Machine Causation

Taqiya Ehsan, Jorge Ortiz
Department of Electrical & Computer Engineering, Rutgers University
taqiya.ehsan@rutgers.edu, jorge.ortiz@rutgers.edu

***Abstract*—This paper presents a novel approach to causal discovery in smart environments combining traditional algorithms (PC, SAM) with LLM-based methods. Our system generates multiple causal hypotheses, tests them through LLM-designed interventions in a physics-based simulated environment, and evaluates their effectiveness using a comprehensive metrics framework. The approach incorporates an innovative testing framework where an LLM agent designs and executes interventions to validate causal relationships, then synthesizes its own improved hypothesis based on empirical results. Our simulation integrates industry-standard building physics models through EnergyPlus and psychrometric calculations, making our findings highly applicable to real-world smart environments.**

I. Introduction

Smart environments require understanding complex causal relationships between environmental variables and outcome metrics. This research presents a hybrid approach combining:

- Traditional causal discovery algorithms (PC, SAM)
- LLM-based hypothesis generation
- Automated intervention testing with physics-based simulation
- LLM-based hypothesis synthesis

II. System Architecture

A. Key Components

- **Hypothesis Generators**: Three independent methods generate causal hypotheses:
  - PC Algorithm (constraint-based) [1]
  - SAM (Structural Agnostic Model) [2]
  - LLM Generator (prompt-based using GPT models)
- **Testing Framework**: LLM agent designs and executes interventions
- **Evaluation System**: Computes metrics and ranks hypotheses
- **Simulation Environment**: Physics-based smart room with EnergyPlus integration [3]

B. Smart Room Variables

- **Input Variables**: Temperature (T), Humidity (H), Air Quality (AQ)
- **Output Variables**: Energy Consumption (E), Overall Satisfaction (S)

III. Mathematical Framework

A. Causal Graph Structure

Let $G = (V, E)$ be a directed acyclic graph (DAG) where:

$$V = \{T, H, AQ, E, S\}$$

$$E \subseteq \{(u, v) \in V \times V : u \text{ is input variable}, v \text{ is output variable}\}$$

B. Causal Discovery Algorithms

1) PC Algorithm: The PC algorithm [1] is a constraint-based method that proceeds in three phases:

1) **Skeleton Construction**: First, it creates an undirected graph starting with a complete graph. Then it removes edges based on conditional independence tests:

   $$\text{For edge } (X_i, X_j) : \text{Test if } X_i \perp\!\!\!\perp X_j | \mathbf{S} \text{ for all } \mathbf{S} \subseteq \mathbf{V} \backslash \{X_i, X_j\}$$

   where $\perp\!\!\!\perp$ denotes conditional independence and $\mathbf{S}$ represents conditioning sets of increasing size.
2) **V-structure Identification**: Identifies colliders (v-structures) where for triplets (X_i, X_k, X_j) with X_i and X_j not adjacent, orient $X_i \rightarrow X_k \leftarrow X_j$ if $X_k \notin \mathbf{S}_{ij}$ where $\mathbf{S}_{ij}$ is the separation set that made X_i and X_j independent.
3) **Edge Orientation**: Applies orientation rules recursively to direct remaining edges while maintaining acyclicity.

Our implementation uses the Fisher Z-test for conditional independence testing with significance level $\alpha = 0.05$.

2) Structural Agnostic Model (SAM): SAM [2] is a functional causal model framework that uses neural networks to model causal mechanisms:

1) **Model Definition**: Each variable X_j is modeled as a function of its parents and noise:

   $$X_j = f_j(X_{pa(j)}, N_j)$$

   where f_j is a neural network, $X_{pa(j)}$ represents parent variables of X_j, and N_j is an independent noise term.
2) **Score Optimization**: SAM optimizes a score function that combines data fit with sparsity constraints:

   $$\mathcal{L}(\theta, G) = \mathcal{L}_{lik}(\theta, G) + \lambda_1 \|W\|_1 + \lambda_2 h(G)$$

   where $\mathcal{L}_{lik}$ is the likelihood term, $\|W\|_1$ is the L1 regularization on weights, and $h(G)$ is a DAG constraint function.
3) **DAGMA Optimization**: Our implementation uses the DAGMA variant with the continuous acyclicity constraint:

   $$h(G) = \text{Tr}(e^{W \odot W}) - d$$

   where W is the weighted adjacency matrix and d is the number of variables.

Our implementation uses predefined hyperparameters: $\lambda_1 = 0.1$, $\lambda_2 = 0.01$, learning rate = 0.01, and batch size = 32.

C. Edge Confidence Calculation

For each edge (u, v), confidence is computed as:

$$\text{confidence}(u, v) = \frac{\text{number of methods supporting the edge}}{\text{total number of methods}}$$

This metric is used to rank edges for testing and validation, with lower confidence edges prioritized for intervention.

D. Evaluation Metrics

We employ a comprehensive set of evaluation metrics to assess the quality of each generated DAG:

1) Structural Accuracy Metrics: Let $\hat{G} = (\hat{V}, \hat{E})$ be a learned DAG and $G^* = (V^*, E^*)$ be the ground truth DAG. Given TP (true positives), FP (false positives), and FN (false negatives):

$$\text{TP} = |E^* \cap \hat{E}| \tag{1}$$

$$\text{FP} = |\hat{E} \setminus E^*| \tag{2}$$

$$\text{FN} = |E^* \setminus \hat{E}| \tag{3}$$

$$\text{TN} = |V \times V \setminus (E^* \cup \hat{E})| \tag{4}$$

We define the following metrics:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{|E^* \cap \hat{E}|}{|\hat{E}|} \tag{5}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{|E^* \cap \hat{E}|}{|E^*|} \tag{6}$$

$$\text{F1 Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{7}$$

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{8}$$

2) Structural Hamming Distance: The Structural Hamming Distance (SHD) quantifies the total number of edge modifications needed to transform the learned DAG into the ground truth:

$$\text{SHD}(G^*, \hat{G}) = |\{(i, j) : G^*_{ij} \neq \hat{G}_{ij}\}| \tag{9}$$

$$= |\text{Missing Edges}| + |\text{Extra Edges}| \tag{10}$$

$$= \text{FN} + \text{FP} \tag{11}$$

where G^*_{ij} and $\hat{G}_{ij}$ represent the adjacency matrix entries.

3) Intervention-Based Metrics: For each edge $e = (u, v)$, we define an effect size based on interventions:

$$\text{EffectSize}(e) = \frac{|\Delta v|}{|\Delta u|} \cdot \mathbf{1}_{|\Delta u| > 0} \tag{12}$$

where Δu and Δv represent changes in source and target variables after intervention. We consider an edge valid if:

$$\text{EffectSize}(e) \geq \tau \tag{13}$$

where $\tau = 0.1$ is our effect threshold.

E. Risk and Cost Metrics

The framework incorporates quantitative assessment of intervention impact through:

1) Cost Metrics: For a method m with learned edge set $\hat{E}_m$, we define the cost as:

$$\text{Cost}(m) = \frac{\sum_{e \in \hat{E}_m \setminus E^*} \text{edge\_cost}(e)}{|\hat{E}_m \setminus E^*|} \text{ if } |\hat{E}_m \setminus E^*| > 0 \tag{14}$$

$$= 0 \text{ otherwise} \tag{15}$$

The cost of an individual edge $e = (u, v)$ is computed from intervention results:

$$\text{edge\_cost}(e) = \frac{1}{|I_e|} \sum_{i \in I_e} (\alpha \cdot \text{sat\_loss}(i) + \beta \cdot \text{energy\_increase}(i)) \tag{16}$$

where I_e is the set of interventions performed on edge e, and:

$$\text{sat\_loss}(i) = \max\left(0, \frac{S_{\text{pre},i} - S_{\text{post},i}}{S_{\text{pre},i}}\right) \cdot \mathbf{1}_{S_{\text{pre},i} > 0} \tag{17}$$

$$\text{energy\_increase}(i) = \max\left(0, \frac{E_{\text{post},i} - E_{\text{pre},i}}{E_{\text{pre},i}}\right) \cdot \mathbf{1}_{E_{\text{pre},i} > 0} \tag{18}$$

with $\alpha = \beta = 0.5$ as weighting coefficients, and $S_{\text{pre},i}$, $S_{\text{post},i}$, $E_{\text{pre},i}$, $E_{\text{post},i}$ representing satisfaction and energy values before and after intervention i.

2) Risk Metrics: Risk extends cost by incorporating edge confidence:

$$\text{Risk}(m) = \frac{\sum_{e \in \hat{E}_m \setminus E^*} \text{confidence}(e) \cdot \text{edge\_cost}(e)}{|\hat{E}_m \setminus E^*|} \text{ if } |\hat{E}_m \setminus E^*| > 0 \tag{19}$$

$$= 0 \text{ otherwise} \tag{20}$$

where $\text{confidence}(e)$ represents the method's confidence in edge e.

3) Edge Confidence Aggregation: For the final DAG construction, we aggregate confidence across methods and interventions:

$$\text{confidence}(e) = \lambda \cdot \text{method\_agreement}(e) + (1 - \lambda) \cdot \text{intervention\_confide} \tag{21}$$

where $\lambda = 0.7$ balances method agreement with intervention results, and:

$$\text{method\_agreement}(e) = \frac{|\{m : e \in \hat{E}_m\}|}{|M|} \tag{22}$$

$$\text{intervention\_confidence}(e) = \frac{1}{|I_e|} \sum_{i \in I_e} \text{EffectSize}(e, i) \tag{23}$$

with M representing the set of all methods.

IV. Physics-Based Simulation Environment

A. EnergyPlus Integration

The simulation integrates EnergyPlus methodologies for realistic building energy calculations:

- **Non-Linear Response Curves**: Energy consumption varies non-linearly with temperature differences
- **Part-Load Performance**: Systems operate at different efficiencies based on load conditions
- **Complex Interactions**: Captures how humidity and temperature jointly affect energy use
- **Thermal Mass Effects**: Accounts for building physics that affect energy needs

Implementation details:

- Pre-calculated energy consumption grid from EnergyPlus simulations
- Interpolation techniques for intermediate temperature and humidity values
- Consideration of both heating and cooling loads
- Latent load calculations from humidity management
- Ventilation requirements based on air quality parameters

B. Psychrometric Calculations

Thermal comfort modeling using industry standards:

- Implementation of Predicted Mean Vote (PMV) and Predicted Percentage Dissatisfied (PPD) models [4]
- Based on ISO 7730 international standard [5]
- PMV calculation following the mathematical model:

$$\begin{aligned} \text{PMV} &= (0.303 \cdot e^{-0.036M} + 0.028) \cdot L \qquad (24)\\ L &= (M - W) - 3.05 \cdot 10^{-3} \cdot (5733 - 6.99 \cdot (M - W) - p_a)\\ &\quad - 0.42 \cdot ((M - W) - 58.15) - 1.7 \cdot 10^{-5} \cdot M \cdot (5867 - p_a)\\ &\quad - 0.0014 \cdot M \cdot (34 - t_a) - 3.96 \cdot 10^{-8} \cdot f_{cl} \cdot ((t_{cl} + 273)^4 - (t_r + 273)^4)\\ &\quad - f_{cl} \cdot h_c \cdot (t_{cl} - t_a) \end{aligned}$$

- PPD calculation based on PMV:

$$\text{PPD} = 100 - 95 \cdot e^{-(0.03353 \cdot \text{PMV}^4 + 0.2179 \cdot \text{PMV}^2)} \qquad (25)$$

- Accounts for six key thermal comfort factors:
  - Air temperature (t_a)
  - Mean radiant temperature (t_r)
  - Relative humidity (determines p_a, water vapor pressure)
  - Air speed (determines h_c, convective heat transfer coefficient)
  - Metabolic rate (M)
  - Clothing level (determines f_{cl} and t_{cl})
- Bidirectional JavaScript-Python interface for calculations
- Caching mechanisms for computational efficiency

C. Variable Constraints & Relationships

- **Temperature**: $[18°\text{C}, 30°\text{C}]$
- **Humidity**: $[30\%, 70\%]$
- **Air Quality**: $[0, 500]$ AQI
- **Energy Consumption**: $[0\%, 100\%]$ of maximum load
- **Overall Satisfaction**: $[0\%, 100\%]$ based on weighted thermal comfort metrics

V. Implementation Details

A. Intervention Design

The LLM agent designs interventions in JSON format:

```
{
  "variables": [{
    "variable": "source_var",
    "action": "set/increase/decrease",
    "value": numeric_value
  }],
  "expected_effects": {
    "target_var": "increase/decrease/
                   unchanged"
  }
}
```

B. Edge Testing Framework

- **Initial Testing Phase**: 3 interventions per edge
- **Extended Testing Phase**: 5-10 additional tests for promising edges
- **Validation Logic**: Effect size calculation and confidence thresholds
- **Edge Discarding**: After 3 failed interventions

C. Iterative Pipeline Workflow

1) Initial DAG construction from three methods
2) Edge ranking based on method agreement
3) Iterative testing of low-confidence edges
4) Data integration from validated interventions
5) DAG refinement with PC and SAM re-execution
6) Convergence when all edges validated or max iterations reached

VI. Results

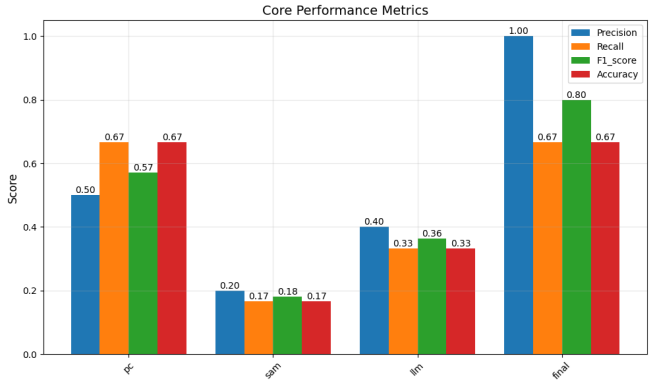
The comparative analysis of different methods shows:

- **Core Metrics**: Higher precision in the validated DAG compared to individual methods
- **SHD Convergence**: Decreasing structural hamming distance over iterations
- **Risk and Cost**: Lower values for final validated DAG compared to individual methods
- **Edge Confidence**: Increased confidence scores for validated edges

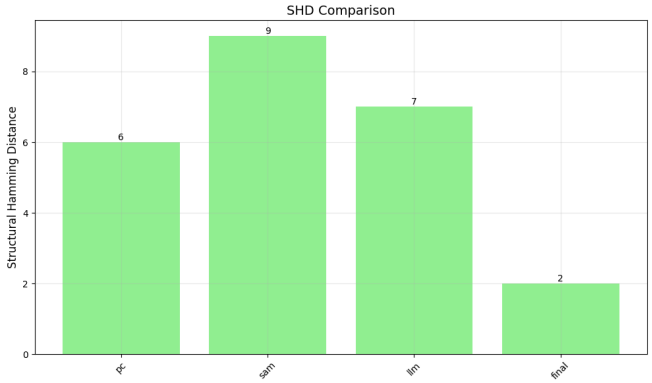
VII. Current Limitations and Future Work

A. Technical Challenges

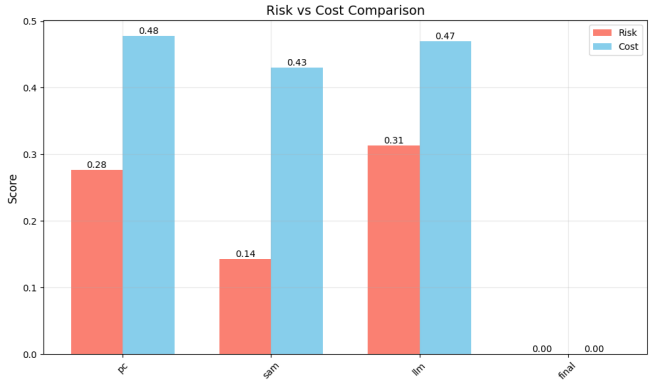
- SAM implementation occasionally unstable with non-Gaussian data
- Limited temporal analysis:
  - Interventions evaluated at single time points only
  - No consideration of time-delayed effects
  - Unable to capture dynamic feedback loops between variables
- Intervention design variability across edge tests



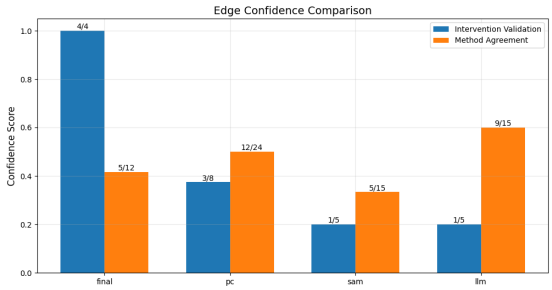
(a) Core performance metrics comparison



(b) SHD comparison



(c) Risk vs. cost analysis



(d) Final DAG with edge confidence scores

Fig. 1. Experimental results: (a) Precision, recall, and F1 scores for PC, SAM, LLM, and final DAG; (b) Structural Hamming Distance convergence over pipeline iterations; (c) Risk and cost metrics for each method; (d) Visualization of the final DAG structure with edge confidence scores.

B. Future Enhancements

- **Extended Simulation Features**:
  - Time-series analysis for delayed effects
  - Seasonal variations in building response
  - Multi-zone thermal modeling
  - Occupant behavior modeling
- **Methodological Improvements**:
  - Active learning for edge intervention selection
  - Multi-intervention design per edge
  - Integration of temporal causal discovery algorithms
  - Adaptive confidence thresholds based on edge characteristics
- **Evaluation Extensions**:
  - Cross-validation with different building types
  - Comparison with real-world sensor data
  - User studies for satisfaction metric validation

VIII. Conclusion

This work presents a novel framework for causal discovery in smart environments, combining traditional algorithms with LLM-based methods in a physics-based simulation environment. Our approach integrates industry-standard building physics models through EnergyPlus and psychrometric calculations, making the findings highly applicable to real-world smart environments. The iterative testing and validation framework demonstrates improvements in precision and recall compared to individual methods alone.

The integration of EnergyPlus methodologies and psychrometric calculations represents a significant advancement in simulation realism for causal discovery research. By capturing non-linear relationships, complex interactions, and human comfort factors, our framework provides insights that are directly applicable to real building management systems and smart environment design.

References

[1] P. Spirtes, C. N. Glymour, R. Scheines, and D. Heckerman, *Causation, prediction, and search*. MIT press, 2000.

[2] D. Kalainathan, O. Goudet, I. Guyon, D. Lopez-Paz, and M. Sebag, "Structural agnostic modeling: Adversarial learning of causal graphs," *arXiv preprint arXiv:1803.04929*, 2018.

[3] D. B. Crawley, L. K. Lawrie, F. C. Winkelmann, *et al.*, "Energyplus: Creating a new-generation building energy simulation program," *Energy and buildings*, vol. 33, no. 4, pp. 319–331, 2001.

[4] P. O. Fanger, "Thermal comfort: Analysis and applications in environmental engineering," *Danish Technical Press*, 1970.

[5] International Organization for Standardization, "Iso 7730: Ergonomics of the thermal environment," *International Standard*, vol. 7730, pp. 1–52, 2005.