

In this section, we present results obtained through the traditional Deep Q Learning framework (Mnih et al., 2013) instead of the Welfare Q-learning multi-objective framework outlined in Section 2.2. Unlike the non-stationary policy employed in the multi-objective framework, the traditional DQN algorithm relies on a stationary policy. We adopt a scalarization approach for the reward vector instead of the Nash

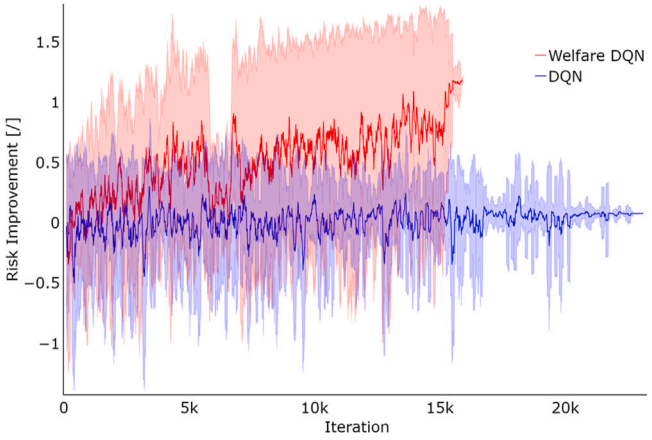


Fig. D.10. Development of the risk improvement metric in the agent training. We present the risk improvement metric as a positive number, whereas it was previously shown as a negative value, to better illustrate its growth throughout the training episodes.

Social Welfare function in Eq. (9). The same initial conditions, hyperparameters, and agent architecture are used for the two approaches, reported in Section 3. The comparison of learning processes between the Welfare Q Learning method and the DQN method is depicted in Fig. D.10, which shows the development of the metric RI in the learning process and demonstrates performance disparities between the two approaches. Through this comparative analysis, we observe that using a traditional DQN model, the agent is not learning strategies to maximize risk improvement despite using equal initial conditions. The traditional DQN algorithm relies on a stationary policy and simpler scalarization of the reward vector, which appear crucial to learning an optimal strategy. Furthermore, more iterations are needed for the traditional DQN approach because fewer Pareto optimal solutions are found, making the episodes longer, see Algorithm 2. The welfare DQN agent learns steadily but shows temporary instability around 5000 iterations. After this drop, the RI increases monotonically, stabilizing performance due to the decaying exploration rate. By the end of the training, the welfare DQN exhibits minimal reward and RI variance, indicating convergence. To maintain stability, we use experience replay and target network updates. These address common DRL challenges like correlated data and non-stationary targets. The decaying exploration rate ensures a balance between exploration and exploitation, aiding convergence. In future work, we plan to explore adaptive learning rate schedules and prioritized experience replay to further enhance stability and robustness.

Data availability

Data will be made available on request.

References

Akbari, T., Rahimikian, A., Kazemi, A., 2011. A multi-stage stochastic transmission expansion planning method. Energy Convers. Manage. 52 (8), 2844–2853. http://dx.doi.org/10.1016/j.enconman.2011.02.023, URL: https://www.sciencedirect.com/science/article/pii/S019689041100104X.

Alhamrouni, I., Khairuddin, A., Ferdavani, A.K., Salem, M., 2014. Transmission expansion planning using AC-based differential evolution algorithm. IET Gener. Transm. Distrib. 8 (10), 1637–1644. http://dx.doi.org/10.1049/iet-gtd.2014.0001, URL: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-gtd.2014.0001, arXiv:https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/iet-gtd.2014.0001.

Andersson, G., Donalek, P., Farmer, R., Hatziargyriou, N., Kamwa, I., Kundur, P., Martins, N., Paserba, J., Pourbeik, P., Sanchez-Gasca, J., Schulz, R., Stankovic, A., Taylor, C., Vittal, V., 2005. Causes of the 2003 major grid blackouts in North America Europe, and recommended means to improve system dynamic performance. IEEE Trans. Power Syst. 20 (4), 1922–1928. http://dx.doi.org/10.1109/TPWRS.2005.857942, Cited by: 995.

Auger, A., Bader, J., Brockhoff, D., Zitzler, E., 2009. Theory of the hypervolume indicator: optimal μ-distributions and the choice of the reference point. In: Proceedings of the Tenth ACM SIGEVO Workshop on Foundations of Genetic Algorithms. In: FOGA '09, Association for Computing Machinery, New York, NY, USA, ISBN: 9781605584140, pp. 87–102. http://dx.doi.org/10.1145/1527125.1527138.

Blank, J., Deb, K., 2020. Pymoo: Multi-objective optimization in python. IEEE Access 8, 89497–89509.

Cao, N.D., Kipf, T., 2018. MolGAN: An implicit generative model for small molecular graphs. arXiv preprint arXiv:1805.11973.

Carreras, B.A., Lynch, V.E., Dobson, I., Newman, D.E., 2002. Critical points and transitions in an electric power transmission model for cascading failure blackouts. Chaos 12, 985–994. http://dx.doi.org/10.1063/1.1505810.

Chen, D., Lin, Y., Li, W., Li, P., Zhou, J., Sun, X., 2020. Measuring and relieving the over-smoothing problem for graph neural networks from the topological view. Proceedings of the AAAI Conference on Artificial Intelligence 34 (04), 3438–3445. http://dx.doi.org/10.1609/aaai.v34i04.5747.

Choi, J., Mount, T.D., Thomas, R.J., 2007. Transmission expansion planning using contingency criteria. IEEE Trans. Power Syst. 22 (4), 2249–2261. http://dx.doi.org/10.1109/TPWRS.2007.908478.

Deb, K., Jain, H., 2014. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints. IEEE Trans. Evol. Comput. 18 (4), 577–601. http://dx.doi.org/10.1109/TEVC.2013.2281535.

Deb, K., Pratap, A., Agarwal, S., Meyarivan, T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. 6 (2), 182–197. http://dx.doi.org/10.1109/4235.996017.

Debreu, G., 1954. Valuation equilibrium and pareto optimum. Proc. Natl. Acad. Sci. USA 40 (7), 588–592. http://dx.doi.org/10.1073/pnas.40.7.588.

Dodu, J., Merlin, A., 1981. Dynamic model for long-term expansion planning studies of power transmission systems: the Ortie model. Int. J. Electr. Power Energy Syst. 3 (1), 2–16. http://dx.doi.org/10.1016/0142-0615(81)90025-9, URL: https://www.sciencedirect.com/science/article/pii/0142061581900259.

Fan, Z., Peng, N., Tian, M., Fain, B., 2023. Welfare and fairness in multi-objective reinforcement learning. arXiv:2212.01382.

Fey, M., Lenssen, J.E., 2019. Fast graph representation learning with PyTorch Geometric. In: ICLR Workshop on Representation Learning on Graphs and Manifolds.

Gjorgiev, B., David, A.E., Sansavini, G., 2022a. Cascade-risk-informed transmission expansion planning of AC electric power systems. Electr. Power Syst. Res. 204, http://dx.doi.org/10.1016/j.epsr.2021.107685.

Gjorgiev, B., Stankovski, A., Sansavini, G., Li, B., David, A., 2022b. Cascades A platform for power system risk analyses and transmission expansion planning.

Gjorgiev, B., et al., 2019. Cascades platform. URL: https://ethz.ch/content/dam/thz/special-interest/mavt/energy-technology/rre-dam/documents/Research/Cascades%20Platform_draft1.pdf.

Godwin, J., Schaarschmidt, M., Gaunt, A.L., Sanchez-Gonzalez, A., Rubanova, Y., Veličković, P., Kirkpatrick, J., Battaglia, P., 2022. Simple GNN regularisation for 3D molecular property prediction and beyond. In: International Conference on Learning Representations.

Golestani, S., Tadayon, M., Pirbazari, A., 2010. Transmission network expansion planning considering unit commitment problem simultaneously. IEEE PES Innov. Smart Grid Technol. Conf. Eur. ISGT Eur. http://dx.doi.org/10.1109/ISGTEUROPE.2010.5638905.

Gomes, P., Saraiva, J., 2019. State-of-the-art of transmission expansion planning: a survey from restructuring to renewable and distributed electricity markets. Electr. Power Syst. Res. 111, 411–424.

Guo, Z., Wang, H., 2021. A deep graph neural network-based mechanism for social recommendations. IEEE Trans. Ind. Inform. 17 (4), 2776–2783. http://dx.doi.org/10.1109/TII.2020.2986316.

Gupta, S., Kambli, R., Wagh, S., Kazi, F., 2015. Support-vector-machine-based proactive cascade prediction in smart grid using probabilistic framework. IEEE Trans. Ind. Electron. 62, 2478–2486. http://dx.doi.org/10.1109/TIE.2014.2361493.

Hamilton, W.L., 2020. Graph Representation Learning, vol. 14.

IEA, 2023. Credible Pathways to 1.5°C. IEA, Paris, https://www.iea.org/reports/credible-pathways-to-150c, Licence: CC BY 4.0.

Kile, H., Uhlen, K., Warland, L., Kjølle, G., 2014. A comparison of AC and DC power flow models for contingency and reliability analysis. In: 2014 Power Systems Computation Conference. pp. 1–7. http://dx.doi.org/10.1109/PSCC.2014.7038459.

Koren, Y., Bell, R., Volinsky, C., 2009. Matrix factorization techniques for recommender systems. Computer 42 (8), 30–37. http://dx.doi.org/10.1109/MC.2009.263.

Li, B., Gjorgiev, B., Sansavini, G., 2018. Meta-heuristic approach for validation and calibration of cascading failure analysis. In: 2018 IEEE International Conference on Probabilistic Methods Applied to Power Systems. PMAPS, pp. 1–6. http://dx.doi.org/10.1109/PMAPS.2018.8440477.

Lumbreras, S., Ramos, A., 2016. The new challenges to transmission expansion planning. Survey of recent practice and literature review. Electr. Power Syst. Res. 134, 19–29. http://dx.doi.org/10.1016/j.epsr.2015.10.013, URL: https://www.sciencedirect.com/science/article/pii/S0378779615003090.

Massaoudi, M., Abu-Rub, H., Refaat, S.S., Chihi, I., Oueslati, F.S., 2021. Deep learning in smart grid technology: A review of recent advancements and future prospects. IEEE Access 9, 54558–54578. http://dx.doi.org/10.1109/ACCESS.2021.3071269.

Mehta, I., Taghipour, S., Saeedi, S., 2022. Pareto frontier approximation network (PA-Net) to solve bi-objective TSP. arXiv:2203.01298.

Mercado, R., Rastemo, T., Lindelöf, E., Klambauer, G., Engkvist, O., Chen, H., Bjerrum, E.J., 2021. Graph networks for molecular design. Mach. Learn.: Sci. Technol. 2, http://dx.doi.org/10.1088/2632-2153/abcf91.

MingKui, W., ShaoRong, C., Quan, Z., Xu, Z., Hong, Z., YuHong, W., 2020. Multi-objective transmission network expansion planning based on reinforcement learning. In: 2020 IEEE Sustainable Power and Energy Conference. ISPEC, pp. 2348–2353. http://dx.doi.org/10.1109/iSPEC50848.2020.9350990.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M., 2013. Playing atari with deep reinforcement learning. arXiv:1312.5602.

Mossalam, H., Assael, Y.M., Roijers, D.M., Whiteson, S., 2016. Multi-objective deep reinforcement learning. arXiv:1610.02707.

Nedic, D.P., Dobson, I., Kirschen, D.S., Carreras, B.A., Lynch, V.E., 2006. Criticality in a cascading failure blackout model. Int. J. Electr. Power Energy Syst. 28, 627–633. http://dx.doi.org/10.1016/j.ijepes.2006.03.006.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. PyTorch: An imperative style, high-performance deep learning library. In: Advances in Neural Information Processing Systems 32. Curran Associates, Inc., pp. 8024–8035.

Raycheva, E., Gjorgiev, B., Hug, G., Sansavini, G., Schaffner, C., 2023. Risk-informed coordinated generation and transmission system expansion planning: A net-zero scenario of Switzerland in the European context. Energy 280, 128090. http://dx.doi.org/10.1016/j.energy.2023.128090.

Shi, Y., Huang, Z., Wang, W., Zhong, H., Feng, S., Sun, Y., 2020. Masked label prediction: Unified massage passing model for semi-supervised classification. CoRR abs/2009.03509 arXiv:2009.03509.

Standard TPL-001-4 NERC, 2015. Transmission System Planning Performance Requirements. North American Electric Reliability Corporation (NERC), Atlanta, Georgia, USA.

Stankovski, A., Gjorgiev, B., Locher, L., Sansavini, G., 2023. Power blackouts in europe: Analyses, key insights, and recommendations from empirical evidence. Joule 7, http://dx.doi.org/10.1016/j.joule.2023.09.005.

Subramonian, A., Sagun, L., Sun, Y., 2023. Networked inequality: Preferential attachment bias in graph neural network link prediction. arXiv:2309.17417.

Texas A&M University Engineering, 2022. IEEE 118-bus system. Available at: https://electricgrids.engr.tamu.edu/electric-grid-test-cases/ieee-118-bus-system/. (Accessed 23 December 2022).

Towers, M., Terry, J.K., Kwiatkowski, A., Balis, J.U., Cola, G.d., Deleu, T., Goulão, M., Kallinteris, A., KG, A., Krimmel, M., Perez-Vicente, R., Pierré, A., Schulhoff, S., Tai, J.J., Shen, A.T.J., Younis, O.G., 2023. Gymnasium. Zenodo, http://dx.doi.org/10.5281/zenodo.8127026, URL: https://zenodo.org/record/8127025.

U.S. Energy Information Administration, 2019. U.s. electric system operating data. URL: https://www.eia.gov/realtime_grid/#/data/graphs?end=20170108T00&start=20170101T00&regions=04 Accessed: (Accessed 6 August 2019).

Varbella, A., Gjorgiev, B., Sansavini, G., 2023. Geometric deep learning for online prediction of cascading failures in power grids. Reliab. Eng. Syst. Saf. 237, 109341. http://dx.doi.org/10.1016/j.ress.2023.109341, URL: https://www.sciencedirect.com/science/article/pii/S0951832023002557.

Vrakopoulou, M., Margellos, K., Lygeros, J., Andersson, G., 2013. A probabilistic framework for reserve scheduling and N-1 security assessment of systems with high wind power penetration. IEEE Trans. Power Syst. 28, 3885–3896. http://dx.doi.org/10.1109/TPWRS.2013.2272546.

Wang, Y., Chen, L., Zhou, H., Zhou, X., Zheng, Z., Zeng, Q., Jiang, L., Lu, L., 2021. Flexible transmission network expansion planning based on DQN algorithm. Energies http://dx.doi.org/10.3390/en14071944.

Watkins, C.J.C.H., 1989. Learning from Delayed Rewards (Ph.D. thesis). King's College, Oxford.

Watkins, C.J.C.H., Dayan, P., 1992. Q-learning. Mach. Learn. 8 (3–4), 279–292. http://dx.doi.org/10.1007/BF00992698.

Yin, L., Mo, N., Lu, Y., 2023. Lazy deep Q networks for unified rotor angle stability framework with unified time-scale of power systems with mass distributed energy storage. Eng. Appl. Artif. Intell. 126, 107129. http://dx.doi.org/10.1016/j.engappai.2023.107129, URL: https://www.sciencedirect.com/science/article/pii/S0952197623013131.

You, J., Liu, B., Ying, R., Pande, V., Leskovec, J., 2018. Graph convolutional policy network for goal-directed molecular graph generation. arXiv:1806.02473.

Yu, Z., Zhang, C., Wang, X., Chao, D., Liu, Y., Yu, Z., 2024. Dynamic graph topology generating mechanism: Framework for feature-level multimodal information fusion applied to lower-limb activity recognition. Eng. Appl. Artif. Intell. 137, 109172. http://dx.doi.org/10.1016/j.engappai.2024.109172, URL: https://www.sciencedirect.com/science/article/pii/S0952197624013307.

Zhang, H., Heydt, G.T., Vittal, V., Quintero, J., 2013. An improved network model for transmission expansion planning considering reactive power and network losses. IEEE Trans. Power Syst. 28 (3), 3471–3479.

Zhang, X.-M., Liang, L., Liu, L., Tang, M.-J., 2021. Graph neural networks and their current applications in bioinformatics. Front. Genet. 12, 690049. http://dx.doi.org/10.3389/fgene.2021.690049.

Zhang, F., Xuan, C., Lam, H.-K., 2024. An obstacle avoidance-specific reinforcement learning method based on fuzzy attention mechanism and heterogeneous graph neural networks. Eng. Appl. Artif. Intell. 130, 107764. http://dx.doi.org/10.1016/j.engappai.2023.107764, URL: https://www.sciencedirect.com/science/article/pii/S0952197623019486.

Zhu, Y., Du, Y., Wang, Y., Xu, Y., Zhang, J., Liu, Q., Wu, S., 2022. A survey on deep graph generation: Methods and applications. arXiv preprint arXiv:2203.06714.