Directed Acyclic Graph (DAG)-Informed Structure Learning from Multi-Dimensional Point Processes

Chunming Zhang[1] (*czhang3@wisc.edu*) Muhong Gao[2] Shengji Jia[3]

[1]University of Wisconsin-Madison School of Computer, Data & Information Sciences UNIVERSITY OF WISCONSIN-MADISON Data Science Institute Powered by American Family Insurance [2]Chinese Academy of Sciences [3]Shanghai Lixin University of Accounting and Finance

1. Introduction: Multivariate Temporal Point Process

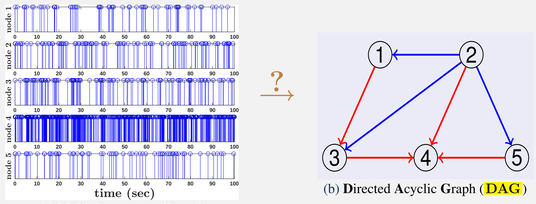
■ **Multivariate *Temporal Point Process*:**

- Random occurrences of a specific type of event (e.g., neuron spike firing, contagious disease incidence, e-commerce transactions, or social media postings) over time, represented as $\{\boldsymbol{T}_1, \ldots, \boldsymbol{T}_d\}$, recorded at d nodes.
- At each node j in the node set $\mathcal{V} = \{1, \ldots, d\}$,

$$\boldsymbol{T}_j = (T_{j,1}, \ldots, T_{j,\mathrm{N}_j}) \quad \text{with } 0 < T_{j,1} < \cdots < T_{j,\mathrm{N}_j} \le \mathrm{T}, \tag{1}$$

signifies a sequence of time points $T_{j,\ell}$, representing the occurrence time of the ℓth event during an experiment of duration T.

■ **Key objective:** Learn the dependency structure among nodes and represent it as a **DAG** from the d sequences of event times.



(a) Multiple neuron spike trains

(b) Directed Acyclic Graph (DAG)

Figure: **[Multiple Neuron Spike Train Data]** *Each node of the network graph in panel (b) corresponds to a point process in panel (a). Arrows indicate interactions (red: excitatory; blue: inhibitory).*

2. Generative Modeling of CIFs for Causal Graphs

■ **Our continuous-time model for the *conditional intensity function*:**

$$\lambda_j(t \mid \mathscr{F}_t) = \exp\big(w_{0,j} + \textstyle\sum_{i=1}^{d} w_{i,j}\, x_i(t)\big) \tag{2}$$

■ **Negative log-likelihood function for point process data $\boldsymbol{T}_j$:**

$$\mathcal{L}_{j,\mathrm{T}}(\boldsymbol{w}_{\cdot,j}) = -\mathrm{T}^{-1} \int_0^{\mathrm{T}} \big[\boldsymbol{w}_{\cdot,j}^\top \boldsymbol{x}(t-)\, \mathrm{d}N_j(t) - \exp\{\boldsymbol{w}_{\cdot,j}^\top \boldsymbol{x}(t)\}\, \mathrm{d}t\big]. \tag{3}$$

■ **Proposed 'DAG-Constrained Regularized Estimator':**

$$\widehat{\mathbf{W}} = \arg\min_{\mathbf{W} \in \mathbb{R}^{(d+1)\times d}} \mathcal{L}(\mathbf{W}) + \mathcal{P}(\mathbf{W}; \boldsymbol{\eta}), \quad \text{subject to} \quad \mathcal{G}(\mathbf{W}) \text{ being a } \mathbf{DAG}, \tag{4}$$

where $\mathbf{W} = (w_{i,j}) \in \mathbb{R}^{d\times d}$ is the **weighted adjacency matrix**,

$$\mathcal{L}(\mathbf{W}) = \textstyle\sum_{j=1}^{d} \mathcal{L}_{j,\mathrm{T}}(\boldsymbol{w}_{\cdot,j}), \tag{5}$$

$$\mathcal{P}(\mathbf{W}; \boldsymbol{\eta}) = \textstyle\sum\sum_{1\le i\ne j\le d} \eta_{i,j}\, |w_{i,j}|, \quad \text{with } \eta_{i,j} \ge 0. \tag{6}$$

■ **Question:**

- How can we solve (4) under the combinatorial DAG-constraint?
- What are the statistical properties of the solution?

3. Optimization Algorithm for DAG-Constrained Estimator

■ **Characterizing a DAG** (Zheng *et al.*, 2018, *NeurIPS*):

$$\mathcal{G}(\mathbf{W}) \text{ is a } \mathbf{DAG} \iff h(\mathbf{W}) = 0,$$

where

$$h(\mathbf{W}) := \mathrm{trace}\{\exp(\mathbf{W} \circ \mathbf{W})\} - d. \tag{7}$$

■ **Our (4) becomes a continuous equality-constrained program:**

$$\min_{\mathbf{W} \in \mathbb{R}^{(d+1)\times d}} \mathcal{L}(\mathbf{W}) + \mathcal{P}(\mathbf{W}; \boldsymbol{\eta}), \quad \text{subject to } h(\mathbf{W}) = 0. \tag{8}$$

■ **The augmented Lagrangian (AL) function for problem (8):**

$$L(\mathbf{W}, \alpha; \rho) = \mathcal{L}(\mathbf{W}) + \mathcal{P}(\mathbf{W}; \boldsymbol{\eta}) + \alpha\, h(\mathbf{W}) + 2^{-1}\rho\, h^2(\mathbf{W}), \tag{9}$$

where $\alpha \in \mathbb{R}$ is the Lagrange multiplier, and $\rho > 0$ is the augmentation parameter.

■ **The classical AL method is a dual ascent algorithm**: For a fixed ρ,

- Given α, solve

$$\mathbf{W}_\alpha = \arg\min_{\mathbf{W} \in \mathbb{R}^{(d+1)\times d}} L(\mathbf{W}, \alpha; \rho). \tag{10}$$

- Define the dual function as the Lagrange multiplier α,

$$D(\alpha) = L(\mathbf{W}_\alpha, \alpha; \rho), \tag{11}$$

and solve

$$\bar{\alpha} = \arg\max_{\alpha \in \mathbb{R}} D(\alpha). \tag{12}$$

■ **The classical AL algorithm for updating $\mathbf{W}$, α, and ρ:**

- **Update $\mathbf{W}$**, by solving the unconstrained subproblem:

$$\mathbf{W}^{(k+1)} = \arg\min_{\mathbf{W} \in \mathbb{R}^{(d+1)\times d}} L(\mathbf{W}, \alpha^{(k)}; \rho^{(k)}), \tag{13}$$

using the previous solution $\mathbf{W}^{(k)}$ as the initial value.

- **Update α and ρ** according to the rules:

$$\alpha^{(k+1)} = \alpha^{(k)} + \beta_\alpha\, h(\mathbf{W}^{(k+1)}), \tag{14}$$

$$\rho^{(k+1)} = \beta_\rho\, \rho^{(k)}, \tag{15}$$

for appropriately chosen step sizes $\beta_\alpha > 0$ and $\beta_\rho \ge 1$.

■ **Limitation of classical AL method:** Slow convergence and high computational cost.

- Computing $h(\mathbf{W})$ in (14) is particularly expensive.
- Solving (13) is computationally intensive.

■ **Proposed Flex-AL algorithm with flexible updates of α and ρ:**

$$\max\{\alpha^{(k)}, \rho^{(k)}\} \to \infty \text{ as } k \to \infty, \quad \text{and } \inf_{k\ge 0} \alpha^{(k)} > -\infty, \quad \inf_{k\ge 0} \rho^{(k)} > 0. \tag{16}$$

■ **Example update:** $\alpha^{(k+1)} = \gamma_\alpha\, \alpha^{(k)}$ and $\rho^{(k+1)} = \gamma_\rho\, \rho^{(k)}$, where $\gamma_\alpha > 1$ and $\gamma_\rho > 1$, starting from $\alpha^{(0)} > 0$ and $\rho^{(0)} > 0$.

Global convergence of Flex-AL algorithm for solving (8): *Assume that the sequence of Lagrange multipliers $\{\alpha^{(k)}\}_{k\ge 0}$ and the sequence of augmentation parameters $\{\rho^{(k)}\}_{k\ge 0}$ satisfy condition* (16)*. Then:*

- *Any limit point of the sequence of global minimizers $\{\mathbf{W}^{(k)}\}_{k\ge 1}$ in* (13) *is a global minimizer of* (8)*.*
- *Moreover, assume condition* A6′ *(the minimum eigenvalue of $\nabla^2 \mathcal{L}_{j,\mathrm{T}}(\boldsymbol{w}_{\cdot,j})$ is bounded away from 0). Then $\{\mathbf{W}^{(k)}\}_{k\ge 1}$ has at least one limit point. Also, if* (8) *has a unique global minimizer $\mathbf{W}$, then $\mathbf{W}^{(k)} \to \mathbf{W}$ as $k \to \infty$.*

4. Statistical Properties of Structure Learning Methods

■ **Expanded estimators:**

$$\widehat{\mathbf{W}}_{\boldsymbol{\kappa},\boldsymbol{\eta}} = \arg\min_{\mathbf{W} \in \mathbb{R}^{(d+1)\times d};\, h(\mathbf{W}) \le \boldsymbol{\kappa}} \mathcal{L}(\mathbf{W}) + \mathcal{P}(\mathbf{W}; \boldsymbol{\eta}). \tag{17}$$

- **Scenario (a): Unregularized Estimation ($\boldsymbol{\eta} = 0$ in (17))**
  - **Case 1: non-DAG-constrained ($\boldsymbol{\kappa} = \infty$), unregularized ($\boldsymbol{\eta} = 0$):**
  - **Case 2: DAG-constrained ($\boldsymbol{\kappa} = 0$), unregularized ($\boldsymbol{\eta} = 0$):**
- **Scenario (b): Regularized Estimation ($\boldsymbol{\eta} \ne 0$ in (17))**
  - **Case 3: non-DAG-constrained ($\boldsymbol{\kappa} = \infty$), regularized ($\boldsymbol{\eta} \ne 0$):**
  - **Case 4: DAG-constrained ($\boldsymbol{\kappa} = 0$), regularized ($\boldsymbol{\eta} \ne 0$):**

■ **Takeaways:**

- Without **sparsity**-regularization, incorporating the **DAG constraint** is **necessary** for DAG-recovery consistency.
- With an appropriate sparsity-regularization **penalty**, the **DAG-constrained** estimator achieves the same parameter-estimation and DAG-recovery consistencies as the **non-DAG-constrained** estimator, while **improving empirical performance**.

5. Numerical Experiments

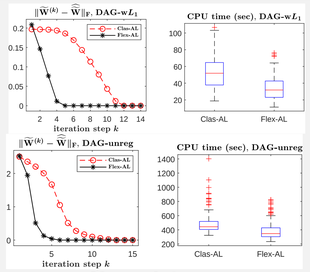


Figure: **(Simulation Study: Network 3)** *Left panel: average approximation error $\|\mathbf{W}^{(k)} - \widehat{\mathbf{W}}\|_{\mathrm{F}}$ over 100 replicates versus the iteration step k; right panel: boxplots of runtime for estimators $\widehat{\mathbf{W}}$ across replications. Top row: 'DAG-wL_1'; bottom row: 'DAG-unreg'.* $\mathrm{T} = 1200$.

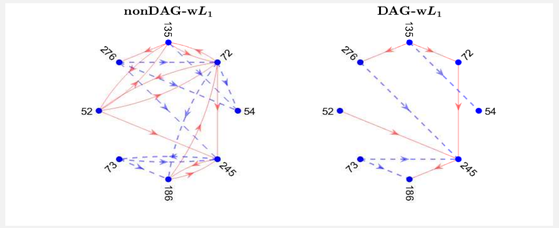


Figure: **(Real spike train data: estimated sub-networks)** *Comparison of estimated sub-networks using two methods. Red solid lines with arrows: excitatory effects; blue dashed lines with arrows: inhibitory effects.*

6. Conclusion and Discussion

- **Flex-AL**: faster DAG-constrained optimization with a global convergence guarantee.
- New insights into **DAG constraints** and **sparsity** regularization.

References:

- Gao, Zhang, & Zhou (2024). *IEEE Trans. Inf. Theory.*
- Zhang, Gao, & Jia (2024). *JMLR.*