High Performance Flow Field Visualization with High-Order Access Dependencies

Jiang Zhang\textsuperscript{1} \textsuperscript{*} Hanqi Guo\textsuperscript{3} \textsuperscript{†} Xiaoru Yuan\textsuperscript{1,2} \textsuperscript{‡}

1) Key Laboratory of Machine Perception (Ministry of Education), and School of EECS, Peking University
2) Beijing Engineering Technology Research Center of Virtual Simulation and Visualization, Peking University
3) Argonne National Laboratory

\textbf{Abstract}

We present a novel model based on high-order access dependencies for high performance pathline computation in flow field. The high-order access dependencies are defined as transition probabilities from one data block to other blocks based on a few historical data accesses. Compared with existing methods which employed first-order access dependencies, our approach takes the advantage of high order access dependencies with higher accuracy and reliability in data access prediction. In our work, high-order access dependencies are calculated by tracing densely-seeded pathlines. The efficiency of our proposed approach is demonstrated through a parallel particle tracing framework with high-order data prefetching. Results show that our method can achieve higher data locality than the first-order access dependencies based method, thereby reducing the I/O requests and improving the efficiency of pathline computation in various applications.

\textbf{Keywords:} Flow visualization, High-Order, Data prefetching

1 \textbf{Introduction}

Pathline tracing is the fundamental technique in unsteady flow visualization and analysis. Various applications require massively tracing streamlines or pathlines. However, pathline computation is both data- and computational-expensive. The dominant cost in pathline tracing is I/O, which may take up to 90\% of the time. The key to mitigating the data I/O burden is data locality, which has been studied for years. To reduce the cost of data accesses, efforts have been made to improve data locality by incorporating data access patterns \cite{2, 1}.

Yet in flow visualization, it is challenging to model data access patterns because they are implicitly determined by flow field features. Hence the access patterns are usually regarded as random in most studies.

In this work, we introduce a novel access dependency model based on high-order Markov chains. The high-order, instead of first-order access dependencies, are calculated in order to greatly improve the data localities and efficiency in field line computation. Specifically, the historical access information is taken into account in high-order access dependencies, which improves the prediction accuracy of future data accesses. Different from the first-order access dependencies employed in \cite{3, 2, 1, 4}, the prediction of future data accesses relies on both current and several previously visited data blocks in our method. We apply high-order access dependencies to a task-parallel particle tracing framework with high-order data prefetching, and demonstrate that our method improves the usage of loaded data which is predicted by high-order access dependencies, and achieves both efficiency and scalability in pathline computation.

2 \textbf{Related Work}

In previous studies, several publications focused on first order access dependencies among data blocks. The first-order access dependencies model data access patterns though recording the access probabilities for each pair of blocks. The probability of accessing one block (a chunk of data) only depends on the previously visited block. According to the first-order access dependencies, file layout can be reorganized and optimized for streamline computation in steady flow field \cite{2} and pathline computation in unsteady flow field \cite{1}. Recently Guo et al \cite{4} introduced a novel advection-based sparse data management for efficient and scalable flow visualization. In their work, high-order access dependencies were learned and recorded as hints during particle tracing. This on-the-fly construction was also employed to build a probability graph which represents the successor relation of blocks for CFD datasets \cite{3}. In fact, more sophisticated access patterns exist in particle tracing. So our work also takes historical access information into account. By matching the high-order historical information of a pathline, future data accesses can be predicted with higher accuracy.

3 \textbf{High-Order Access Dependencies Computation}

Given an unsteady flow data, we first trace pathlines densely in the domain. And then we compute high-order access dependencies based on the data access information recorded in these pathlines. The high-order access dependencies are reusable for further various applications once they are generated.

In our work, the entire data domain is evenly partitioned into blocks. Each block contains an equal-size spatial range with sequential timesteps. Particles are densely seeded in each data block. For convenient calculation, particles originated from each data block are traced in both forward and backward directions. The corresponding pathlines are called forward pathlines and backward pathlines, respectively. Step by step, forward pathlines are traced by increasing the time while backward pathlines are traced by decreasing the time. The forward and backward pathline seeded in the same position can be actually considered as one pathline which integrates both historical and future access information.

Considering each block as the current data block, all pathlines including forward and backward pathlines started from this block are gathered for the generation of corresponding high-order access dependencies. The data access information has been recorded in these pathlines. We merge each pair of forward and backward pathline which are seeded in the same position to form a new pathline. For the merged pathlines, the data blocks accessed by corresponding backward pathlines are considered as the historical access information, while the blocks accessed by corresponding forward pathlines are seen as the possible access information in the immediate future. Since the historical access information of the merged pathlines may be different, we further cluster these pathlines into groups. For the pathlines in each group, the historical access information is the same, but the probabilities of future data accesses

\textsuperscript{*}e-mail: jiang.zhang@pku.edu.cn
\textsuperscript{†}e-mail: hguo@anl.gov
\textsuperscript{‡}e-mail: xiaoru.yuan@pku.edu.cn
Figure 1: A 2D example to show the computation of first-order and third-order access dependencies. (a) The pathlines originated from block (2, 1); (b) The graph model of access dependencies generated from (a); (c) The comparison between first-order and third-order accesses dependencies.

Figure 2: The running time and data usage of full-range analysis using Isabel dataset under different number of processes. Note that the running time and the number of processes are plotted in logarithm scale. In (a) and (b), as the order increases, the running time decreases while the data usage increases.

The performance results of different orders under different number of processes are displayed in Figure 2(a). The scalability is shown to be good as the number of processes increases. We can also clearly see that the running time will decrease as the order increases, which demonstrates that our method achieves better efficiency than the first-order access dependencies based method. To make further exploration, we compare the usage of prefetched data between different access dependencies based method. The data usage is the percentage of prefetched data that is really used. It’s obvious that with higher data usage less data will be loaded since the data is more likely to be used instead of being replaced by new data. In Figure 2(b), the result shows that using higher-order access dependencies will achieve higher prefetched data usage. This means our method can reduce I/O requests, and thus improving the efficiency of pathline computation.

5 Conclusions
In this work, we explore high-order access dependencies in unsteady flow visualization. The historical access information is integrated to support more accurate predictions of data access patterns, which greatly improves the data locality. We further apply high-order access dependencies to a parallel particle tracing framework with high-order data prefetching for pathline computation. Results demonstrate that our method achieves better efficiency than the first-order access dependencies based method.

Acknowledgements
This work is supported by NSFC No. 61170204. This work is also partially supported by the Strategic Priority Research Program - Climate Change: Carbon Budget and Relevant Issues of the Chinese Academy of Sciences Grant No. XDA05040205.

References