Supplementary Material for "FedECE: Federated Estimation of Causal Effect based on Causal Graphical Modelling"

Yongsheng Zhao, Kui Yu*, Guodu Xiang, Xianjie Guo, and Fuyuan Cao

I. The pseudo-codes of FedECE-B, FedECE-L and FedECE-O

Algorithm 1 gives the pseudo-codes of the FedECE-B algorithm, where FedECE-B consists of two main modules: a federated global causal structure learning module (Lines 1-18) and a federated global causal effect computation module (Lines 19-30). Among them, federated causal structure learning includes two submodules: a federated global skeleton learning submodule (Lines 1-13) and a federated skeleton orientation submodule (Lines 14-18).

Specifically, in the construction of the federated global skeleton, at each client, called Client cn, FedECE-B uses the PCstable algorithm to independently learn the global skeleton at the ℓ-layer and obtains the potential skeleton $\mathcal{G}^{\ell}_{cn}$ of all variables (Line 6). It is important to note that the learned potential skeletons may be different for different clients. To address this issue, at Line 10, a layer-wise cooperative optimization (LCO) strategy is employed to determine an optimal skeleton at each layer by aggregating all skeletons learned by all clients at the server, which then sends the optimal skeleton $\mathcal{G}^{\ell}$ to all clients as an initial skeleton for skeleton learning at the next layer. The federated skeleton learning phase continues until the value of ℓ is greater than the maximum number of direct neighbors of the variables in the ℓ-th skeleton learned by all clients. We record the final skeleton as $\mathcal{G}^{*}$.

In the federated skeleton orientation, a DOC mechanism is employed for federated V-structure identification based on the learned global skeleton $\mathcal{G}^{*}$ (Line 14). For an unshielded triple $\langle X_i, X_k, X_j\rangle$, based on the identified optimal separation set $\boldsymbol{SepSet}(X_i, X_j)$, if $X_k \notin \boldsymbol{SepSet}(X_i, X_j)$ holds, then $X_i - X_k - X_j$ oriented as $X_i \rightarrow X_k \leftarrow X_j$ (Lines 15-17). Then for the remaining undirected edges, the Meek's rules is applied on the server to orient edges as many as possible, resulting in a CPDAG $\hat{\mathcal{G}}$.

In the federated global causal effect computation, due to the existence of undirected edges in the learned CPDAG $\hat{\mathcal{G}}$, the causal effect output is often a multiset. We design the PIM strategy to address the multiset problem in causal effect computation within a federated setting. FedECE-B first exhausts all the valid DAGs existing in the learned CPDAG at the server, i.e., $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_K$ (Line 19). Then, each $\mathcal{D}_k$ ($k \in \{1, 2, \ldots, K\}$) is sent to all clients to obtain the value of the causal effect of X_i on X_j, denoted as θ^{k}_{cn}, using the backdoor criterion (Lines 22-24). Due to the quality of the datasets, different causal effect values may be obtained by different clients. To solve this problem, the server adopts an aggregation strategy to determine the causal effect value for each DAG, i.e., $\theta^k = \frac{1}{CN}\sum_{cn=1}^{CN}\theta^{k}_{cn}$. The federated causal effect value for this round of DAGs is computed and θ^k is added to the multiset θ, until all valid DAGs have been traversed.

In Algorithm 1, we find that the key to computing the effect lies in determining the parent set of X_i. Therefore, instead of exhaustively enumerating the complete DAG from the equivalence class, it is only necessary to locally identify the possible parent set $\boldsymbol{posspa}(X_i)$ of X_i in the learned CPDAG for computing causal effects. We propose an efficient algorithm, called FedECE-L.

Since the existence of undirected edges in CPDAG, when Algorithm 2 performs $\boldsymbol{posspa}(X_i)$ = $\{\boldsymbol{posspa}_1$, $\boldsymbol{posspa}_2$,..., $\boldsymbol{posspa}_K\}$ at the server based on the learned CPDAG $\hat{\mathcal{G}}$, it has to perform the local validity judgment of the parent set first, i.e., if X_k which is connected to X_i through an undirected edge is identified as a valid parent set, it must be ensured no V-structure containing X_i as a collider. Then $\boldsymbol{posspa}_k$ is sent to each client for causal effect computation. Then the server averages over the computed causal effects sent by all clients and obtains θ^k corresponding to the parent set $\boldsymbol{posspa}_k$. Then k is set to $k+1$, and the server continues to send the valid parent set to all clients, until the set $\boldsymbol{posspa}(X_i)$ is traversed and the multiset θ_L of the causal effect of X_i on X_i is obtained.

Since the valid adjustment set is not unique, different valid adjustment sets usually provide different causal effect estimations. In Algorithm 3 of FedECE-O, we introduce the O-set instead of the parent set in Algorithm 2 as the valid adjustment set for accurate estimation of causal effects. Note that another difference between Algorithm 2 and Algorithm 3 lies in the fact that Algorithm 2 only checks whether $X_j \notin \boldsymbol{posspa}(X_i)$ holds, while Algorithm 3 checks further a strong condition $X_j \in \boldsymbol{possde}(X_i)$. These two conditions ensure that the adjustment set is valid.

This work was supported by the National Science and Technology Major Project of China (2021ZD0111801) and the National Natural Science Foundation of China (under Grants 62376087 and 62176082). Corresponding author: Kui Yu.

Yongsheng Zhao, Kui Yu, Guodu Xiang, and Xianjie Guo are with with the School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230601, China (e-mail: yszhao@mail.hfut.edu.cn, yukui@hfut.edu.cn, xgd600600@mail.hfut.edu.cn, and xianjieguo@mail.hfut.edu.cn).

Fuyuan Cao is with the School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China (e-mail: cfy@sxu.edu.cn).

Algorithm 1: FedECE-B

INPUT: Dataset $\mathcal{D}(\mathcal{X})$ generated from a probability distribution faithful to a DAG $\mathcal{D}_{\text{true}}$, the number of clients CN and the significance level of the statistical test α
OUTPUT: the multisets θ of possible causal effects

```
// Phase 1: Federated causal structure learning
  // Step 1: Federated causal skeleton learning
Form complete undirected graph G^c on the variable set X
Let depth ℓ = 0
repeat
    when ℓ = 0, G^{ℓ-1} = G^c
    for Client cn ∈ {1, 2, ..., CN} do
        Use the local dataset to update the skeleton G^{ℓ-1} to get G^ℓ_cn
    end for
    Send the independently learned skeleton G^ℓ_cn at the ℓ-th layer at each client to the server
    At the server, do the following steps:
    - Aggregate the skeletons sent by the clients to get the skeleton of the ℓ-th layer G^ℓ
    - Send the aggregated skeleton G^ℓ to each client as the initial skeleton for skeleton learning at the (ℓ+1)-th layer
    ℓ = ℓ + 1
until the maximum number of neighbors of a variable learned by all clients at the ℓ-th layer < ℓ
  // Step 2: Federated causal skeleton Orientation
Adopt DOC mechanism to get SepSet_{i,j} of the unshielded triple < X_i, X_k, X_j >
if X_k ∉ SepSet_{ij} then
    Orient < X_i, X_k, X_j > as X_i → X_k ← X_j
end if
Use Meek's rules to orient as many of the remaining undirected edges as possible to obtain CPDAG Ĝ
// Phase 2: Federated casual effect calculation
At the server, determine all DAGs D_1, D_2, ..., D_K in the Ĝ, then send D_k to each client
Let k = 1
repeat
    for Client cn ∈ {1, 2, ..., CN} do
        Use local dataset to compute causal effect of X_i on X_j as θ^k_cn, i.e. θ^k_cn = γ_{x_i|pa(D_k)}
    end for
    Send the θ^k_cn independently calculated by each client to the server simultaneously
    At the server, do the following steps:
    - θ^k = (1/CN) Σ_{cn=1}^{CN} θ^k_cn
    - Add θ^k to θ
    k = k + 1
until the set DAGs is traversed
```

TABLE I: Comparison of the MAE values of FedECE-B, FedECE-L and FedECE-O with the ten baseline methods on the synthetic dataset of 10 variables with $CN \in \{5, 8, 10, 15\}$.

Method	$CN=5$	$CN=8$	$CN=10$	$CN=15$
IDA-Avg^L	0.0882	0.1000	0.1084	0.1245
IDA-Best^L	0.0716	0.0871	0.0881	0.0936
IDA-Avg^O	0.0954	0.1118	0.1206	0.1408
IDA-Best^O	0.0767	0.0924	0.0976	0.1054
FedECE$^L_{min}$	0.0788	0.1060	0.1209	0.1529
FedECE$^L_{max}$	0.0764	0.1088	0.1238	0.1568
FedECE$^L_{vote}$	0.0707	0.0861	0.0880	0.0893
FedECE$^O_{min}$	0.0760	0.0984	0.1087	0.1341
FedECE$^O_{max}$	0.0745	0.1011	0.1118	0.1384
FedECE$^O_{vote}$	0.0625	0.0630	0.0701	0.0806
FedECE-B	**0.0474**	**0.0552**	**0.0540**	**0.0582**
FedECE-L	**0.0474**	**0.0552**	**0.0540**	**0.0582**
FedECE-O	0.0540	0.0625	0.0624	0.0724

II. Additional experimental results

A. Experiment results on four synthetic datasets

In this section, we present the full experimental results on the four synthetic datasets. The synthetic datasets are generated based on the following parameter settings: each dataset consists of 5000 samples to ensure reliable statistical estimations. Random DAGs are generated using the Erdos-Renyi model [1] with an expected number of edges per variable of EN = 2, ensuring moderately sparse structures. The weights of causal edges in the DAGs are randomly sampled from the range $[-1, -0.5] \cup [0.5, 1]$, ensuring all edges have significant weights. Gaussian noise with a mean of 0 and a standard deviation which is dynamically determined by the covariance matrix derived from the random DAG structure is added to each variable to simulate realistic data variability. Table I to IV show the MAE values of FedECE-B, FedECE-L and FedECE-O and their rivals using four synthetic datasets, respectively.

Generally, we can see that FedECE-B, FedECE-L and FedECE-O achieve lower MAE values than their competitors, indicating the superiority of our methods. This is due to the following reasons: the superior performance of FedECE relies on accurately learned causal structures, and the proposed Fed-CSL module constructs a more accurate CPDAG. Additionally, the PIM strategy makes full use of the local datasets of each client to identify the valid adjustment set for federated causal effect calculation, resulting in more accurate causal effect values.

The MAE values computed by IDA-Avg^L and IDA-Best^L are higher than those computed by FedECE-L and FedECE-O, indicating that FedECE-L and FedECE-O achieve a more accurate multiset of causal effects. This is likely because IDA-

TABLE II: Comparison of the MAE values of FedECE-B, FedECE-L and FedECE-O with the ten baseline methods on the synthetic dataset of 20 variables with $CN \in \{5, 8, 10, 15\}$. A value NA means that the calculation took more than 48 hours, so the calculation was terminated.

Method	$CN = 5$	$CN = 8$	$CN = 10$	$CN = 15$
IDA-Avg^L	0.0855	0.1043	0.1098	0.1303
IDA-Best^L	0.0747	0.0862	0.0990	0.1038
IDA-Avg^O	0.0864	0.1077	0.1147	0.1383
IDA-Best^O	0.0762	0.0873	0.0970	0.1067
FedECE$^L_{min}$	0.0708	0.1002	0.1175	0.1535
FedECE$^L_{max}$	0.0704	0.0980	0.1148	0.1507
FedECE$^L_{vote}$	0.0666	0.0831	0.0790	0.0863
FedECE$^O_{min}$	0.0572	0.0793	0.0921	0.1221
FedECE$^O_{max}$	0.0583	0.0809	0.0954	0.1242
FedECE$^O_{vote}$	0.0662	0.0827	0.0877	0.1028
FedECE-B	NA	NA	NA	NA
FedECE-L	0.0382	0.0418	**0.0448**	**0.0487**
FedECE-O	**0.0357**	**0.0405**	0.0462	0.0554

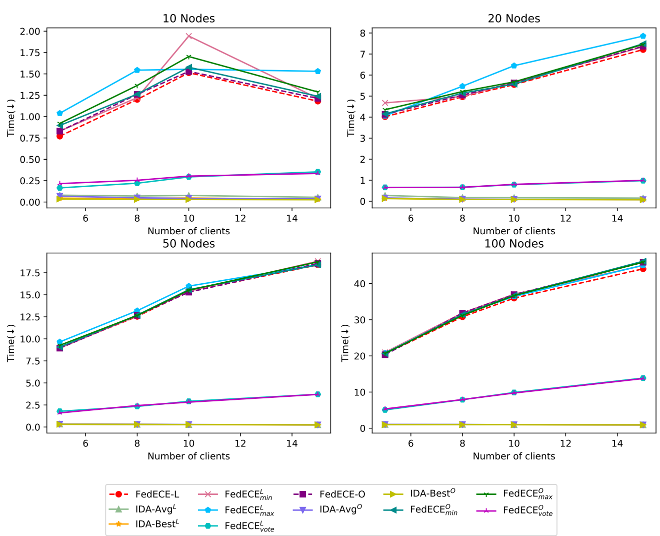

Fig. 1: Runtime on 4 synthetic datasets.

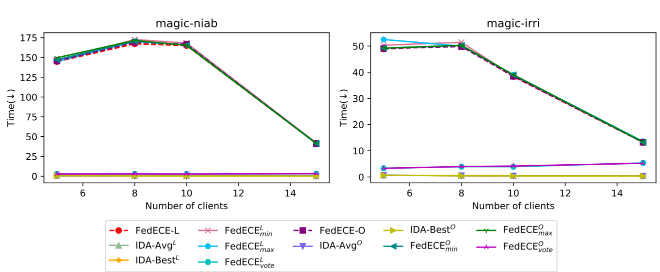

Fig. 2: Runtime on 2 BN datasets.

Algorithm 2: FedECE-L

Output: Dataset $\mathcal{D}(\mathcal{X})$ generated from a probability distribution faithful to a DAG $\mathcal{D}_{\text{true}}$, the number of clients CN and the significance level of the statistical test α

Input: the multisets θ_L of possible causal effects

// Phase 1: Federated causal structure learning

// Phase 2: Federated causal effect calculation

At the server, do the following steps:
- ***ne***$(\hat{\mathcal{G}}, X_i) \leftarrow \{X_k \in \mathcal{X} \setminus X_i : X_i - X_k$ in $\hat{\mathcal{G}}\}$

for each subset SS of ***ne***$(\hat{\mathcal{G}}, X_i)$ **do**
  if $\hat{\mathcal{G}}_{SS}$ is locally valid (i.e., has no new V-structure with collider X_i) **then**
    Add $SS \cup$ ***pa***(X_i) to ***posspa***(X_i)
  end if
end for
Send the ***posspa***$_k \in$ ***posspa***(X_i) to each client
Let $k = 1$
repeat
  for Client $cn \in \{1, 2, ..., CN\}$ **do**
    if $X_j \notin$ ***posspa***$_k$ **then**
      $\theta^k_{cn} = \gamma_{x_i|\textit{\textbf{posspa}}_k}$
    else
      $\theta^k_{cn} = 0$
    end if
  end for
  Send the θ^k_{cn} independently calculated by each client to the server simultaneously
  At the server, do the following steps:
  - $\theta^k = \frac{1}{CN}\sum_{cn=1}^{CN} \theta^k_{cn}$
  - Add θ^k to θ_L
  $k = k + 1$
until the set ***posspa***(X_i) is traversed

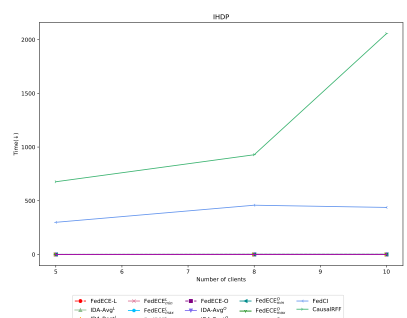

Fig. 3: Runtime on IHDP dataset.

Avg^L and IDA-Best^L do not exchange information between clients, whereas FedECE-L leverages information exchange between clients for both federated structure learning and federated causal effect computation. This further validates the effectiveness of FedECE-L and FedECE-O.

In summary, on all four synthetic datasets, FedECE-L and FedECE-O significantly outperform all of their rivals. FedECE-O outperforms FedECE-L when the number of clients CN = 5 and 8. However, it is inferior to FedECE-L in all cases where CN = 10 and 15. This may be attributed to the fact

Algorithm 3: FedECE-O

Output: Dataset $\mathcal{D}(\mathcal{X})$ generated from a probability distribution faithful to a DAG $\mathcal{D}_{\text{true}}$, the number of clients CN and the significance level of the statistical test α
Input: the multisets θ_O of possible causal effects

```
// Phase 1: Federated causal structure learning
// Phase 2: Federated causal effect calculation
At the server, the optimal adjustment set
  O(X_i, X_j) = {O_1, O_2, ..., O_K} is learned based on Ĝ.
  Then send O_k to each client
Let k = 1
repeat
    for Client cn ∈ {1, 2, ..., CN} do
        if X_j ∈ possde(X_i) then
            θ^k_cn = γ_{x_i|O_k}
        else
            θ^k_cn = 0
        end if
    end for
    Send the θ^k_cn independently calculated by each client
      to the server simultaneously
    At the server, do the following steps:
    - θ^k = (1/CN) Σ_{cn=1}^{CN} θ^k_cn
    - Add θ^k to θ_O
    k = k + 1
until the set O(X_i, X_j) is traversed
```

TABLE III: Comparison of the MAE values of FedECE-B, FedECE-L and FedECE-O with the ten baseline methods on the synthetic dataset of 50 variables with $CN \in \{5, 8, 10, 15\}$. A value NA means that the calculation took more than 48 hours, so the calculation was terminated.

Method	$CN = 5$	$CN = 8$	$CN = 10$	$CN = 15$
IDA-Avg^L	0.0918	0.1049	0.1148	0.1317
IDA-Best^L	0.0671	0.0847	0.0871	0.0897
IDA-Avg^O	0.0927	0.1082	0.1186	0.1407
IDA-Best^O	0.0653	0.0866	0.0878	0.0898
FedECE$^L_{min}$	0.0719	0.0989	0.1155	0.1651
FedECE$^L_{max}$	0.0655	0.0927	0.1087	0.1544
FedECE$^L_{vote}$	0.0635	0.0816	0.0865	0.0895
FedECE$^O_{min}$	0.0546	0.0747	0.0861	0.1213
FedECE$^O_{max}$	0.0518	0.0731	0.0843	0.1190
FedECE$^O_{vote}$	0.0680	0.0708	0.0875	0.1007
FedECE-B	NA	NA	NA	NA
FedECE-L	0.0316	0.0335	**0.0332**	**0.0406**
FedECE-O	**0.0293**	**0.0327**	0.0353	0.0471

that an increase in the number of clients and a decrease in the amount of data allocated to each client, leads to an inaccurate CPDAG, which in turn results in an inaccurate adjustment set.

B. Time Efficiency

Fig. 1 to 3 present the execution times of FedECE-L and FedECE-O, along with their competitors, on the four synthetic datasets, two BN datasets and one IHDP dataset (due to the small scale of the DREAM4 dataset, its execution times are trivial and thus not reported). For most datasets, FedECE-L is slower than IDA-Avg, IDA-Best, and FedECE$^L_{vote}$, but comparable to FedECE$^L_{min}$ and FedECE$^L_{max}$. This is because FedECE-L requires additional time for communication between clients and the server during skeleton learning, finding separation sets at each client, and aggregating causal effect values computed at each client to obtain a consistent multiset. As the number of clients increases, the running time of most algorithms also increases. In summary, FedECE-L is generally competitive with FedECE$^L_{min}$ and FedECE$^L_{max}$. The comparison of FedECE-O with its competing algorithms is similar to that of FedECE-L. Notably, both FedECE-L and FedECE-O are significantly faster than FedCI and CausalRFF on the IHDP dataset.

C. Stability Analysis of Experimental Results

To verify the stability of our proposed methods, we conduct extensive experiments using synthetic datasets under identical parameter settings. Specifically, for each network, we randomly generate 5 datasets, each with a sample size of 5000, to evaluate the stability of the results based on multiple datasets generated from the same network. We employ the Hausdorff distance metric to measure the distance between the causal effect estimated set $\hat{\theta}$, computed by the proposed algorithms, and the true causal effect set θ^*.

The stability analysis examines scenarios with varying numbers of network clients and variables, where for each network configuration, 5 datasets are generated using different random seeds to evaluate consistency. The mean performances and variances of the Hausdorff distance for FedECE-L and FedECE-O under these scenarios are presented in Table V and Table VI, respectively. Due to the constraints of the global estimator, FedECE-B is unsuitable for datasets with more than 15 variables and thus its stability analysis is excluded. The calculation formulas of the mean and variance are shown in the Eq. (1) and Eq. (2), where N represents the number of experiments.

$$Mean = \frac{1}{N}\sum_{i=1}^{N} H(\hat{\theta}, \theta^*) \tag{1}$$

$$Variance = \frac{1}{N}\sum_{i=1}^{N} (H(\hat{\theta}, \theta^*) - Mean)^2 \tag{2}$$

As shown in Table V and Table VI, the variance is particularly low for smaller networks, indicating the efficiency and stability of the algorithm when dealing with simpler problem Settings. As the number of variables increases, a

TABLE IV: Comparison of the MAE values of FedECE-B, FedECE-L and FedECE-O with the ten baseline methods on the synthetic dataset of 100 variables with $CN \in \{5, 8, 10, 15\}$. A value NA means that the calculation took more than 48 hours, so the calculation was terminated.

Method	$CN = 5$	$CN = 8$	$CN = 10$	$CN = 15$
IDA-Avg^L	0.0955	0.1014	0.1064	0.1182
IDA-Best^L	0.0685	0.0853	0.0936	0.1057
IDA-Avg^O	0.1021	0.1066	0.1132	0.0495
IDA-Best^O	0.0684	0.0882	0.0989	0.0439
FedECE$^L_{min}$	0.0852	0.0957	0.1127	0.1501
FedECE$^L_{max}$	0.0852	0.0954	0.1120	0.1473
FedECE$^L_{vote}$	0.0684	0.0754	0.0693	0.0786
FedECE$^O_{min}$	0.0626	0.0695	0.0834	0.1093
FedECE$^O_{max}$	0.0614	0.0699	0.0834	0.1096
FedECE$^O_{vote}$	0.0587	0.0558	0.0771	0.0949
FedECE-B	NA	NA	NA	NA
FedECE-L	0.0337	0.0286	**0.0304**	**0.0321**
FedECE-O	**0.0296**	**0.0272**	0.0320	0.0398

TABLE V: Mean performances and variances of FedECE-L under different networks and clients.

Variables	Clients	Mean± Variance
10	5	0.018864±0.000104
	8	0.015104±0.000102
	10	0.020512±0.000086
	15	0.021206±0.000058
20	5	0.007358±0.000009
	8	0.009922±0.000080
	10	0.006544±0.000020
	15	0.006322±0.000020
50	5	0.012498±0.000039
	8	0.015308±0.000025
	10	0.016380±0.000080
	15	0.015588±0.000102
100	5	0.030874±0.000038
	8	0.021288±0.000052
	10	0.021698±0.000056
	15	0.020088±0.000072

slight increase in variance is observed. This increase is due to the increasing complexity of the problem space as the dimension increases. However, the variance remains within a small range, which indicates that the proposed method is robust and adaptable even in large-scale network environments.

For smaller networks with fewer variables, the variance is very low regardless of the number of clients. In large networks with more variables, the variance increases slightly as the number of clients increases. This trend is evident in both algorithms, especially for configurations with 15 clients.

TABLE VI: Mean performances and variances of FedECE-O under different networks and clients.

Variables	Clients	Mean± Variance
10	5	0.014290±0.000150
	8	0.011826±0.000111
	10	0.014480±0.000099
	15	0.010578±0.000115
20	5	0.008130±0.000023
	8	0.007378±0.000087
	10	0.007824±0.000016
	15	0.006650±0.000007
50	5	0.012508±0.000039
	8	0.015268±0.000025
	10	0.016246±0.000084
	15	0.014702±0.000107
100	5	0.013688±0.000136
	8	0.013778±0.000073
	10	0.010924±0.000052
	15	0.011850±0.000076

However, the overall variance remains small, indicating that both algorithms are robust to an increase in the number of clients even in complex scenarios.

REFERENCES

[1] P. ERDdS and A. R&wi, "On random graphs i," *Publ. math. debrecen*, vol. 6, no. 18, pp. 290–297, 1959.