Scalable and Versatile Identification for Hierarchical Structural Causal Models: A New Look at Project STAR

Janis Aiad, Aghiles Drali, Aymen El Ouadrhiri, Anass Ettahiri,
Yasser Oufqir, Simon Patry,
David Cortes, Marianne Clausel, Emilie Devijver

August 26, 2026

Abstract

The STAR (Student-Teacher Achievement Ratio) experiment (1985, Tennessee, USA) is a landmark hierarchical dataset designed to assess the impact of class size on student outcomes, with observations nested within classes. To encode class-level interventions in such hierarchical settings, we develop a complete, scalable, open-source pipeline for Hierarchical Structural Causal Models (HSCM) that bridges symbolic identification and practical estimation. Our approach integrates graph transformations, `pyAgrum`'s do-calculus for automatic identification of causal effects, adaptation of symbolic expression into closed-form HSCM formulas, and numerical estimation from fitted local probability models. A key innovation is our adapted Abstract Syntax Tree (AST), which decomposes `pyAgrum`'s identified formulas into independent density, expectation, and marginalization tasks, enabling parallel and scalable computation. We validate the pipeline on canonical HSCM motifs and benchmark scenarios with known ground truth, then apply it to STAR kindergarten mathematics outcomes. The results show that flat baselines (ignoring hierarchy) recover associations but fail to encode class-level interventions, and that symbolic identification alone is not enough for practical Hierarchical Structural Causal inference; scalable estimation and numerical stability checks are central parts of the scientific object. The code, including the HSCM implementation and STAR replication scripts, is available here.

1 Introduction

Project STAR is a landmark randomized study in education, conducted in Tennessee in 1985, initiated by Krueger (1999) and later analyzed for long-run impacts of class assignment (Krueger and Whitmore, 2001; Chetty et al., 2011). STAR assigned in a fully randomized way students within schools to one of three class types: small classes, regular classes, or regular classes with a teacher's aide. Outcomes included standardized test scores in mathematics and reading, measured at the student level. While the canonical econometric analysis relies on student-level regressions with school fixed effects, the treatment mechanism is inherently hierarchical: class type is shared by all students in the same class, and class composition (such as gender balance, socioeconomic status, or peer environment) can influence both outcomes and the interpretation of the treatment effect. Crucially, the intervention (class size) is assigned at the class level, yet its effects are measured at the student level. The intervention target is therefore naturally modelled as adistribution of sub-unit treatments within a unit—and not as a binary treatment assigned independently to each observation (here the students) in a flat dataset.

Classical causal inference provides a framework for identifying causal estimands from observed data (Pearl, 2009; Imbens and Rubin, 2015), available for example in the library `pyAgrum` (Gonzales et al., 2017). It relies on the ID algorithm or the adjustment formulae when all variables are observed, and allow to rewrite a causal effect from observational distribution. However, most models assume a flat set of random variables, which fails to capture the nested structure of many real-world datasets, and, in particular, for our primary case study, Project STAR. In Project STAR, a flat model cannot distinguish between a class-level intervention (e.g., reducing class size for all students in a class) and a student-level intervention (e.g., moving a single student to a smaller class). If the analyst aggregates students into class

means, within-class heterogeneity (e.g., peer effects, teacher-student interactions) is lost. If the analyst pools all students, unit-level latent variables (e.g., class composition) induce unobserved dependence and confounding.

Weinstein and Blei (2026) introduced the formal framework for Hierarchical Structural Causal Models (HSCM) to model hierarchidal data (e.g., sutdents nested in classes). They propose their graphical representation, identification via an extended version of do-calculus and estimation. Unlike flat models, HSCM explicitly encode the hierarchical structure, enabling analysts to model interventions at the unit-level, instead of restricting to individual-level interventions only (that is, in our running example class-level instead student-level). Once an HSCM estimand is identified, it is usually not a single regression parameter but a complex formula combining unit-level probability models, response curves, and averages over auxiliary variables, especially introduced to handle HSCM. Without an adapted structured representation, such a hierarchical estimation procedure becomes a monolithic formula that is hard to audit and hard to scale. Existing work enables analysts to model hierarchical interventions and propose estimators of the causal effect, but stops short of providing a complete pipeline from raw nested data to numerical estimates for general hierarchical causal graph. The notebook associated to Weinstein and Blei (2026) is illustrating the method on a particular example, and the `y0` library (y0 contributors, 2025) recently added symbolic HSCM support via a domain-specific language, enabling the exploration of alternative scenarios, paving the way to counterfactual analysis.

However, these tools leave critical operational steps unresolved: fitting local probability models, scaling computations, and validating results on real-world datasets like those provided by Project STAR. This paper addresses these challenges by operationalizing HSCMs. Starting from a hierarchical graph, our implementation collapses the graph, augments it with outcome-distribution nodes when needed, and marginalizes the variables that will not be involved in the causal estimand. It then calls `pyAgrum` (Gonzales et al., 2017) for symbolic do-calculus identification, adapts the returned expression into closed-form HSCM formulas, and evaluates the resulting functional using probability and response models fitted from the nested data. The novelty of our approach relies in the use of the Abstract Syntax Tree (AST) to record the formal estimand, drive numerical evaluation, and exposes which density factors were actually fitted. The use of an AST provides major benefits in terms of estimation scalability. By decomposing the identified formula into explicit computational units, its hierarchical representation enables each node—such as a conditional density, expectation, or summation—to be estimated independently. The overall formula is then evaluated by combining many small, interpretable estimators rather than relying on a single opaque global model.

We validate the pipeline on synthetic benchmarks and real-world data, focusing on STAR's kindergarten mathematics outcomes. Our results demonstrate that reliable hierarchical structural causal inference requires the interplay between hierarchical modeling and symbolic methods. In particular, ignoring the hierarchical structure (e.g., by using flat models) may recover associations but fails to capture unit-level interventions. Conversely, symbolic identification alone is insufficient; scalable estimation and numerical stability checks are necessary for reliable inference.

Our main contributions are:

- a full, scalable and operational pipeline for estimating causal effects in hierarchical data, available as open-source Python code at `AI-vidence/hierarchicalcausalmodels`;
- an illustration of the method in the analysis of STAR's data, illustrating how hierachical models outperform flat models.

The remainder of the paper is organized as follows. Section 2 introduces standard notions of SCMs and HSCM, Section 3 describes the symbolic-to-numeric method, Section 4 validates the implementation on controlled settings, Section 5 gives the Project STAR analysis, and Section 6 discusses limitations and future work. The code, including the HSCM implementation and STAR replication scripts, is available in this repository.[1]

[1] https://github.com/AI-vidence/hierarchicalcausalmodels

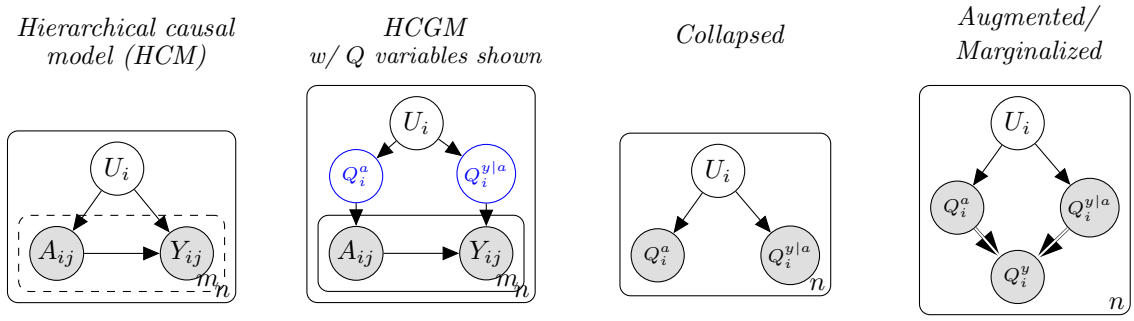

Figure 1: **An example of hierarchical causal model and its transformation.** (a) an HSCM with n units, each of which containing m subunits, (b) HCGM with explicit HCM's latent Q variables, (c) HCM's matching collapsed model and (d) its augmented/marginalized version. We apply do-calculus to this final model to perform identification in the original HCM.

2 Preliminaries: causal inference and hierarchical models

2.1 Structural Causal Models and interventions.

A *Structural Causal Model* (SCM, Pearl (2009)) is defined as $\mathcal{M} := \langle \boldsymbol{U}, \boldsymbol{V}, P(\boldsymbol{U}) \rangle$ where $\boldsymbol{U}$ is a set of exogenous variables taking values in $\mathcal{U}$, $\boldsymbol{V} = \{V_1, \ldots, V_d\}$ is a set of endogenous variables taking values in $\mathcal{V} = \mathcal{V}_1 \times \ldots \times \mathcal{V}_d$, $\mathcal{F} = \{f_1, \ldots, f_d\}$ is a set of functions such that for each $1 \leq j \leq d$, $f_j : (pa_j, u_j) \in (\mathcal{V}_{\mathrm{Pa}(V_j)} \times \mathcal{U}_j) \mapsto v_j \in \mathcal{V}_j$ and $P(\boldsymbol{U})$ is a probability distribution over mutually independent exogenous variables $\boldsymbol{U}$, with strictly positiveness on $\mathcal{U}$. $\mathrm{Pa}(V_j)$ is the set of parents of V_j and pa_j its realization. We assume that the SCM $\mathcal{M}$ induces a *directed acyclic graph* (DAG) $\mathcal{G} = (\boldsymbol{V}, \mathcal{E})$ with an edge from V_i to V_j whenever $V_i \in \mathrm{Pa}(V_j)$. The ancestor set $\mathrm{An}(V_j)$ contains all nodes with a directed path to V_j; $\mathrm{De}(V_j)$ denotes descendants.

In a given SCM $\mathcal{M} = \langle \mathcal{U}, \mathcal{V}, \mathcal{F}, P(\mathcal{U}) \rangle$, let $\mathbf{V} = (V_1, \ldots, V_n)$ be a subset of $\mathcal{V}$ and $\mathbf{v}$ be a valuation of $\mathcal{V}$. A hard intervention $\mathrm{do}(\mathbf{V} = \mathbf{v})$ replaces each structural function f_i for $V_i \in \mathbf{V}$ by the constant assignment $V_i := v_i$, which in the causal diagram $\mathcal{G}_{\mathcal{M}}$ corresponds to removing all incoming edges into each V_i. A soft intervention assigns possibly non-constant functions to variables without adding new causal connections.

2.2 Hierarchical Structural Causal models.

In an HSCM, endogenous variables are partitioned into *unit-level* (e.g., classes) and *subunit-level* (e.g., students) variables, with distributions defined at both levels. The idea of HSCM is to introduce a new concept of intervention at the unit level alongside the classical intervention at the individual level.

To explain it properly, let us illustrate this on an example depicted in Figure 2. In this running example, we are given n units (for e.g. classes), each containing m subunits (for e.g. m students in each class). The treatment variable for each individual j of the unit i is denoted A_{ij} (for e.g. tutoring hours for student j in class i) whereas the corresponding outcome is denoted Y_{ij} (for example mathematic grade of student j in class i at the end of the year). We assume that we have a common counfounder at the unit level, denoted U_i for each unit i, that may describe for example the size of the class. Classically, interventions on variables A_{ij} can be performed at the individual level. HCSM paves the way to a new kind of interventions, namely unit level interventions, by introducing auxiliary variables, the so called Q-variables. They allow to transform the Structural Causal Model into a Bayesian hierarchical model by adding a new level of hierarchy related to interventional distributions (respectively conditional distributions with respect to interventions). For e.g., in our running example, one introduces two auxiliary variables $Q^{(a)}$ and $Q^{(y|a)}$ which are respectively a distribution over interventional distributions on A and a distribution over conditional distributions of Y conditionally to A. The HSCM framework consists in turning the causal model into a generative model. We explicit it for the example in Figure 2:

- We draw i.i.d. counfouders U_i modelling the common covariates for each unit i (in our example for each class)

- Thereafter we generate the individual treatments $A_{i,j}$ in the following way:
  - Conditionally to U_i, we draw a distribution $Q_i^{(a)}$ on interventional distribution common for each unit depending on a parameter a. In the running example corresponding to the STAR project modelling, for each class, we draw a distribution on interventional distributions of the number of hours of tutoring for each student.
  - We draw from the distribution $Q_i^{(a)}$ the treatment variables for each individual j in unit i. In our running example, for each student we draw a distribution of the number of hours of tutoring for each student j in unit i.
- Now we generate the individual outcomes Y_{ij} in the following way:
  - Conditionally to U_i, we draw a distribution on conditional interventional distribution $Q_i^{(y|a)}$ common for each unit i
  - We draw from $Q_i^{(y|a)}$ the outcome Y_{ij} of each individual j of each unit i.

The Q-variables are then summarizing subunit-level mechanisms (e.g., treatment assignment, outcomes) at the unit level, enabling causal reasoning in hierarchical settings. This aggregation allows us to define causal targets that operate on distributions of subunit-level variables rather than individual observations. Our primary causal target is the contrast between expected unit-level outcomes under two different distributions of subunit-level treatments:

$$\tau(q_0 \to q_1) = \mathbb{E}[Y \mid \mathrm{do}(Q^{(a)} = q_1)] - \mathbb{E}[Y \mid \mathrm{do}(Q^{(a)} = q_0)], \tag{1}$$

which compares the expected unit-level outcome distribution under two different treatment distributions. Other target may be considered, and similar analysis can be derived, based on the identification of each term. We restrict here (without loss of generality) to the average treatment effect.

In general, to apply standard causal identification tools (e.g., do-calculus) to HSCMs, Weinstein and Blei (2026) propose a three-step graphical transformation that converts the hierarchical model into an equivalent flat model:

- *Step 1: collapsing.* Replace every sub-unit mechanism by its Q-node and lift all edges to a flat directed acyclic graph over unit-level variables and Q-variables.
- *Step 2: augmenting.* Augment the collapsed model with the auxiliary variable $Q^{(y)}$ modelling the distributions of outcomes.
- *Step 3: marginalizing.* Project out variables that are not part of the identification problem.

Then, standard do-calculus can be applied to determine whether a causal target like $\tau(q_0 \to q_1)$ is identifiable from the observed data. This approach leverages the power of graphical methods while accounting for the hierarchical structure of the data.

3 Causal effect estimation in Hierarchical Structural Causal Models

3.1 Overview of the approach

Our contribution is a complete and scalable pipeline that bridges the gap between hierarchical causal models (HSCMs) and practical estimation, by combining existing tools in a novel and principled way. While each component (graphical transformations, symbolic identification, numerical evaluation) exists in isolation, their integration into a single, automated workflow for hierarchical data is, to our knowledge, new. This pipeline enables analysts to move seamlessly from a hierarchical graph and nested data to a numerical estimate of causal contrasts like $\tau(q_0 \to q_1)$ introduced in Eq. 1. The pipeline consists of three stages, illustrated in Figure 2:

Step 1 **Modeling step**: From the HSCM, we construct a flat causal graph using the Q-variable trick, collapsing, augmenting, and marginalizing steps of Weinstein and Blei (2026). This step replaces subunit-level variables with Q-variables (e.g., class-level treatment distributions) and ensures the graph is compatible with standard do-calculus.

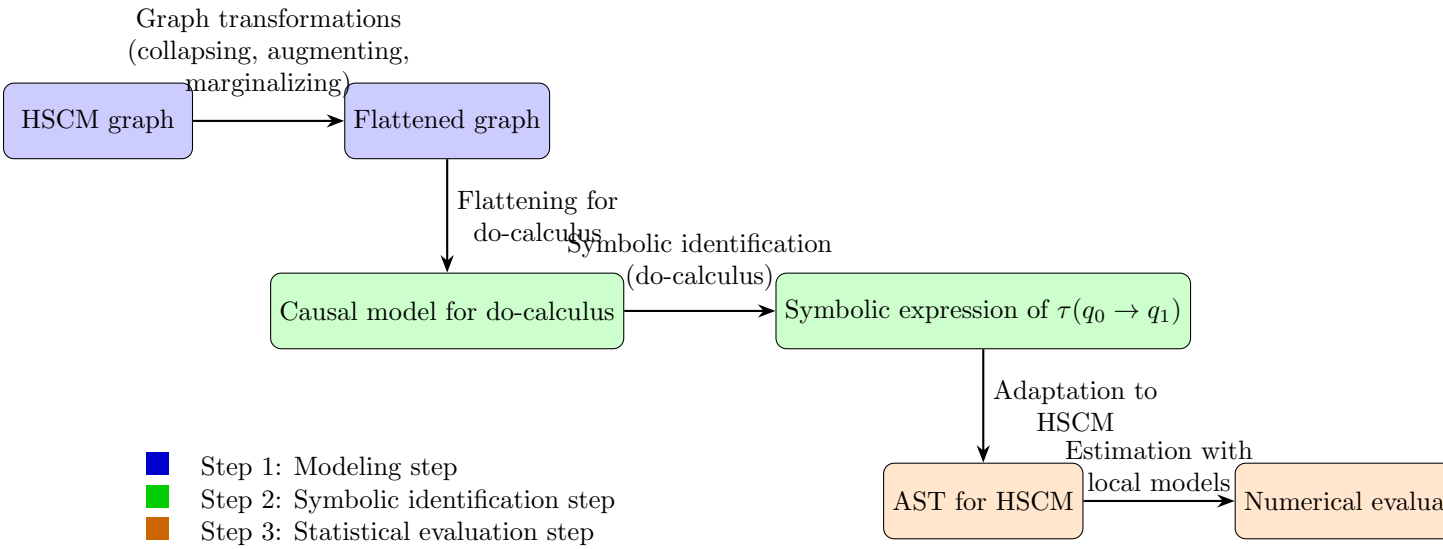

Figure 2: Pipeline for causal identification in Hierarchical Structural Causal Models (HSCMs).

Step 2 **Symbolic identification step**: The flattened graph is passed to `pyAgrum` (Gonzales et al., 2017), which performs do-calculus to derive a symbolic expression for the causal target (e.g., $\tau(q_0 \to q_1)$). This expression is a functional of observed distributions, but it is initially agnostic to the hierarchical structure. Our contribution here is to translate this output back into the HSCM framework mapping each term in the expression to a hierarchical mechanism

Step 3 **Statistical evaluation**: The translated expression is decomposed into a set of independent estimation tasks (e.g., estimation of conditional densities, expectations, or marginalizations), each of which can be fitted locally to the data. The modularity of this step, done by Abstract Syntax Tree (AST), allows for parallel estimation across units, factors, or Monte Carlo samples, making the pipeline scalable to large hierarchical datasets.

This separation between symbolic and numerical layers is key: the symbolic layer determines *what* needs to be estimated (based on the graph and intervention), while the numerical layer handles *how* to estimate each component (using local models).

Step 1, the modeling step, has been introduced in generality in Weinstein and Blei (2026) and summarized in our preliminaries (Section 2).

Step 2 consists in identifying a causal effect $\mathbb{E}(Y|\text{do}(Q^{(a)} = q))$, for a flatten graph: the routine implemented in `pyAgrum`, based on the ID algorithm, is diretly doing that. As we assume causal sufficiency, adjustment by the parent is one way for example to identify. Remark that `pyAgrum` never sees the original nested table: it sees the collapsed HSCM graph and returns a symbolic expression for $\mathbb{E}[Y \mid \text{do}(Q^a = q^a)]$ in terms of observed variables of that graph. Keeping this step separate from estimation is what allows the same identified expression to be evaluated later with Gaussian, mixture, categorical, non-parametric, or neural models for the required local probabilities and responses.

Now we describe the last step of our procedure, the numerical evaluation. Particularly, we describe in details the Abstract Syntax Tree (AST) structure in Step 3, which ensures that the pipeline is both **general** (applicable to any HSCM) and **efficient** (exploiting parallelism inherent in the hierarchical structure). Importantly, it also allows for diagnostics (e.g., checking the stability of the final contrast $\tau(q_0 \to q_1)$ to perturbations in local models). Then, we illustrate how to estimate the formula on two examples, but the pipeline can consider any estimation method.

3.2 Step 3: Statistical evaluation of the causal hierarchical effects

The general target is $\tau(q_0 \to q_1)$ given in Equation (1), seen as a difference between the two numerical evaluations, related to the interventions $Q^{(a)} = q_0$ and $Q^{(a)} = q_1$. We explain how to estimate one term: changing from $\text{do}(Q^a = 1)$ to $\text{do}(Q^a = 0)$ does not change the method, only the intervention value used.

3.2.1 Abstract Syntax Tree for estimation

The symbolic expression returned by `pyAgrum` is adapted into the Abstract Syntax Tree (AST) data type of `pyAgrum` and then written as the closed-form do-calculus formula used by the estimator. The AST data type is a nested object representation of the `pyAgrum`-identified formula after translation into HSCM density factors. Each node of the AST has a computational role. A summation node stores the variables to marginalize; a product node stores the multiplicative factors; a conditional node stores one density key of the form $P(\text{outcome} \mid \text{parents})$; and a leaf node is used for constants or fallback terms. The do-calculus engine demo prints this adapted object directly. The example below is not the STAR graph: it is a small synthetic collapsed-graph example using the same transformation logic as Figure 1, with two extra variables added only to make the AST traversal visible. It contains an outer marginalization over $Q^{z \mid a}$ and W, an inner marginalization over Q^a, and four fitted density factors:

Abstract Syntax Tree (AST)

```
sum on Qz_a,W for
| *
| | *
| | | P(W|Qa)
| | | P(Y|Qa,Qz_a,W)
| | sum on Qa for
| | | *
| | | | P(Qz_a|Qa,W)
| | | | joint P(Qa)
```

Identified formula

$$\sum_{Q^{z \mid a}, W} P(W \mid Q^a)\, P(Y \mid Q^a, Q^{z \mid a}, W) \times \left[\sum_{Q^a} P(Q^{z \mid a} \mid Q^a, W)\, P(Q^a) \right].$$

On the right, we give an causal estimand and on the left, is detailed the way it is encoded in the AST. The first line of the AST is the outer marginalization node over $Q^{z \mid a}$ and W. The product node of the AST then has two children: a direct product of fitted conditionals, and an inner summation over Q^a. From this tree, we need to have access to the density keys $P(W \mid Q^a)$, $P(Y \mid Q^a, Q^{z \mid a}, W)$, $P(Q^{z \mid a} \mid Q^a, W)$, and $P(Q^a)$, that must be fitted or read from data. In larger formulas, this same mechanism may produce dozens of density keys and many repeated unit-level prediction problems. Those tasks share the same formal estimand but do not need to be fitted as a single global regression, which is why the framework can scale to large nested datasets without changing the causal definition of the target.

3.2.2 Estimation procedure

The estimation step depends on the graph and the quantity to be estimated. The AST returns the conditional factors that must be estimated, and the numerical backend then selects an estimator according to the variable type and the available parent set. In the current implementation, discrete variables can be fitted with Bernoulli or categorical/multinomial estimators, continuous score variables with Gaussian or two-component Gaussian mixture estimators, and other supported families include Poisson, exponential, Gamma, Beta, lognormal, inverse-Gaussian, Student, Laplace, half-Cauchy, and non-parametric density estimators. We illustrate our estimation procedure over two classical causal patterns, but the same AST-to-estimator dispatch can be used for general HSCM formulas.

We stress the inherent scalability of our algorithm, which naturally derives from its modular implementation. Indeed, each conditional density factor can be estimated from its own parent set and then each $\widehat{\mu}_i$ can be fitted from the sub-units of unit i, before averaging.

Counfounder case We come back to the confounder case used as the running example, where a latent unit variable U_i affects both treatments (A_{ij}) and outcomes (Y_{ij}) of each individual j of the unit i. The collapsed graph contains Q_i^a and $Q_i^{y \mid a}$ as unit-level objects, and the intervention is evaluated by averaging the per-unit response: it reduces to the familiar Difference In Means estimator considered at the unit level

$$\widehat{\tau}(q_0 \to q_1) = \frac{1}{n} \sum_{i=1}^{n} \left[\widehat{\mu}_i(q_1) - \widehat{\mu}_i(q_0) \right], \qquad \widehat{\mu}_i(q) = \int y\, \widehat{Q}_i^{(y \mid a)}(y \mid q)\, \mathrm{d}y. \tag{2}$$

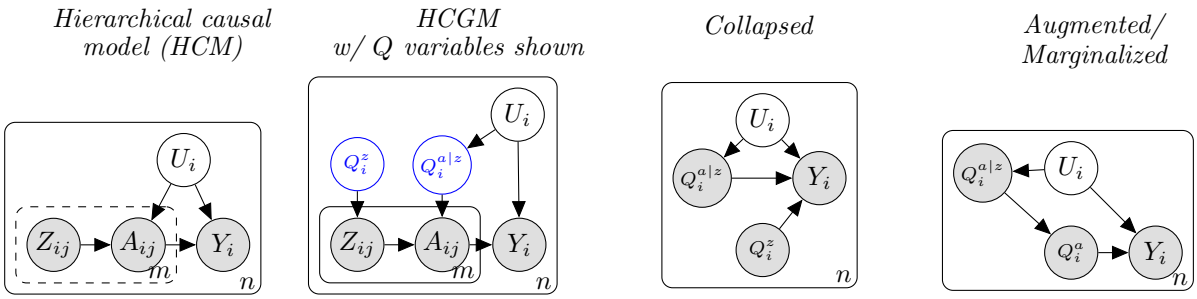

Figure 3: **An example of hierarchical causal model and its transformation.** (a) an HSCM with n units, each of which containing m subunits, (b) HCGM with explicit HCM's latent Q variables, (c) HCM's matching collapsed model and (d) its augmented/marginalized version. We apply do-calculus to this final model to perform identification in the original HCM.

To compute the term $\widehat{\mu}_i(q)$, we considered in our implementation two cases. For parametric families the integral is analytic or quadrature-based. In the non parametric setting, the backend draws Monte Carlo samples and averages the predicted outcomes.

Instrumental variable The instrumental variable case corresponds to the situation, where additionally to a common counfounder U_i at the unit level to the treatment variables A_{ij} and outcomes Y_{ij}, we add instrumental variables Z_{ij} at the individual level, which are parents of the A_{ij}'s. For the instrumental motif, we get the following expression, including a reweighting term analogous to inverse probability weighting:

$$\mathbb{E}\left[Y \mid \mathrm{do}(A \sim q_\star^{(a)})\right] = \mathbb{E}_{Q^{(a\,|\,z)},Q^{(z)}}\left[\frac{q_\star^a(a)}{Q^{(a\,|\,z)}((a\,|\,z))}\,Y\right], \tag{3}$$

provided the positivity conditions are met. In the backend, the denominator $Q^{(a\,|\,z)}(a\,|\,z)$ is a fitted density node. If this estimator is missing or has no support at the sampled value, the calculation becomes unstable. The ratio $q_\star^{(a)}(a)/Q^{(a\,|\,z)}(a\,|\,z)$ is the likelihood-ratio weight between the target intervention law and the fitted treatment mechanism, exactly the quantity that appears in importance sampling and in Horvitz–Thompson/inverse-probability estimators (Rubinstein and Kroese, 2016; Horvitz and Thompson, 1952; Robins et al., 2000). This is why the implementation reports both the symbolic formula and factor-level diagnostics.

Practical considerations In practice, an estimation formula may contain high-variance conditional density factors. This can happen if the outcome have several parents, leading to the introduction of several Q-variables related to each parent of the outcome. For example in the running example of the STAR project, we may consider that the math grade distribution Y depends jointly on treatment A, ethnicity E, gender G, and lunch status L.

The implementation records such cases and, in the normalized-factor runs studied below, rescales unstable multiplicative factors before evaluation. Operationally this is a controlled truncation of the raw product scale: the graph structure is retained, but the unstable factor is normalized so that it cannot overwhelm the rest of the functional. If the identified functional is $\mathfrak{f}$, the evaluated functional becomes a numerically stabilized object $\widetilde{\mathfrak{f}}$ with the same graph structure but normalized factor weights. This is not a harmless implementation detail: normalizing $P(Z\,|\,W)$ changes the weighting induced by that factor and therefore changes the numerical functional being evaluated. We therefore report both the nominal identified formula and the factor normalization actually used in computation.

3.3 Computational cost

The computational efficiency of our pipeline stems from its modular design, which separates symbolic identification from numerical evaluation. This allows us to exploit parallelism at multiple levels (units,

Table 1: Algorithmic cost and parallelization structure of the pipeline. The effective cost assumes ideal parallelization across P workers. We denote n is the number of units, m is the average number of sub-units per unit, d is the maximum parent count, v and e are the numbers of transformed graph nodes and edges, K is the number of density factors in the identified formula, B is the number of Monte Carlo samples or quadrature points.

Step	Sequential cost	Parallelization structure	Effective cost with P workers
Collapse/augmentation	$O(e+v)$ graph operations	graph-level (negligible)	$O(e+v)$
Build `pyAgrum` model	$O(e+v)$ (copy plus identification search)	symbolic backend	$O(e+v)$
AST translation	$O(K)$ expression traversal	factor-level traversal	$O(K)$
Per-unit Q fitting	$O(nmd)$ for simple parametric factors	independent accross units and factors	$\sim O(nmd/P)$ + scheduling overhead
Monte Carlo evaluation	$O(BK)$ per intervention	Independent across samples, units, and branches	$\sim O(BK/P)$ when branches are balanced

factors, and Monte Carlo samples), making the approach scalable to large hierarchical datasets. In Table 1, we summarize the computational cost of each step.

The global complexity of the pipeline is dominated by the numerical evaluation step, and scales as $O\left((nmd + BK)/P\right)$ under ideal parallelization. This reflects the fact that the symbolic steps (graph transformation and identification) are typically negligible compared to the numerical tasks, which involve fitting local models and Monte Carlo evaluation.

4 Experimental results over synthetic benchmarks

The experimental validation is done with three canonical motifs: confounding, confounding with interference/frontdoor structure, and an instrumental-variable design. These motifs are validation cases rather than restrictions of the pipeline: their ground truth is known, so they check that the graph transformation, identification, AST evaluation, and numerical estimators agree on controlled hierarchical structures.

The convergence diagnostic of the estimator in Figure 4 reports the confounder motif; the CUDA benchmark in Figure 5 uses the same three motifs at larger synthetic sizes to isolate batching speed. For a GPU implementation, the relevant parallelism is not a number of Python workers but a feasible tensor batch size, which depends on both n and m. We therefore treat GPU throughput as a hardware-specific benchmark rather than as a fixed worker-count comparison. The synthetic motifs are generated from known HSCMs, so they are used as controlled checks of the causal contrast rather than as real-data results.

The confounder experiment, in Figure 4, also varies the number of units. With $m = 50$ sub-units per unit, the estimate converges toward the true ATE as n increases from 10 to 200, while pooled OLS remains biased because it mixes unit-level latent heterogeneity with the treatment effect. This behaviour is important for interpretation: the HSCM estimator is recovering the unit-level intervention encoded by the graph, rather than only fitting an association in the pooled student table. It is also important computationally: for these motifs, the expensive work consists of repeated unit-level response evaluations, factor evaluations, and Monte Carlo or quadrature branches. These tasks are independent conditional on the identified AST, so the sequential work T_1 can ideally be reduced to $T_P \simeq T_1/P$ with P CPU workers, up to scheduling overhead. The same structure is highly GPU-parallelizable: units, sub-units, intervention values, and Monte Carlo samples are stacked into tensor batches and evaluated simultaneously when the density estimators are implemented in a tensor backend such as PyTorch or JAX.

On log–log axes, the fitted slope for speedup is approximately -0.32 on average across motifs. This slope summarizes the measured RTX A5000 regime. It means that GPU remains thousands of times faster in absolute time, but the speedup decreases over the measured range because of memory traffic or occupancy limits become visible.

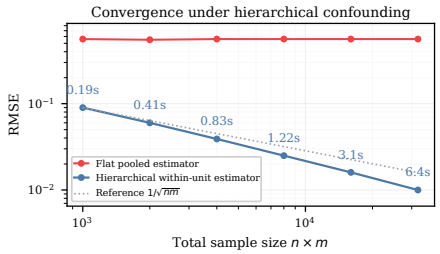

Figure 4: Convergence diagnostic under hierarchical confounding. The flat pooled estimator does not approach the intervention target because the unit-level latent cause is ignored. The hierarchical within-unit estimator removes the unit effect and follows the expected $1/\sqrt{nm}$ decay.

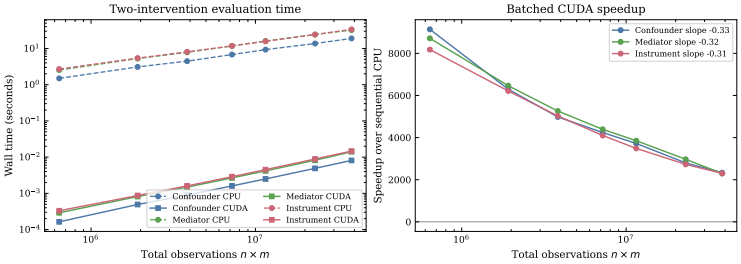

Figure 5: GPU speedup on synthetic HSCM evaluations. The left panel reports wall-clock time for sequential CPU evaluation and batched CUDA evaluation. The right panel reports speedup over sequential CPU; legend labels include the fitted log–log slope.

5 Real data analysis: Project STAR

5.1 Data and hierarchical representation

Project STAR randomized kindergarten students within schools to small classes, regular classes, or regular classes with an aide. The full STAR-and-Beyond public-use dataset is documented by Achilles et al. (2008) and distributed through Harvard Dataverse at https://doi.org/10.7910/DVN/SIWH9f; it contains the raw student- and school-level records from the longitudinal Tennessee experiment, including demographics, class assignments, school identifiers, teacher information, and achievement outcomes. After removing missing values, the kindergarten cohort contains 5,745 student records across 322 classes and 79 schools; for the class-as-unit HSCM analysis and diagnostics, the scripts then use a balanced sample of 10 students per class, yielding 3,220 student records across 322 classes. The main outcome in this paper is Mathematics, written Y. Reading, written B, is retained as a pre-outcome achievement mechanism because it enters some discovered graph structures and appears in the identified HSCM formulas. Classes are treated as units and students as sub-units. The treatment is the student-level small-class assignment, written $A_{ij} = \mathbf{1}\{\text{small class}\}$, with student covariates given by Gender (G_{ij}), Ethnicity (E_{ij}), and free-lunch status (L_{ij}). School urbanicity (S_i) is treated as an observed class-level covariate, while U_i collects all unobserved unit-level causes, including but not limited to class heterogeneity, latent school context, teacher effects, and other shared classroom factors.

The intervention is not a row-level replacement of A_{ij}, but a class-level intervention $\mathrm{do}(Q^a = q^a_\star)$ that fixes the small-class assignment distribution within the class. In plain terms, the Q-quantities used below are class summaries of student-level variables: Q^a is the small-class assignment probability or proportion in a class, Q^g is the gender composition, and Gaussian score mechanisms such as the Mathematics outcome Q^y or the Reading mechanism Q^b are represented by fitted class-specific means and variances.

Figure 6 illustrates the main class-level diagnostics. The mathematics Q-Q panel compares the standardized empirical score quantiles with a Gaussian reference and highlights the central range separately from the lower and upper extremes. The upper extreme contains only a few repeated score values, so it is shown as an empirical diagnostic rather than fitted with a separate tail model.

5.2 Baseline regression analysis

As baselines, we reproduce standard student-level regression contrasts using the notation introduced in Table 2, following the public STAR regression replication code of Zhang et al. (2025). Let A_{ij} denote assignment of student j in class i to a small class, let R_{ij} denote assignment to a regular class with an aide, let Y_{ij} be the kindergarten Mathematics outcome, and let $s(i)$ denote the school of class i. The OLS specifications $\mathcal{M}^1$-$\mathcal{M}^4$ all report $\hat{\beta}_A$, the coefficient of the small-class assignment indicator A_{ij}.

Table 3: Baseline regression estimates for kindergarten mathematics. For M1–M4 and M6, the reported coefficient is $\hat{\beta}_A$, the coefficient of the small-class assignment indicator A_{ij}. For M5, the reported coefficient is the IV class-size coefficient $\hat{\theta}$.

Model	Specification	Coef.	SE	p-value	R^2
$\mathcal{M}^1$	No controls	$\hat{\beta}_A$=8.096	1.597	4.0×10^{-7}	0.006
$\mathcal{M}^2$	School fixed effects	$\hat{\beta}_A$=9.499	1.463	8.5×10^{-11}	0.219
$\mathcal{M}^3$	Student controls	$\hat{\beta}_A$=9.380	1.415	3.4×10^{-11}	0.268
$\mathcal{M}^4$	Full OLS replication	$\hat{\beta}_A$=9.432	1.419	3.0×10^{-11}	0.269
$\mathcal{M}^5$	IV class-size contrast	$\hat{\theta}$=-1.219	0.163	7.0×10^{-14}	–
$\mathcal{M}^6$	Random class intercept	$\hat{\beta}_A$=9.099	2.263	5.8×10^{-5}	–

The pooled OLS contrast is

$$Y_{ij} = \alpha + \beta_A A_{ij} + \beta_R R_{ij} + \varepsilon_{ij}, \tag{$\mathcal{M}^1$}$$

the second OLS model adds school fixed effects,

$$Y_{ij} = \alpha + \beta_A A_{ij} + \beta_R R_{ij} + \delta_{s(i)} + \varepsilon_{ij}, \tag{$\mathcal{M}^2$}$$

the third one adds observed student controls,

$$X_{ij}^{\text{stu}} = (G_{ij}, E_{ij}, L_{ij}),$$

$$Y_{ij} = \alpha + \beta_A A_{ij} + \beta_R R_{ij} + \gamma_{\text{stu}}^\top X_{ij}^{\text{stu}} + \delta_{s(i)} + \varepsilon_{ij}, \tag{$\mathcal{M}^3$}$$

and finally, we consider the full OLS replication. It adds teacher controls,

$$X_{ij}^{\text{full}} = (G_{ij}, E_{ij}, L_{ij}, \text{teacher race}, \text{teacher experience}, \text{teacher degree}),$$

and estimates

$$Y_{ij} = \alpha + \beta_A A_{ij} + \beta_R R_{ij} + \gamma_{\text{full}}^\top X_{ij}^{\text{full}} + \delta_{s(i)} + \varepsilon_{ij}, \tag{$\mathcal{M}^4$}$$

The coefficient β_A is therefore a student-level small-class contrast, conditional on the chosen controls.

The instrumental-variable replication, denoted $\mathcal{M}^5$, uses assignment indicators as instruments for actual class size C_{ij}. It decomposes into two steps:

$$C_{ij} = \pi_0 + \pi_A A_{ij} + \pi_R R_{ij} + \Pi^\top X_{ij} + \delta_{s(i)} + \nu_{ij}, \tag{$\mathcal{M}_1^5$}$$

$$Y_{ij} = \alpha + \theta \widehat{C}_{ij} + \gamma^\top X_{ij} + \delta_{s(i)} + \varepsilon_{ij}. \tag{$\mathcal{M}_2^5$}$$

We get then a class-size contrast estimate $\hat{\theta}$ rather than another estimate of β_A.

The corresponding non-causal hierarchical regression baseline, denoted $\mathcal{M}^6$, can be written as a random-intercept or Bayesian OLS model:

$$Y_{ij} = \alpha + \beta_A A_{ij} + \beta_R R_{ij} + \gamma^\top X_{ij} + b_{\text{class}(i)} + \varepsilon_{ij}, \qquad b_c \sim \mathcal{N}(0, \sigma_b^2). \tag{$\mathcal{M}^6$}$$

We fit M6 with the same fixed controls as M4 and a random intercept at the class level. The resulting small-class coefficient is $\hat{\beta}_A = 9.099$ (SE 2.263), so the hierarchical regression baseline remains close to the OLS small-class contrasts but with larger uncertainty.

This model is valuable because it acknowledges between-class heterogeneity, but its coefficient β_A remains a regression coefficient with a random class intercept; it treats heterogeneity as a residual random effect rather than as a distributional causal mechanism.

Results are given in Table 3. The models which are estimating $\hat{\beta}_A$ ($\mathcal{M}^i$ for $i \in \{1, 2, 3, 4, 6\}$) are providing similar results, but we trust more the value of $\mathcal{M}^6$ which is benefiting of the hierarchical structure of the data. The IV model $\mathcal{M}^5$ is giving another information, with a class-size contrast estimate. The sign of the estimator, negative, is expected.

Table 4: Class-level Q-variables used by the STAR HSCM evaluator. These are distributional summaries or mechanisms fitted at the class level from the student-level variables in Table 2.

Symbol	Type	Interpretation
Q^a	Bernoulli/proportion	Small-class assignment distribution within a class
Q^g	Bernoulli/proportion	Gender composition within a class
Q^e	Bernoulli/proportion	Ethnicity composition within a class
Q^l	Bernoulli/proportion	Free-lunch composition within a class
Q^b	Gaussian score mechanism	Reading score distribution for the class
Q^y	Gaussian score mechanism	Mathematics score distribution for the class
$Q^{b \mid a,g}$	Conditional mechanism	Reading mechanism conditional on treatment and gender composition
$Q^{y \mid e,b}$	Conditional mechanism	Mathematics mechanism conditional on ethnicity and reading mechanisms

However, in the previous standard baselines, the heterogeneity is absorbed into fixed effects, controls, or random intercepts; it is not decomposed into class-level Q-mechanisms such as the treatment distribution Q^a, gender composition Q^g, or response factors depending jointly on treatment and composition. For STAR, the scientific question is not only whether an individual assigned to a small class has a different expected score, but how a class-level intervention on the distribution of small-class assignment propagates through heterogeneous classroom mechanisms. This distinction motivates the use of HSCM below.

5.3 Causal graph discovery and associated HSCM

The STAR HSCM analysis starts from graph structures learned on the flat discovery variables A, B, Y, G, E, L and S, since no causal graph is assumed a priori. We compared PC (Spirtes et al., 2000), FCI (Spirtes et al., 1995), DirectLiNGAM (Shimizy et al., 2011), and ExactBIC (Yuan and Malone, 2013). The graph reported in Figure 7 (left) is the ExactBIC graph selected for the HSCM analysis: it is not claimed to be the unique recovered causal graph, but it was the most informative discovered structure for the STAR translation, because treatment, school context, class composition, reading, and mathematics remained connected after collapse and augmentation.

For the ExactBIC run, the flat graph is first read as a directed graph over the observed STAR variables. The HSCM construction then adds the latent class-level context used in the analysis, in particular the unobserved class heterogeneity node U and the school-context node S. Since the outcome of interest is kindergarten Mathematics, the graph is then collapsed and augmented for the class-level Mathematics outcome mechanism Q^y. This gives the Q-variable graph shown in the right panel of Figure 7.

The STAR HSCM analysis starts from graph structures learned on the flat discovery variables A, B, Y, G, E, L and S, as there is no a priori knowledge on the causal graph. We compared four discovery procedures on this flat representation: PC (Spirtes et al., 2000), FCI (Spirtes et al., 1995), DirectLiNGAM (Shimizy et al., 2011), and ExactBIC (Yuan and Malone, 2013). The graph reported in Figure 7 is the ExactBIC graph, not because it is claimed to be the uniquely recovered causal graph, but because among the discovered candidates it gave the most informative structure for the HSCM translation: the treatment, achievement, composition, and school-context variables remained connected in a way that produced a non-degenerate class-level functional after collapse and augmentation. For the ExactBIC run, we used one score-based DAG stored in the discovery output, with the school-urbanicity/ethnicity relation oriented as Urbanicity (S) $\rightarrow$ Ethnicity (E) for substantive interpretability. This is one plausible graph, not a claim that the discovery step has recovered a unique causal structure.

Before identification, the HSCM construction also adds the latent class-level confounding edges Unobserved class heterogeneity (U) $\rightarrow$ Small-class assignment (A) and Unobserved class heterogeneity (U) $\rightarrow$ Mathematics outcome (Y), because the target outcome in this paper is Mathematics. The resulting hierarchical graph is then collapsed and augmented for Q^y (the class-level Mathematics outcome mechanism), giving the Q-variable graph in Figure 7. The new variables, introduced in the collapsed and augmented graph, are summarized in Table 4.

Table 5: STAR HSCM estimates and normalized-factor diagnostic. The Mathematics row is the main outcome analysis, where the outcome is denoted Y; the Reading row is a separate diagnostic run used to inspect the normalized $P(Q^{b \mid a,g} \mid S)$ factor.

Run	Target	$E[\cdot \mid \mathrm{do}(1)]$	$E[\cdot \mid \mathrm{do}(0)]$	ATE
ExactBIC	Mathematics outcome (Y)	2402.75	2365.95	36.80
ExactBIC, normalized factors	Reading (B) diagnostic	1788.10	1766.47	21.63

5.4 Identified formulae

In the second step, the transformed graph is passed to `pyAgrum` to identify the query. It gives the following functional after the translated graph has been collapsed and augmented for the mathematics outcome:

$$\tau_{\mathrm{BIC}}(q^a_\star) = \sum_{q^e,q^g,q^{y \mid e,b},q^{b \mid a,g},s} \mathrm{p}(q^e \mid s)\mathrm{p}(q^{b \mid a,g} \mid s)\mathrm{p}(q^{y \mid e,b})\mathbb{E}\left[Q^y \mid Q^a = q^a_\star, Q^e, Q^g, Q^{y \mid e,b}, Q^{b \mid a,g}\right] \mathrm{p}(q^g)\mathrm{p}(s). \tag{4}$$

In this expression, $\mathrm{p}(q^{y \mid e,b})$ is the class-level Mathematics outcome distribution induced by the discovered Reading (B) $\to$ Mathematics outcome (Y) and Ethnicity (E) $\to$ Mathematics outcome (Y) parents, $\mathrm{p}(q^e \mid s)$ encodes the substantively oriented Urbanicity (S) $\to$ Ethnicity (E) relation, and $\mathrm{p}(q^{b \mid a,g} \mid s)$ comes from the Reading (B) mechanism with Small-class assignment (A), Gender (G), and Urbanicity (S) in the translated graph. The expectation term is evaluated twice, at $Q^a = 1$ and $Q^a = 0$, and the remaining Q-variables are marginalized by the product of fitted density factors. This is a do-calculus functional in the sense of Pearl (2009) and Weinstein and Blei (2026); its product weights are also close in spirit to importance-sampling and inverse-probability weighting constructions (Rubinstein and Kroese, 2016; Horvitz and Thompson, 1952; Robins et al., 2000). The formula therefore makes clear why multi-parent Q-estimation matters: the identified effect is a product of conditional densities and outcome-response factors, not a single regression coefficient.

5.5 Estimation

Finally, in the third step, this query is estimated with the fitted model families used in the run: Bernoulli models for Small-class assignment (A), Gender (G), Ethnicity (E), and free lunch (L), a categorical model for Urbanicity (S), and two-component Gaussian mixtures for Reading (B) and Mathematics outcome (Y). After this translation, the class-as-unit HSCM model gives the estimates reported in Table 5, together with the separate normalized-factor diagnostic for the Reading mechanism.

These values should not be read as direct replacements for the student-level OLS coefficient, but the comparison is still essential. Across the three OLS specifications with school fixed effects or student controls, the small-class coefficient is approximately 9 to 9.5 mathematics points, whereas the ExactBIC HSCM mathematics contrast in Table 5 is 36.80 points. OLS gives a student-level contrast for a binary treatment indicator, conditional on observed covariates and school fixed effects; IV gives a class-size contrast using assignment as an instrument. The HSCM estimand is instead a class-level intervention on Q^a, with additional dependence through class-specific distributions. The fact that the ExactBIC HSCM contrast remains much larger than the OLS coefficient is a warning sign: once the hierarchy is ignored, the target no longer contains the class-composition mechanisms that drive the HSCM functional. The HSCM magnitude therefore describes the behaviour of the translated hierarchical estimand, while the separate normalized-factor diagnostic below shows that the numerical result is sensitive to the scaling of multiplicative factors retained in the identified formula.

We also report the speed up gain with our parallel implementation with a CPU. In sequential, the evaluation take 3.11 seconds, while with 4 processes it took 1.9 seconds, with a speed up of 1.64.

Figure 8 shows that the response mechanisms are not constant across classes. For both Reading and Mathematics, classes with larger fitted means also tend to have larger fitted variances, so the Gaussian Q-summaries capture both level differences and heteroskedasticity across classrooms. This matters for the HSCM numerical estimator because an intervention on the class-level treatment distribution is evaluated through these class-specific response mechanisms, not through a single pooled outcome regression. The

right panel shows a strong positive association between class-level Reading and Mathematics means, indicating that the two achievement mechanisms share substantial classroom-level structure. Consequently, the mathematics effect should not be interpreted as an isolated student-level coefficient: it is evaluated in a distribution of classrooms whose baseline achievement and score variability differ markedly.

5.6 Effective graphs and factor normalization

The normalized-factor diagnostic in Table 5 makes explicit the difference between a nominal identified functional and the stabilized functional actually evaluated by the numerical code. This diagnostic is needed because the effective sample size for HSCM estimation is the number of classes, not the number of student rows. In the balanced STAR analysis there are 322 classes and only 10 sampled students per class, so class-level mechanisms with several parents are estimated with substantial uncertainty. When such uncertain mechanisms enter a formula as multiplicative probability terms, a single poorly estimated term can dominate the final numerical product. For ExactBIC, the diagnostic run concerns the Reading (Y) mechanism and the normalized term was $P(Q^{y \mid a,g} \mid S)$, which is the distribution of the class-level Reading mechanism conditional on Small-class assignment (A) and Gender (G), further conditioned by Urbanicity (S). Normalizing such a term does not remove variables from the nominal causal graph and should not be read as a new estimand selected after seeing the answer. It is a stability analysis: it asks whether the qualitative contrast is being carried by the response mechanism itself or by the raw scale of one uncertain fitted probability term. The effective graph in Figure 7 visualizes this distinction: dashed red arrows are dependencies present in the nominal graph whose factor contribution is normalized in the stabilized functional.

The ExactBIC gender factor analysis clarifies what this means numerically. In Table 5, the normalized factor $P(Q^{y \mid a,g} \mid S)$ is stabilized before the two intervention evaluations, so its raw scale does not dominate the product. The corresponding factor-level traces vary on roughly 10^3 scales in the raw product, which is precisely the numerical symptom that motivates the normalized analysis. It nevertheless marks an important part of the estimand: the graph says that the class-level Reading (Y) response should depend on Small-class assignment (A), Gender (G) composition, and Urbanicity (S). When the same run is analysed in unit context, the only non-zero factor change comes from the fitted response term $P(Q^y \mid Q^a, Q^g, Q^{y \mid a,g})$, namely Reading ($Y$) conditional on Small-class assignment (A), Gender (G), and the class-level Reading mechanism $Q^{y \mid a,g}$, whose mean changes from 434.83 under $\mathrm{do}(Q^a = 0)$ to 440.15 under $\mathrm{do}(Q^a = 1)$. The resulting delta, 5.32, explains why the gender mechanism matters even when the conditional factor is normalized for stability.

5.7 Diagnostics and interpretation

The STAR analysis also illustrates why diagnostics are essential. The experiments expose a central operational issue: an identified formula may contain multi-parent Q-variables whose density factors are numerically unstable. In a small class-level dataset, these fitted terms can carry large estimation uncertainty; the normalized-factor runs rescale them to check whether the final contrast is stable to this raw product scale. In the ExactBIC STAR run, the normalized factor was the gender-dependent term $P(Q^{y \mid a,g} \mid S)$. The diagnostic is scientifically useful only when it is read at the factor level: factor levels show which terms dominate the product numerically, while unit-context factor changes show which term changes between the two interventions.

The gender-composition diagnostic in Figure 6 shows that class composition varies substantially across classes; an OLS coefficient on `female` adjusts for an average individual difference, but it does not represent the distribution of class composition. HSCM explicitly represents such within-class distributions through the estimated probability Q-quantities in Figure 9 and the response parameters and means in Figure 8. The ExactBIC factor analysis above shows how gender composition enters the intervention path through $Q^{y \mid a,g}$. The current implementation remains two-level and treats schools only through unit-level covariates rather than as a full student–class–school hierarchy. This is the main substantive lesson of the STAR analysis. Earlier non-causal hierarchical or graphical summaries could describe that classes differ, but they did not make class heterogeneity part of an identified intervention functional. By combining class-level Q-heterogeneity with factor analysis of the identified formula, the HSCM pipeline shows that part of the small-class effect can be hidden when class composition and multi-parent distributional factors are treated only as background variation. In that sense, the previous hierarchical graph analysis

tended to smooth or understate the intervention contrast, whereas the causal HSCM analysis gives a new interpretation of STAR: the effect of class size is not only an average student-level coefficient, but a distributional class-level effect whose magnitude depends on which class mechanisms are included, estimated, or normalized in the numerical functional.

The substantive conclusion is therefore conservative. HSCM does not simply "improve" OLS by producing a larger number; it changes the estimand from a student-level coefficient to a distributional class-level intervention. In STAR this change is scientifically meaningful, but it also makes graph choice, positivity, and completeness of the fitted local models part of the reported result rather than secondary implementation details.

6 Conclusion

We developed a scalable automatic identification and estimation implementation for Hierarchical Structural Causal inference. The pipeline starts from a hierarchical graph, applies the HSCM transformations needed for identification, obtains a symbolic do-calculus estimand from `pyAgrum`, adapts it into closed-form HSCM formulas, and evaluates the associated AST using fitted local probability and response models. This turns HSCM from a formal identification framework into an operational workflow for nested data. The key computational point is that the AST does not leave the formula as an opaque expression: it decomposes the estimand into many local conditional densities, response models, and marginalization branches that can be estimated independently and then recombined into a single causal functional. The implementation is available at `AI-vidence/hierarchicalcausalmodels`, so the graph transformations, identification calls, estimators, diagnostics, and STAR replication artefacts are reproducible.

The experiments validate the pipeline on known structures and reveal the practical regimes in which estimation is stable. The STAR application demonstrates the methodological payoff: when the intervention is assigned at the class level and outcomes are measured at the student level, the natural estimand is hierarchical. OLS, IV, and non-causal hierarchical summaries remain essential benchmarks, but they do not encode interventions on within-class treatment distributions or class-composition mechanisms. The same framework also clarifies why large hierarchical applications are computationally feasible: adding units, factors, or sampled branches increases the number of local estimation tasks, but it does not require redefining the estimand or replacing it by a flat pooled regression.

The remaining limitations point directly to the next statistical developments. The present implementation is two-level, whereas STAR is naturally a student–class–school system; extending scalable HSCM estimation to deeper hierarchies is therefore necessary for a fully faithful education application. Multi-parent Q-density estimation is the main statistical bottleneck, because missing or weakly estimated factors change the effective estimand rather than merely adding numerical noise. Uncertainty quantification for normalized or otherwise stabilized functionals also remains open. future work should therefore develop robust multi-parent estimators, extend the implementation to deeper hierarchies, preserve the parallel structure of the AST evaluator, and connect the numerical backend to symbolic DSLs such as `y0` so that formula manipulation and estimation can share a common representation. For discovery-driven applications, the next step is to evaluate families of plausible DAGs rather than a single graph, using cluster-DAG ideas to group graphs that agree on coherent blocks of variables and to report how the HSCM estimand changes across those graph clusters.

Author contributions

All authors contributed to the methodology. A.E., Y.O., S.P., A.E.O., A.D. and J.A. contributed to the software devlopment and experiments running. All authors contributed to the analysis of the results. D.C., J.A, E.D. and M.C. contributed to the writing.

Data availability

The full Project STAR public-use data are available as the STAR-and-Beyond dataset documented by Achilles et al. (2008) and hosted by Harvard Dataverse at this link. Code and generated artefacts are maintained at github.com/AI-vidence/hierarchicalcausalmodels.

References

C. Achilles, H. P. Bain, f. Bellott, J. Boyd-Zaharias, J. finn, J. folger, J. Johnston, and E. Word. Tennessee's Student Teacher Achievement Ratio (STAR) Project. STAR-and-Beyond public-use data, Harvard Dataverse, 2008. Available at this link.

J. D. Angrist, G. W. Imbens, and D. B. Rubin. Identification of causal effects using instrumental variables. *Journal of the American Statistical Association*, 91(434):444–455, 1996.

D. Card. Using geographic variation in college proximity to estimate the return to schooling. In *Aspects of Labour Market Behaviour: Essays in Honour of John Vanderkamp*. University of Toronto Press, 1995.

R. Chetty, J. N. friedman, N. Hilger, E. Saez, D. W. Schanzenbach, and D. Yagan. How does your kindergarten classroom affect your earnings? Evidence from Project STAR. *The Quarterly Journal of Economics*, 126(4):1593–1660, 2011.

C. Gonzales, L. Torti, and P.-H. Wuillemin. aGrUM: a graphical models framework. In *Proceedings of the 10th International Conference on Scalable Uncertainty Management*, 2017.

D. G. Horvitz and D. J. Thompson. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47(260), 663–685, 1952.

G. W. Imbens and D. B. Rubin. *Causal Inference for Statistics, Social, and Biomedical Sciences*. Cambridge University Press, 2015.

A. B. Krueger. Experimental estimates of education production functions. *The Quarterly Journal of Economics*, 114(2):497–532, 1999.

A. B. Krueger and D. M. Whitmore. The effect of attending a small class in the early grades on college-test taking and middle school test results. *The Economic Journal*, 111(468):1–28, 2001.

J. Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2nd edition, 2009.

J. M. Robins, M. A. Hernan and B. Brumback. Marginal structural models and causal inference in epidemiology. *Epidemiology*, 11(5), 550–560, 2000.

R. Y. Rubinstein and D. P. Kroese. *Simulation and the Monte Carlo Method*. Wiley, 3rd edition, 2016.

S. Shimizu, T. Inazumi, Y. Sogawa, A. Hyvärinen, Y. Kawahara, T. Washio, P. O. Hoyer and K. Bollen. DirectLiNGAM: A direct method for learning a linear non-Gaussian structural equation model. *Journal of Machine Learning Research* 12: 1225–1248, 2011.

P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. 2nd ed. Cambridge, MA: MIT Press, 2000.

P. Spirtes, C. Meek and T. Richardson Causal inference in the presence of latent variables and selection bias. Proceedings of the Eleventh conference on Uncertainty in artificial intelligence, 1995.

y0 contributors. y0: causal inference in Python, hierarchical module and HSCM pull request. Available at this link, 2025.

E. N. Weinstein and D. M. Blei Hierarchical Structural Causal models. *Journal of Machine Learning Research*, 27:1–73, 2026.

C. Yuan and B. Malone. Learning optimal Bayesian networks: A shortest path perspective. *Journal of Artificial Intelligence Research*, 48, 23-65, 2013.

C. Zhang, K. Lee, R. Duan, X. Sang. STA-207: public Project STAR regression replication code. Available at this link, 2025.

Symbol	Level	Definition
A	student	Small-class assignment
Y	student	Mathematics score
B	student	Reading score
G	student	Gender indicator
E	student	Ethnicity indicator
L	student	free-lunch status
S	class	School urbanicity
U	latent	Unobserved class heterogeneity

Table 2: STAR variables used in the HSCM analysis.

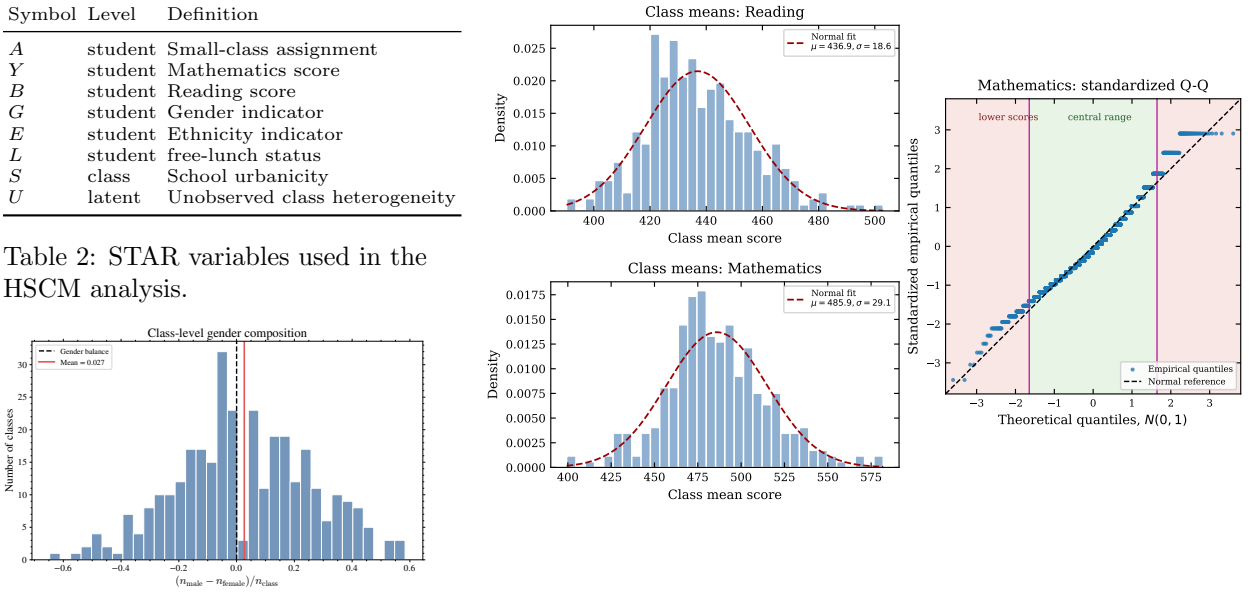

Figure 6: Class-level STAR diagnostics. Left: gender heterogeneity across classes, showing why an additive individual gender coefficient does not represent the full distribution of class composition. Right: reading and mathematics class-level score means, together with a standardized Q-Q diagnostic for Mathematics. In the Q-Q panel, blue points are empirical quantiles and the dashed black line is the Gaussian reference. The green band marks the central range where the Gaussian approximation is most relevant, while the red bands mark lower and upper score extremes. The rightmost extreme is kept as an empirical check rather than fitted separately because it contains only a few repeated score values.

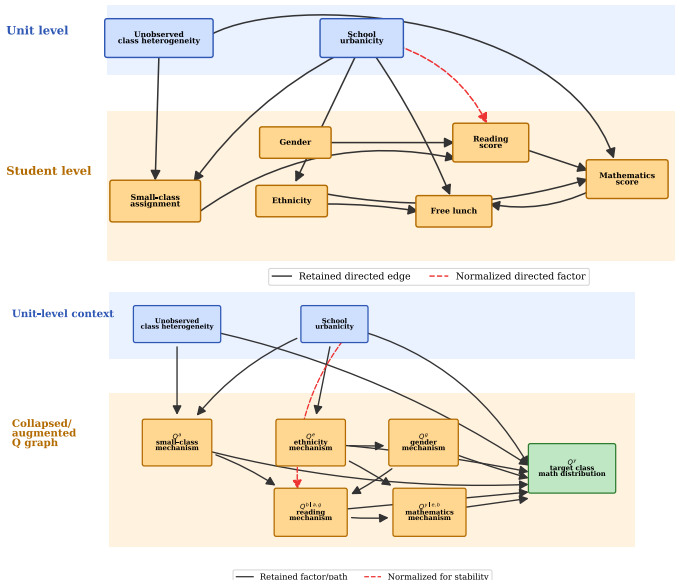

Figure 7: ExactBIC-derived graphs for the STAR HSCM analysis. The left panel shows the directed ExactBIC graph over the flat STAR variables, after adding the unit-level context used by the HSCM analysis. The right panel shows the corresponding collapsed and augmented Q-variable graph for the Mathematics outcome. Blue nodes are unit-level context variables, orange nodes are observed variables or fitted Q-mechanisms, and the green node is the target class-level Mathematics outcome distribution Q^y. Solid arrows are retained directed dependencies; dashed red arrows mark factors normalized in the stabilized numerical evaluation.

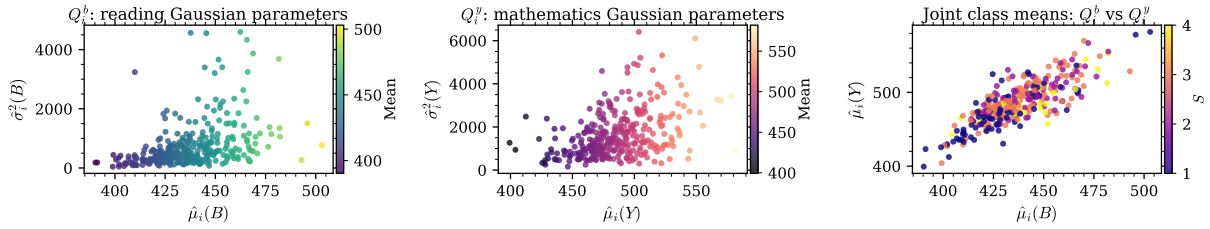

Figure 8: Estimated class-level HSCM response parameters and means in STAR. Reading (Q^b) and the Mathematics outcome (Q^y) are represented by Gaussian summaries ($\hat{\mu}_i, \hat{\sigma}_i^2$), and the right panel reports the joint class means used to inspect the relationship between the two score mechanisms. In the right panel, colors encode the class-level urbanicity category S; the panel is a descriptive diagnostic rather than an oriented causal graph.

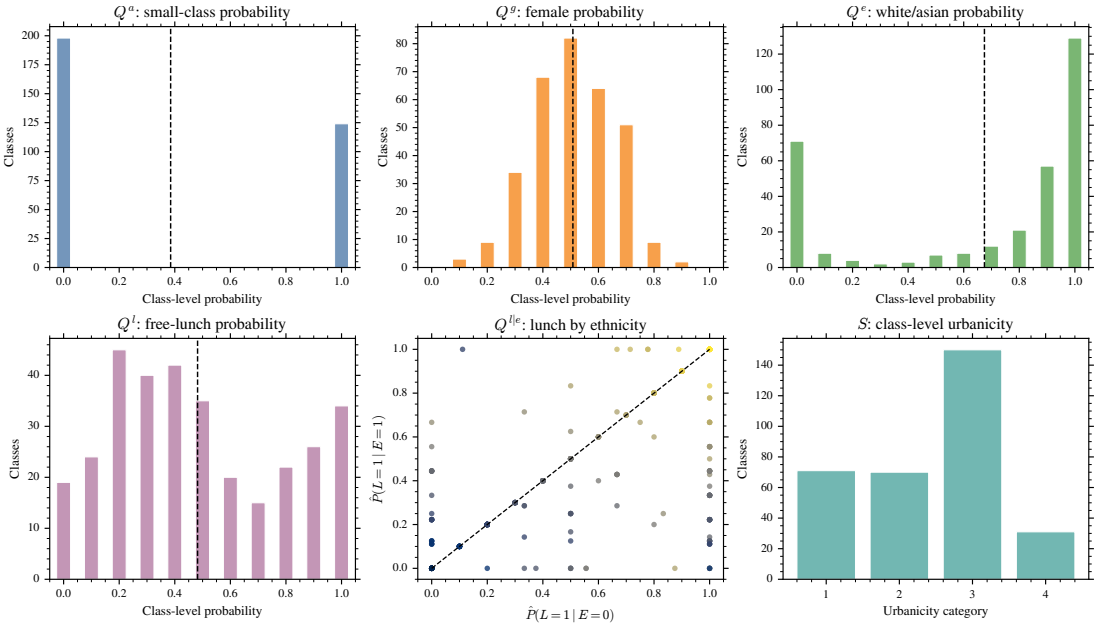

Figure 9: Estimated class-level HSCM probability quantities used in the STAR analysis. Binary sub-unit variables are represented by class-level Bernoulli probabilities for Small-class assignment (Q^a), Gender (Q^g), Ethnicity (Q^e), and free lunch (Q^l); $Q^{l \mid e}$ summarizes the free-lunch mechanism conditional on Ethnicity; and S is the observed class-level Urbanicity variable. In the conditional free-lunch scatter, point colors encode the urbanicity category S; in the S panel itself, the bar color is only a visual grouping choice.