Fast Fourier Transform Reductions for Bayesian Network Inference

Vincent Hsiao[1], Dana Nau[1], Rina Dechter[2]

DEPARTMENT OF COMPUTER SCIENCE

University of Maryland, College Park[1], University of California, Irvine[2]

Background

Bayesian Network (BN)

- Bayesian Network: graphical model (X,D,F)
  - Variables: $X = \{X_1, X_2, ..., X_N\}$
  - Domains: $D = \{D_{X_1}, D_{X_2}, \dots, D_{X_N}\}$
  - Parent Functions: $F = \{F_1, F_2, \dots, F_N\}$

Causal Independence (CI)

- Probabilistic relationship between a set of causes $\{c_1, ..., c_n\}$ and an effect e, such that:

  $$e = h_1 * h_2 ... * h_n$$

  where each hidden variable h_i is some probabilistic function of its corresponding c_i and * is some commutative and associative binary operator
  - ***Summation-type CI***: operator "*" is addition "+"

- **Network Transformation**: Given a CI BN fragment {X,D,F} with a set of causes $\{c_1, ..., c_n\}$, an effect e, and hidden variables $\{h_1, \dots, h_n\}$, a network transformation is a new network {X',D',F'} constructed over some computational ordering:

  $$e = (\dots(((h_1 * h_2) * h_3) * h_4) * \dots) * h_n$$

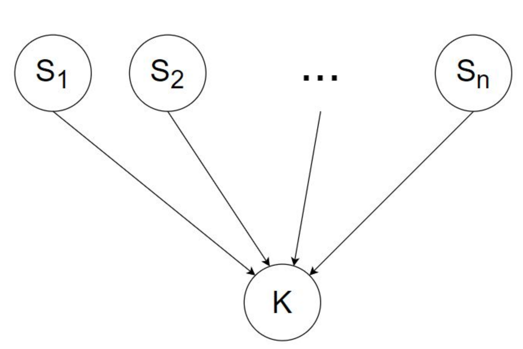
  with new intermediate variables:

  $$y_i : \{y_1 = h_1 * h_2, \; y_2 = y_1 * h_3 \dots\}$$

  - Algorithms such as ***ci-elim-bel*** (Zhang and Poole, 1996; Rish and Dechter, 1998) exploit network transformations to accelerate bucket elimination

Applications

- Distributed Computing, Fault Tree Analysis
  - N different resource providers with stochastic availability (k-out-of-n model)



k/n sum network (1)

- Evolutionary Game Theory (source-sink networks)

Problem Statement

Current Approach

- Network transformations (temporal decomposition), *ci-elim-bel*

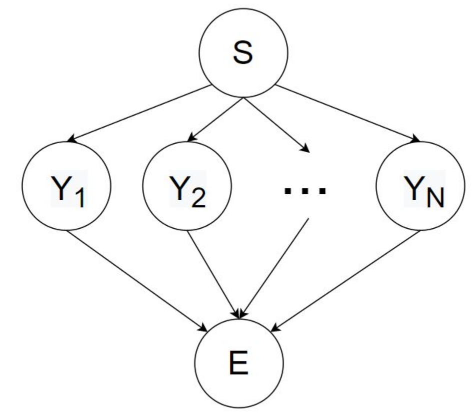
Proposed Approach

- Eliminate unneeded nodes in summation-type CI efficiently through the use of the FFT
- Integrate the FFT with bucket elimination into an efficient inference algorithm

FFT Reduction

Computing Random Variable Sums

Given a source-sink network:



source-sink network (2)

The size of the parent function for the node E expressed as a conditional probability table (CPT) is exponential in the number of Y nodes

We would like to reduce the size of the CPT of the network. This is useful in applications where one wants to find the marginal distribution of node E such as in:
- Test score prediction
- Evolutionary games

It can be shown that the distribution P(E|S) can be expressed as a convolution of individual distributions $P(Y_i|S)$

From the convolution theorem and the application of the FFT, we know that:

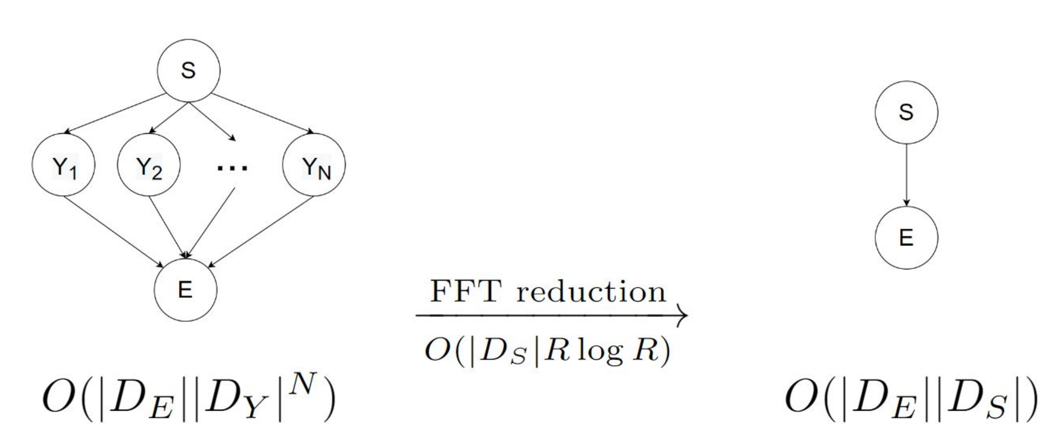
> **Theorem 2.4** (FFT Time Complexity). *For i.i.d random variables $X = \{X_1, X_2, \dots, X_N\}$ where each variable has a domain size $|D|$ and their sum $Z = \sum X_i$, computing $P(Z = k)$ using the FFT will take time $O(N|D|\log(N|D|))$. For non-i.i.d variables, the time complexity is $O(N^2|D|\log(N|D|))$.*

It follows that:

> **Theorem 3.1** (FFT Reduction). *Let $B = \{X, D, F\}$ with $X = \{S, Y_1, \dots, Y_N, E\}$ be a source-sink network with N i.i.d paths. The network can be transformed into $\{X', D', F'\}$ such that $X' = \{S, E\}$ reducing the CPT for E from size $O(|D_E||D_Y|^N)$ to size $O(|D_S||D_E|)$ in $O(|D_S|R\log R)$ time where R is the numerical range of random variable E.*

Furthermore, it can be shown that certain computations performed in the algorithm *ci-elim-bel* can also be formulated as convolutions, leading to an improved algorithm: *ci-elim-FFT*

FFT Reduction



CI-Elim-FFT

Algorithm 2: ci-elim-FFT
Input: A Bayesian network B, evidence e
Output: $P(x_1|e)$

Generate a decomposition network from B
Generate ordering $o = \{Z_1, \dots, Z_n\}$ with $Z_1 = \{x_1\}$ using B'
for $i = n \to 1$ **do** // create buckets
    $\forall x \in Z_i$, put all network functions with x as highest ordered variable in $bucket_i$
end
for $i = n \to 1$ **do** // process buckets
    // $h_1, \dots, h_m$ are functions in $bucket_i$
    if $Z_i = \{u_l; u_k\}, u = u_l + u_k$ **then**
        $h^{u_l} = \prod_{j, u_l \in h_j} h_j$;
        $h^{u_k} = \prod_{j, u_k \in h_j} h_j$;
        $h^{Z_i} = \mathcal{F}^{-1}\{\mathcal{F}\{h^{u_l}\} \cdot \mathcal{F}\{h^{u_k}\}\}$
    else
        use regular ci-elim-bel to compute h^{Z_i}
    Put h^{Z_i} in the highest bucket that mentions h^{Z_i}'s variable.
end
Return αh^{x_1}, (α is a normalizing constant).

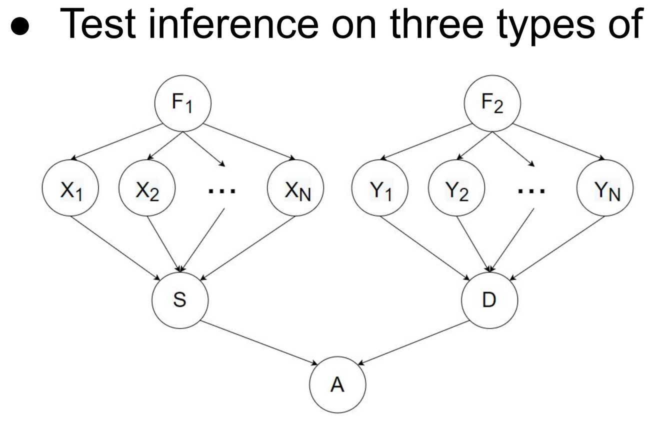
Complexity on k/n sum network (1)

	$i.i.d \; Y_i^\prime s$	non-$i.i.d \; Y_i^\prime s$
BE	$O(N\|D_X\|R_Z^2)$	$O(N\|D_X\|R_Z^2)$
FFT	$O(\|D_X\|R_Z \log R_Z)$	$O(N\|D_X\|R_Z \log R_Z)$

Experimental Setup

Experimental Setup (Set 1)

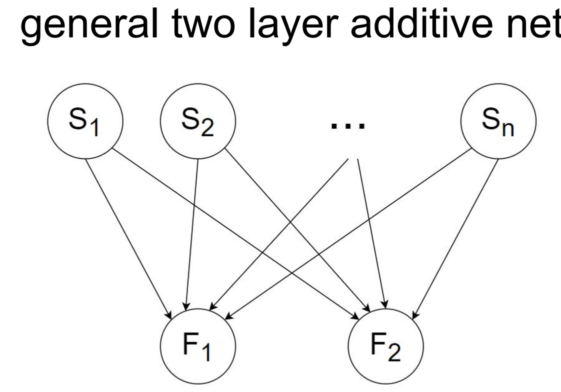
- Evaluate FFT reduction for inference on a selection of networks with summation-CI
- Compare with:
  - Vanilla Bucket Elimination (Naive)
  - Temporal Decomposition (Temporal)
- Test inference on three types of networks (1), (2), (3)



Supply demand network (3)

Experimental Setup (Set 2)

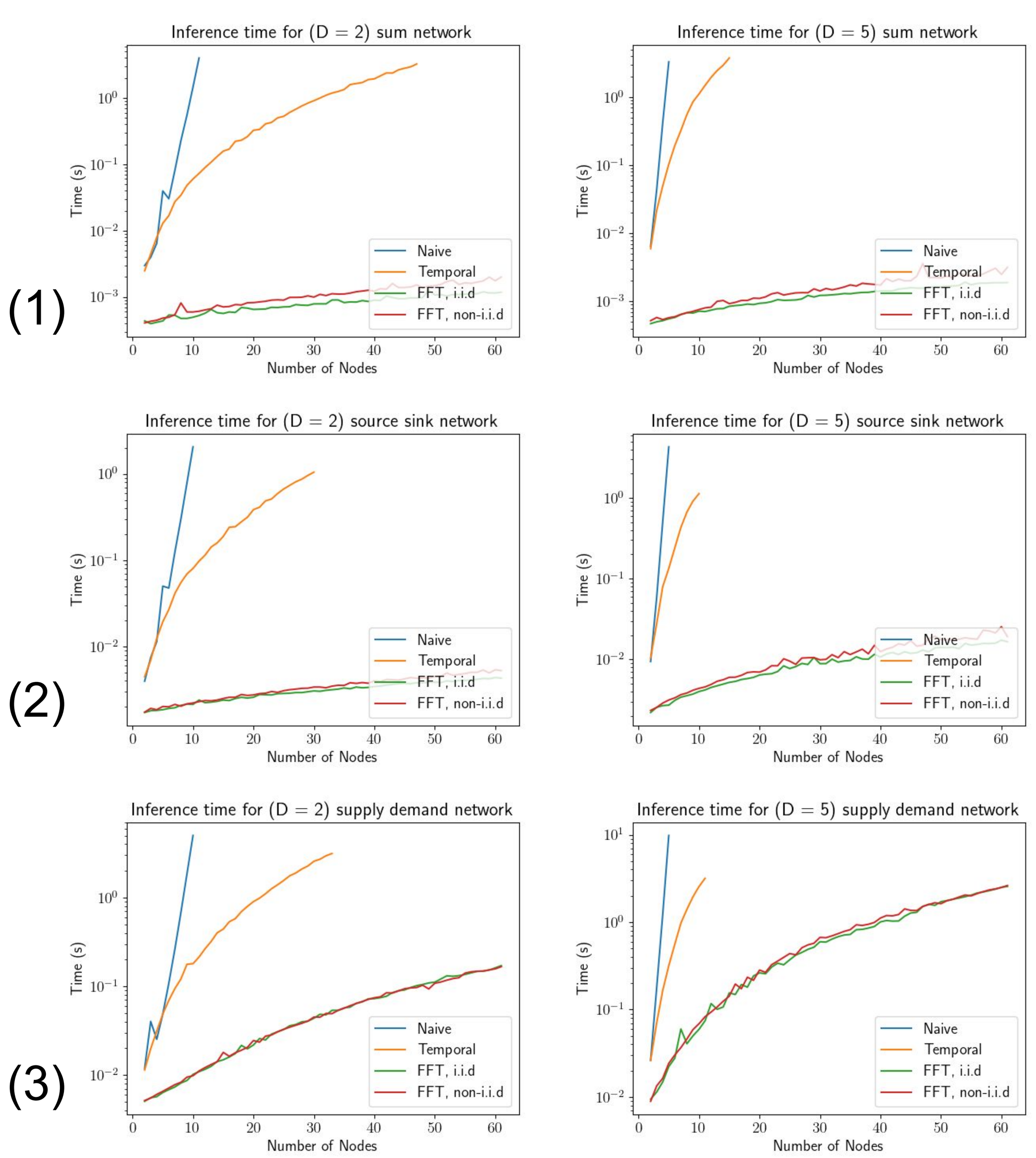
- Evaluate *ci-elim-FFT* compared to *ci-elim-bel* on general two layer additive networks (4):



Two layer additive network (4)

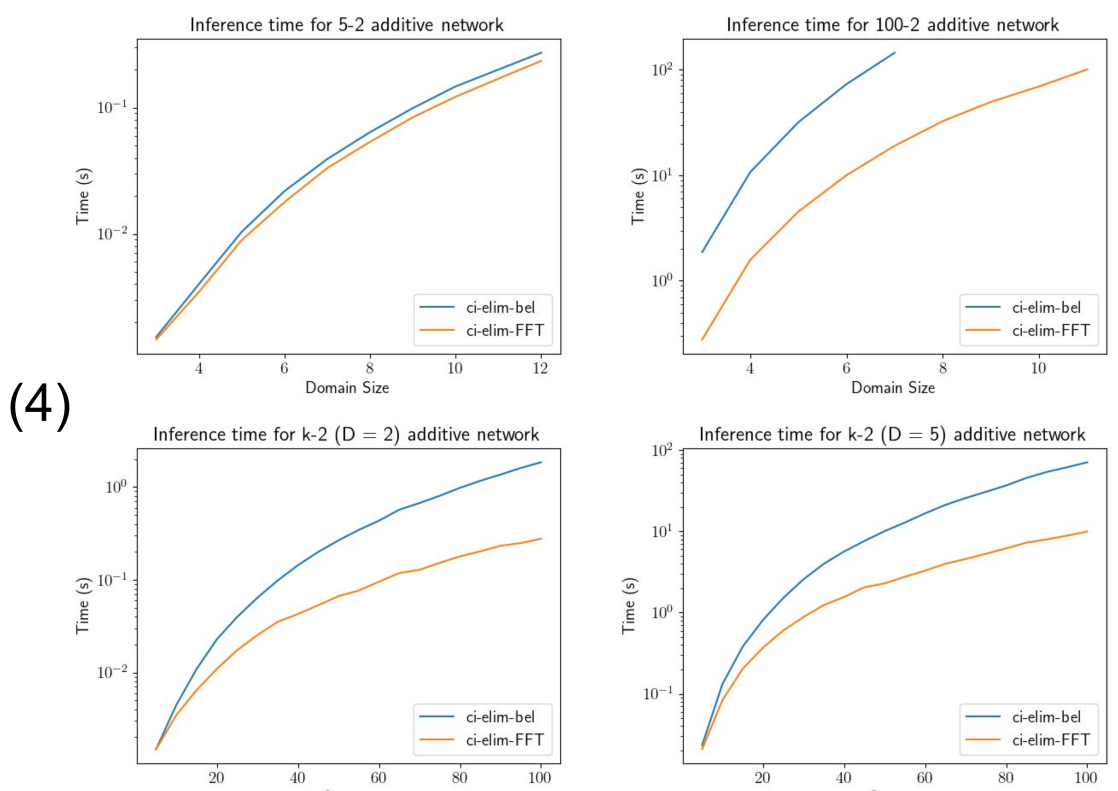
Results

(1) (2) (3)



The FFT reduction method obtains a significant computational advantage over the naive approach as well as inference on the temporal decomposition

(4)



ci-elim-FFT demonstrates better scaling with respect to number of source nodes and domain size in general additive networks compared to *ci-elim-bel*

Future Research

- Explore integration into existing lifted variable elimination algorithms

Acknowledgements

This work supported in part by NSF grant IIS-2008516 and AFOSR grant 1010GWA357.