1 Empirical demonstration (appendix section)

For the purposes of empirically demonstrating and evaluating our algorithm, we use a simple setting with a single treatment variable and a single outcome variable, i.e., $\mathbf{X} = \{X\}$ and $\mathbf{Y} = \{Y\}$; thus, we use unbolded fonts in this section. We run our experiments in the standard statistical setting, where each algorithm is provided with a dataset of n samples, rather than being given sample access to the distribution $\mathbb{P}(\mathbf{V})$. The code for replicating our experiments can be found at `https://anonymous.4open.science/r/amba-bamba-submit-4932/README.md`.

In Section 1.1, we describe the baseline algorithm against which we compare, along with some details about how these algorithms are implemented. In Section 1.2, we describe our procedure for synthetic data generation, and in Section 1.3, we describe our evaluation metrics and our approach for hyperparameter selection. We conclude in Section 1.4 by discussing our results.

1.1 Comparisons and Implementation Details

We compare our AMBA and BAMBA methods to three other methods:

- Z-ADJUST: Adjust by the given adjustment set $\mathbf{Z}$
- MB-ADJUST: Adjust by the Markov blanket of X in the (unknown) data-generating graph $\mathcal{G}$
- MIN-ADJUST: Adjust by the minimum-sized adjustment set for $\mathbb{P}_x(y)$ in the (unknown) data-generating graph $\mathcal{G}$

We note that MB-ADJUST and MIN-ADJUST require *oracle* knowledge of the graph and are not possible to run in practical (non-synthetic) scenarios; we have included them here for the sake of comparison.

Implementation details One of our focuses in this empirical demonstration is to slightly adapt the methods to better reflect how they are used in practice. Our methods and others use two key subroutines: *conditional independence testing* and *estimation by covariate adjustment*, which both require certain design choices.

First, let us describe our estimators of marginal and conditional probabilities used for both subroutines. Let $\mathbf{A}$, $\mathbf{B}$ be disjoint sets. For each $\mathbf{a}$ and $\mathbf{b}$, let $n_{\mathbf{a}}$ denote the number of samples where $\mathbf{A} = \mathbf{a}$, and let $n_{\mathbf{a},\mathbf{b}}$ denote the number of samples where $\mathbf{A} = \mathbf{a}$ and $\mathbf{B} = \mathbf{b}$. Then, given $\beta \geq 0$, for all $\mathbf{a}$ and $\mathbf{b}$, we define

$$\widehat{P}_\beta(\mathbf{a}) := \frac{n_{\mathbf{a}} + \beta}{n + \beta \cdot |\mathbf{\Sigma}_{\mathbf{A}}|} \tag{1}$$

and

$$\widehat{P}_\beta(\mathbf{a} \mid \mathbf{b}) := \frac{n_{\mathbf{a},\mathbf{b}} + \beta}{n_{\mathbf{b}} + \beta \cdot |\mathbf{\Sigma}_{\mathbf{A}}|}, \tag{2}$$

where $\frac{0}{0} := 0$. This technique, which can be interpreted as adding "pseudo-counts" for each value of $\mathbf{A}$, is known as *additive* or *Laplace* smoothing, and is well-known to have favorable statistical properties when β is sufficiently small [Kamath et al., 2015]. Hence, it is likely that our theory extends to such an estimator. For the sake of estimation by covariate adjustment, given the set $\mathbf{A}$, we use the following estimator of $T_{\mathbf{A},x,y}$:

$$\widehat{T}_{\mathbf{A},x,y} := \sum_{\mathbf{a}} \widehat{P}_\beta(\mathbf{a}) \cdot \widehat{P}_\beta(\mathbf{y} \mid \mathbf{a}, x).$$

For the sake of conditional independence testing, we find that a simple *plug-in* estimator of $\Delta_{\mathbf{A} \perp\!\!\!\perp \mathbf{B} \mid \mathbf{C}}$ worked better than the more complicated conditional independence testers cited in our theoretical analysis. In particular, we use the following estimator:

$$\widehat{\Delta}_{\mathbf{A} \perp\!\!\!\perp \mathbf{B} \mid \mathbf{C}} := \sum_{\mathbf{c}} \widehat{P}_0(\mathbf{c}) \cdot \left| \widehat{P}_0(\mathbf{a}, \mathbf{b} \mid \mathbf{c}) - \widehat{P}_0(\mathbf{a} \mid \mathbf{c}) \cdot \widehat{P}_0(\mathbf{b} \mid \mathbf{c}) \right|.$$

Then, we have a hyperparameter τ controlling the threshold for conditional independence; i.e., $\widehat{\Delta}_{\mathbf{A} \perp\!\!\!\perp \mathbf{B} \mid \mathbf{C}} > \tau$, we reject conditional independence. For a given choice of $\mathbf{A}, \mathbf{B}$, and $\mathbf{C}$, this estimator requires computing the empirical counts $n_{\mathbf{a},\mathbf{b},\mathbf{c}}$. In our algorithms, such computations can often be re-used, e.g. $n_{\mathbf{a},\mathbf{b},\mathbf{c}} = n_{\mathbf{a}',\mathbf{b}',\mathbf{c}'}$ whenever $\mathbf{A} \cup \mathbf{B} \cup \mathbf{C} = \mathbf{A}' \cup \mathbf{B}' \cup \mathbf{C}'$. Thus, our implementation performs *memoization* of such empirical counts to reduce computational complexity.

Finally, MIN-ADJUST requires an algorithm for finding the minimum-sized adjustment set. For this step, we use the `optimaladj` package, which implements the methods described in Smucler and Rotnitzky [2022].

1.2 Synthetic data generation

We use a single 10-node DAG structure $\mathcal{G}$ inspired by Figure 1, with $M = 50$ different distributions $(P_m(\mathbf{V}))_{m=1}^{50}$ on this structure. For each $m = 1, 2, \ldots, 50$, we sample $D = 10$ datasets, each of size $n = 500$. In more detail:

> *Structure.* We use a fixed 10-node DAG $\mathcal{G}$ with $\mathbf{X} = \{X\}$, $\mathbf{Y} = \{Y\}$, $|\mathbf{Z}| = 8$, $|\mathrm{pa}(x)| = 6$, and $|\mathbf{S}_{\min}| = 2$, where $\mathbf{S}_{\min}$ is the unique minimum-sized valid adjustment set for $P_x(y)$. In particular, the DAG has the following edges and no others: $X \to Y$, $V \to X$ for each $V \in \mathrm{pa}(X)$, $W \to V$ for each $W \in \mathbf{S}_{\min}$ and $V \in \mathrm{pa}(X)$, and $W \to Y$ for each $W \in \mathbf{S}_{\min}$.
>
> *Distribution, Step 1: Sampling from a beta prior.* For each $m = 1, 2, \ldots, 50$, we construct a distribution $\mathbb{P}^{(m)}(\mathbf{V})$ as follows. We take all variables in $\mathbf{V}$ to be binary. For each $V_i \in \mathbf{V}$ and each value $\mathbf{w} \in \{0,1\}^{|\mathrm{pa}(V_i)|}$, we sample $p_{\mathbf{w}} \sim \mathsf{Beta}(1,1)$, and assign $\mathbb{P}^{(m)}(V_i \mid \mathrm{pa}(V_i) = \mathbf{w}) = p_{\mathbf{w}}$.

Algorithm	Smoothing coefficient β	CI threshold τ
Z-ADJUST	0.1	-
MB-ADJUST	0.1	-
MIN-ADJUST	0.1	-
AMBA	0.1	0.2
BAMBA	0.1	0.1

Table 1: **Selected hyperparameters for the tested algorithms.**

Distribution, Step 2: Tilting the distribution of Y. By symmetries of the beta distribution, the true causal effect $\mathbb{P}_x^{(m)}(y)$ is likely to be close to 0.5; this creates and artificial bias towards estimator with (implicit) shrinkage toward 0.5. Thus, we *tilt* the distribution of Y toward higher values by taking $\widetilde{\mathbb{P}}^{(m)}(Y \mid \text{pa}(Y) = \mathbf{w}) \propto \mathbb{P}^{(m)}(Y \mid \text{pa}(Y) = \mathbf{w}) \cdot e^{2Y}$ for all $\mathbf{w} \in \{0,1\}^{|\text{pa}(Y)|}$.

Samples. For each $m = 1, 2, \ldots, 50$, we draw $D = 10$ datasets of $n = 500$ samples, giving us an indexed family $\{\mathcal{D}^{(m,d)}\}_{m=1,d=1}^{50,10}$ of datasets.

1.3 Evaluation and hyperparameter selection

Empirical MSE Let $\text{ALG} \in \{\text{Z-ADJUST}, \text{MB-ADJUST}, \ldots\}$. For each $m \in [50]$ and $d \in [10]$, let $\widehat{P}^{(m,d)}(\text{ALG})$ denote the causal effect estimated by ALG when given dataset $\mathcal{D}^{(m,d)}$ as input. Then, for each $m \in [50]$, we compute the *empirical mean squared error* of ALG as follows:

$$\widehat{\mathsf{MSE}}_m(\text{ALG}) = \frac{1}{50} \sum_{m=1}^{50} \left(\widehat{P}^{(m,d)}(\text{ALG}) - P_x^{(m)}(y) \right)^2$$

Note that, for each instance $m = 1, 2, \ldots, 50$, the difficulty of the estimation problem may be different, i.e., $\widehat{\mathsf{MSE}}_m(\text{ALG})$ may depend as much on m as on ALG). This observation informs how we perform hyperparameter selection and report our final evaluation metric.

Hyperparameter selection To fairly select the best hyperparameters for each algorithm, we perform a simple grid search on a set of "hyperparameter selection" datasets $\{\mathcal{D}^{(m',d)}\}_{m'=51,d=1}^{100,10}$ that are only used for this step, and not for our final evaluation.[1]

Fix an algorithm ALG, and let ALG_η denote the algorithm run with hyperparameters η. For each η in the grid and for each $m' = 51, 52, \ldots, 100$, we

[1]This procedure ensures a good *upper bound* on the best possible performance of each algorithm, and reflects real-world scenarios in which either (i) the practitioner has domain knowledge about which hyperparameters are likely to perform well, or (ii) the practitioner uses an effective method for hyperparameter selection. In general data-driven hyperparameter selection is an interesting and complicated problem for *all* of these approaches, not just AMBA and BAMBA.

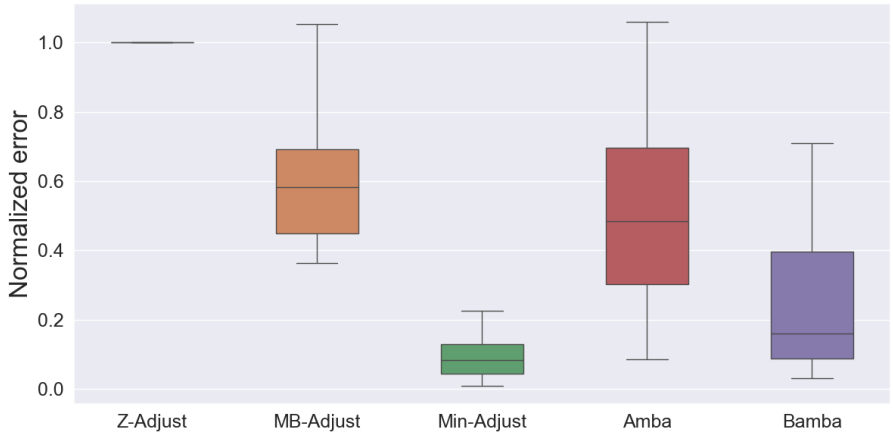

Figure 1: BAMBA **performs the best of any non-oracle algorithm on our synthetic evaluation.** We compared AMBA and BAMBA to four other approaches, including two approaches with additional oracle information (MB-ADJUST and MIN-ADJUST). The middle denotes the median, the lower and upper ends of the box denote the 25% and 75% quantiles, and the whiskers denote the range of the rest of the distribution (except for outliers). See text for more details.

compute $\widehat{\mathsf{MSE}}_{m'}(\text{ALG}_\eta)$. Then, for each η in the grid, we compute the number of instances where that η performed better than all other values η' in the grid, and we select the η with the largest fraction of "best performances".

The possible hyperparameters for our algorithms are the smoothing coefficient β (for (1) and (2)) and the conditional independence threshold τ. We search over possible values $\beta \in \{0.1, 0.25, 0.5, 1.5, 2\}$ and $\tau \in \{0.05, 0.075, 0.1, 0.2\}$. The selected hyperparameters for all algorithms are given in Table 1.

Final evaluation metric Using the best hyperparameter values, we normalize the MSEs of each algorithm to account for the difficulty of the instance. In particular, we define

$$\widehat{\mathsf{N\text{-}MSE}}_m(\text{ALG}) = \frac{\widehat{\mathsf{MSE}}_m(\text{ALG})}{\widehat{\mathsf{MSE}}_m(\text{BASELINE})},$$

where we select BASELINE = Z-ADJUST. In particular, for all $m = 1, \ldots, 50$, we always have $\widehat{\mathsf{N\text{-}MSE}}_m(\text{Z-ADJUST}) = 1$.

1.4 Results

In Figure 1, we demonstrate the performance of our algorithms and the baselines on our synthetic datasets. The performance levels match our theoretical

predictions: AMBA has lower error that Z-ADJUST, since it typically uses a smaller adjustment set; similarly, BAMBA has lower error than AMBA.

References

Sudeep Kamath, Alon Orlitsky, Dheeraj Pichapati, and Ananda Theertha Suresh. On learning distributions from their samples. In *Conference on Learning Theory*, pages 1066–1100. PMLR, 2015.

Ezequiel Smucler and Andrea Rotnitzky. A note on efficient minimum cost adjustment sets in causal graphical models. *Journal of Causal Inference*, 10 (1):174–189, 2022.