
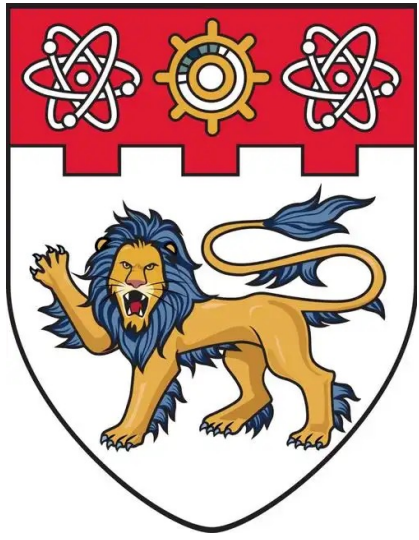


Paper No: 25PESGM0612



Large Language Model for Power System Scenario Generation Considering Weather Condition



Xiyuan Zhou^a, Yuheng Cheng^{b,*}, Yuan Zhong^c, Huan Zhao^{d,*}, Yifan Zhu^b, Xinlei Cai^e, Yan Xu^{a,*}, Junhua Zhao^{b,*}

^a School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore.
^b School of Science and Engineering, The Chinese University of Hong Kong (Shenzhen), Shenzhen, 518100, China.
^c College of Arts & Science, New York University, New York, NY 10003, United States.
^d Department of Building Environment and Energy Engineering, The Hong Kong Polytechnic University, Hong Kong, China.
^e Guangdong Power Grid Corporation, Guangzhou, 510600, China.

xiyuan002@e.ntu.edu.sg, yuhengcheng@link.cuhk.edu.cn, yz7628@nyu.edu, huan-paul.zhao@polyu.edu.hk, yifanzhu2@link.cuhk.edu.cn, 517665114@qq.com, xuyan@ntu.edu.sg, zhaojunhua@cuhk.edu.cn



Introduction

Background

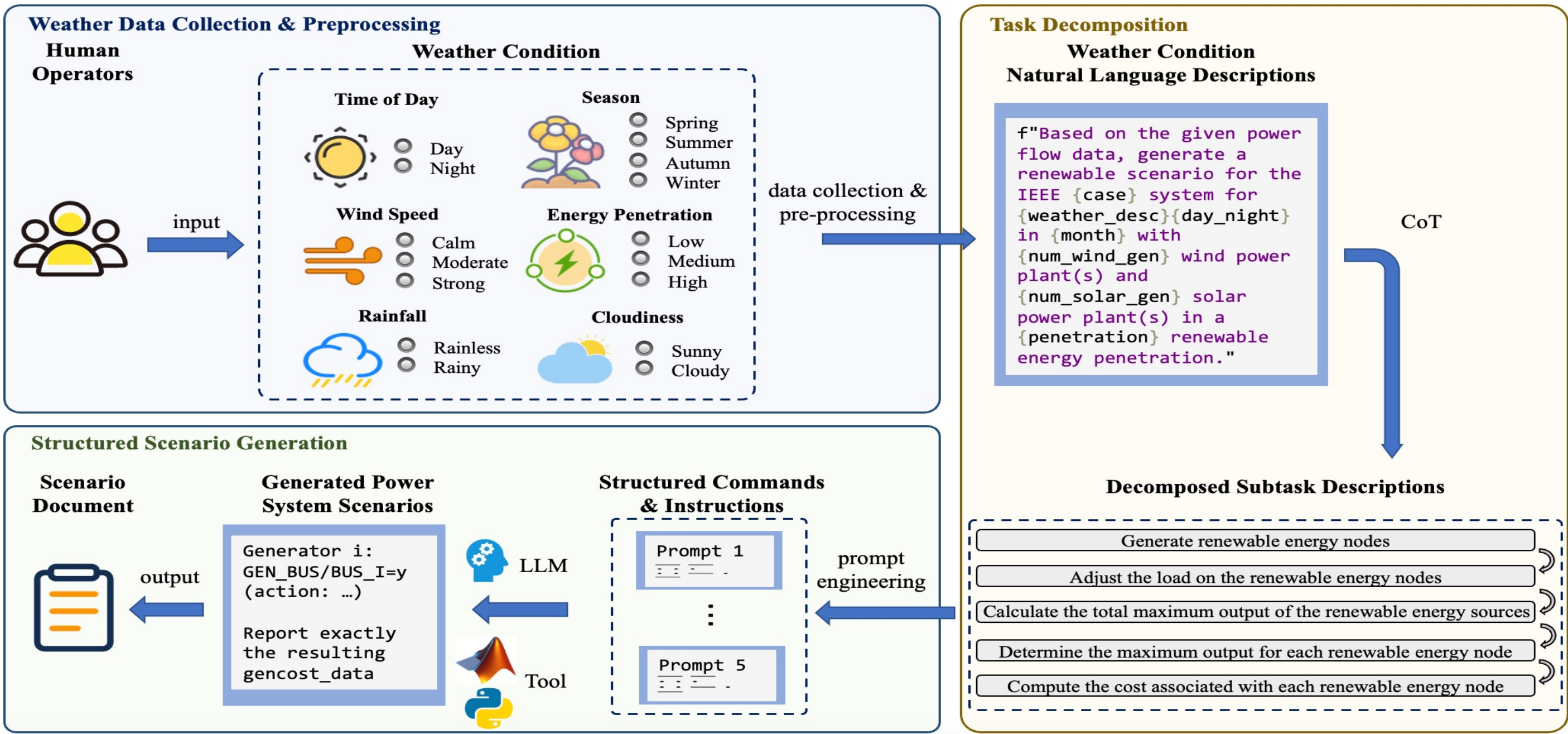
- High **renewable penetration** increases uncertainty in power systems.
- Uncertainty is strongly influenced by complex weather dynamics.
- Existing methods **lack flexibility** to model both stochasticity and weather impacts.
- LLMs** offer reasoning and context awareness for **scenario generation**.

Contributions

- This is the first application of LLMs to weather-aware, stochastic power scenario generation.
- Propose an **LLM-based** framework for weather-aware power system **scenario generation** with natural language interaction.
 - Integrate **CoT** reasoning and **prompt engineering** to enhance scenario quality.
 - Evaluate and compare multiple LLMs, demonstrating their potential for scenario generation.

Methodology

Overall Framework



A framework integrating an **LLM**, **CoT reasoning**, and **prompt engineering** is proposed to generate power system scenarios under weather conditions.

- Weather Input:** Weather conditions are standardized and described in natural language.
- Task Decomposition:** Scenario generation is broken into logical subtasks for step-by-step reasoning.
- Scenario Generation:** Structured prompts guide the LLM to produce feasible and cost-effective scenarios.

Chain-of-Thought Reasoning

- Decomposes complex scenario generation into logical subtasks.
- Enhances feasibility, diversity, and interpretability of outputs.
- Enables step-by-step reasoning aligned with domain constraints.

Prompt Engineering

- Translates weather and system parameters into structured model instructions.
- Ensures outputs are technically feasible and economically reasonable.
- Guides the LLM to generate controlled, high-quality scenarios.

Task 1: Generate renewable energy nodes
Based on the given power flow data, you are required to perform the following tasks to generate a renewable scenario for the IEEE 30 case system with {num_wind_gen} wind power plants and {num_solar_gen} solar power plants:
1. Randomly choose n target buses that you want to change to renewable buses (n = num of renewable power plants given).
• Format: Target Buses = a,b,...
2. Shuffle these buses in random order.
• Format: Target Buses (shuffled) = ...
3. Pair each Target Bus with a renewable power plant, and set the new bus type to 2 (1=PQ, 2=PV, 3=ref) in bus_data (except for Bus 1, which is the Reference Bus and type = 3). List each target bus:
• Format: Target Bus x: Wind/Solar Power Plant y, new Bus Type = z
4. Update the BUS_TYPE in bus_data to new Bus Type. Keep other parameters unchanged.
5. Report exactly the resulting modified bus_data in code block.

Task 2: Adjust the load on the renewable energy nodes
1. List Pd values of each target bus. If Pd=0, annotate that bus as no need to adjust Pd.
Format: Target Bus i: Pd = j (action: ...)
2. Calculate total Pd:
1) Get the Pd column from bus_data. Report the results. Format: Pd values={...},
2) Slice these Pd values into small groups and calculate the sum of each group cumulatively, showing formulae. Format: sum_p1=a+b+c...=s1,sum_p2=d+e+g...=s2...
3) Calculate the current total Pd. Format: Total Pd=x+y+...=n
3. Adjust the nonzero Pd of each target bus by adding some uncertainty (if number of nonzero Pds in target bus = 0, skip this step).
1) Calculate the noise factor range for total renewable pd
• min factor range = 0 (since pd is a non-negative value)
• max factor range = (0.2 * Total Pd) / Total renewable Pd + 1
2) For each nonzero Pd, list the target bus and multiply the pd with a random noise factor chosen from the factor range. Set Pd in bus_data to new Pd. Format: Target Bus x: noise factor = y, new Pd = Pd * y = z.
4. Report exactly the resulting modified bus_data in code block.

Task 3: Calculate the total maximum output of the renewable energy sources
1. List Pd values of each target bus. If Pd=0, annotate that bus as no need to adjust Pd.
Format: Target Bus i: Pd = j (action: ...)
2. Calculate Total Pmax cumulatively
Format: Total Pmax=1x2+...+y = x+...+y=n
3. Calculate total Pmax range (150% to 250% of total Pd)
Format: new total Pmax range = ... to ... = a to b
4. Pick an arbitrary number in that range as the desired total Pmax:
Format: desired new total Pmax = c
5. Infer the renewable energy penetration from the main prompt (low/medium/high, if not given, default penetration is high) and the corresponding total renewable Pmax percentage range: low -> Pmax range is 5% to 20% of the total, Pmax: medium -> Pmax range is 20% to 40% of the total, Pmax: high -> the Pmax range is 40% to 60% of the total, Pmax.
Format: Renewable energy penetration Percentage range
Total renewable Pmax range = ... to ... = a to b
6. Pick an arbitrary number in that range as the desired total renewable Pmax
Format: desired total renewable Pmax = c

Task 4: Determine the maximum output for each renewable energy
1. Infer the given time of the day, season and weather conditions from the main prompt.
Format: Day or night: ..., Season: ..., Weather: ...
2. List the GEN_BUS/BUS_I of current generators in gen_data. Format: Generator i: GEN_BUS = i (refer to Bus j - Target Bus/ordinary bus).
3. For each target bus, determine a new Pmax based on the generator type, the given time of the day, season and weather conditions, and annotate the action type. Ensure that the total renewable Pmax equals the given desired value.
Format: Target Bus i: typewind/solar, new Pmaxxy, action: no/referred gen -> add, has referred gen -> overwrite.
4. For each target bus, apply the action:
Add: add a new renewable generator for the target bus. Let the new GEN_BUS be the bus id.
Overwrite: change the generator to a renewable generator.
5. Report exactly the resulting gen_data in code block.

Task 5: Compute the cost associated with each renewable energy node
1. Decide generator cost data: gencost_data indexing rule: The nth row of gencost_data corresponds to the generator in the nth row of gen_data, for each generator in gen_data (report GEN_BUS/BUS_I), annotate if they were Unchanged, Added, Overwritten.
Format: Generator i: GEN_BUS/BUS_I=iy (action: ...) If a generator was overwritten, update the corresponding gen_data in gencost_data. If a renewable generator was added, add new data to gencost_data. In either case, parameter values for any renewable gen must be: MODEL and NCOST: consistent with the corresponding value of the other gen. all other parameter (STARTUP, SHUTDOWN, COST_2, COST_1, COST_0): set to 0.
2. Report exactly the resulting gencost_data in code block.

Test Scenario Construction

The IEEE **30-bus** system is used to construct scenarios under **228 weather combinations**. Six representative weather conditions are selected, and ten scenarios per condition are generated, considering **season, time of day, renewable generator type and quantity, and penetration level**.

Test Criteria

- Success Rate:** Measures the proportion of scenarios that are feasible under OPF constraints.
- Typicality:** Evaluates how representative and diverse a scenario is within and across clusters.
- Distribution Accuracy:** Assesses how closely the generated distributions match the actual solar and wind power distributions.

Testing Results

Comparison of Different Methods

TABLE I. COMPARISON OF DIFFERENT METHODS.

	GPT-3.5+CoT	LLaMA2 13b+CoT	LLaMA2 70b+CoT
Success Rate	65.75%	10.25%	38.33%
Typicality	0.5423	0.4570	0.4816
Distribution Accuracy	0.001223	0.001220	0.001221

- GPT-3.5+CoT** generates more **feasible and representative** scenarios than LLaMA2-based models, showing stronger reasoning and generalization in complex tasks.
- Model **architecture** and **pretraining quality** significantly impact scenario generation performance.

Comparison of Chain of Thought Method

TABLE II. PERFORMANCE COMPARISON USING CoT.

	GPT-3.5	GPT-3.5+CoT	LLaA2 13b	LLaMA2 13b+CoT
Success Rate	0	65.75%	0	10.25%
Typicality	0	0.5423	0	0.4570
Distribution Accuracy	0	0.001223	0	0.001220

- CoT is **essential** for task success: without CoT, both models fail completely.
- CoT **improves** feasibility, diversity, and interpretability by guiding LLMs through logical subtasks.
- CoT **enhances robustness** of LLMs under domain constraints.

Case Study Example

Response to “Compute the cost associated with each renewable energy node.”

```
# Decide generator cost data
Generator 1: GEN_BUS = 1 (action: Unchanged)
Generator 2: GEN_BUS = 2 (action: Unchanged)
Generator 22: GEN_BUS = 22 (action: Unchanged)
Generator 9: GEN_BUS = 9 (action: Added)
Generator 13: GEN_BUS = 13 (action: Added)
Since generators 9 and 13 were added as renewable generators, we need to update gencost_data for these generators while keeping MODEL and NCOST consistent with the corresponding values of the other generators. All other parameters (STARTUP, SHUTDOWN, COST_2, COST_1, COST_0) should be set to 0.
# Report exactly the resulting gencost_data
gencost_data = [[2, 0, 0, 3, 0.02, 2, 0], // Unchanged
[2, 0, 0, 3, 0.0175, 1.75, 0], // Unchanged
[2, 0, 0, 3, 0.0625, 1, 0], // Unchanged
[2, 0, 0, 3, 0.00834, 3.25, 0], // Unchanged
[2, 0, 0, 3, 0.025, 3, 0], // Unchanged
[2, 0, 0, 3, 0.025, 3, 0], // Unchanged
[2, 0, 0, 3, 0, 0, 0], // Overwritten
[2, 0, 0, 3, 0, 0, 0]] // Overwritten
The gencost_data has been updated to reflect the changes for the added renewable generators (Generator 9 and Generator 13).
```

- The framework generates realistic, cost-effective scenarios that support weather-aware power system planning.