An FPGA Accelerator of Bayesian Network Structure Learning Using Parallel Calculation of Local Scores

Ryota Miyagi
miyagi.ryota.86s@st.kyoto-u.ac.jp
Kyoto University, Japan

Hideki Takase
takasehideki@hal.ipc.i.u-tokyo.ac.jp
The University of Tokyo, Japan

1 BAYESIAN NETWORKS

A Bayesian network (BN) is a probabilistic graphical model that encodes conditional independence relations among random variables to a directed acyclic graph (DAG) [1]. Using BN, we can efficiently estimate the posterior distribution from the given evidence. Thus, BN has various applications in various fields, such as medical diagnosis, financial analysis, genetic phylogenetic analysis, etc.

There is significant interest in learning a DAG structure of a BN from data. The log marginal likelihood with Dirichlet priors over the BN parameters enables us to quantitatively evaluate how well a DAG structure candidate of a BN encodes the conditional independence relations among random variables. The entire graph score by log perimeter likelihood can be represented as a sum of substructure scores called local scores. Therefore, we can find the optimal DAG structure by dynamic programming based on the local scores calculated in advance [2]. In general, to reduce the number of required local scores, we choose a specific value as an upper limit of the number of parent variables because a substructure in which one variable has many parent variables is unlikely to have a high local score. Despite the limitation, the calculation of the local score is still time-consuming.

2 LOCAL SCORES CALCULATION

The local score calculation is reduced to counting the number of data supporting a specific substructure in the whole dataset. However, it is impossible to store a vast dataset for each local score calculation module. Thus, we store the dataset in one place and stream it to each parallel calculation module at the appropriate time. FPGAs are the ideal device for such data flow architectures.

Fig. 1 shows the overview of the proposed accelerator. The accelerator has P parallel calculation modules according to the amount of FPGA resources, and the dataset is stored in one place of global memory. In the count-up phase, the accelerator simultaneously streams the dataset to each module. Each module counts the number of data supporting the target substructure in parallel and does not store the dataset. In this manner, by utilizing FPGA resources to configure domain-specific parallel dataflows, the accelerator extracts a high degree of data parallelism and pipeline parallelism with few memory resources. In addition, it promises to speed up larger-scale problems because the performance improves as the FPGA resources increase.

3 EVALUATION AND CONCLUSION

We designed two accelerators with different degrees of parallelism in C++, applied Vitis 2020.2 HLS to them targeting Xilinx Alveo U50, and compared single-core software execution (SW) and execution with an FPGA accelerator (HW) using Vitis 2020.2 on Ubuntu 18.04 with Intel Xeon W-2265 processor and 64Gb RAM. For artificial BN with 30 binary variables with datasets of 1000 or 10000, we

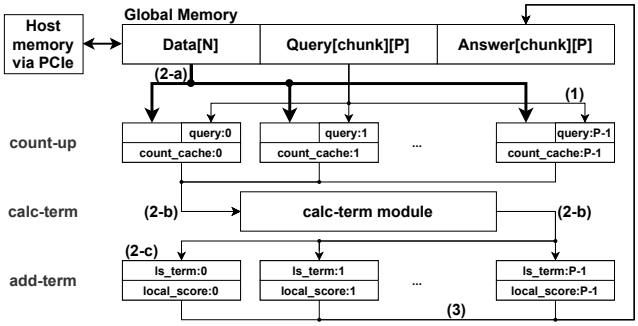

Figure 1: Overview of the FPGA accelerator

respectively evaluated: the time to calculate all local scores for a specific number of parent variables (Table 1) and the time to learn the optimal structure when varying the upper limit of the number of parent variables. Each execution was terminated after 5h.

The evaluation results show that the proposed method calculates the local score up to 230 times faster and learns the optimal DAG structure 3.5 times faster, thus enabling calculations that are too time-consuming for software. Furthermore, a comparison of the accelerators proves that the performance improves as the FPGA resources increase. The proposed method promises to extract a higher degree of parallelism using an FPGA cluster.

Acknowledgments The part of this work was supported by JST PRESTO Grant Number JPMJPR18M8. The authors would like to thank Prof. Kentaro Sano at RIKEN R-CCS for helpful discussions.

REFERENCES

[1] Judea Pearl. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
[2] Tomi Silander and Petri Myllymaki. 2012. A simple approach for finding the globally optimal Bayesian network structure. *Proceedings of the 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI-06).*

Table 1: Comparison of local scores calculation time [s]

N	m	SW	HW(P=1024)	HW(P=2048)
1000	5	130.059	1.824	1.667
1000	6	949.879	14.270	12.987
1000	7	5930.838	93.125	84.612
1000	8	N/A	510.549	463.556
1000	9	N/A	2378.568	2158.771
10000	5	1268.540	7.552	5.010
10000	6	9175.792	60.080	39.712
10000	7	N/A	394.153	260.212
10000	8	N/A	2166.152	1429.310
10000	9	N/A	10104.604	6665.651