Novel Graph Alignment Algorithms for Identifying Non-Determinism in Large-Scale Simulations

Dhroov Pandey[1], Jack Marquez[2], Michela Taufer[2] and Sanjukta Bhowmick[1]
1.University of North Texas
2. University of Tennessee, Knoxville

ACM Reference Format:
Dhroov Pandey[1], Jack Marquez[2], Michela Taufer[2] and Sanjukta Bhowmick[1]. 2025. Novel Graph Alignment Algorithms for Identifying Non-Determinism in Large-Scale Simulations. In . ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION AND MOTIVATION

As HPC simulations increase in size and complexity, ensuring their reproducibility and reliability becomes increasingly challenging. One critical area affecting reproducibility is the non-determinism (ND) induced by asynchronous MPI communications. In cases, where the results of the simulations change across runs, it is challenging to understand whether the discrepancy is due to errors in the code or the inherent non-determinism. This issue can be addressed by comparing the event graphs (graphs of the MPI communications) across different executions [3].

ANACIN-X [2] is a software framework that traces the point-to-point communication in MPI codes and generates event graphs for the same. These event graphs can then be compared to locate areas of ND. However, due to the near regular properties of the event graph, and their large size, current network alignment algorithms or deep learning based graph encoder methods cannot accurately pinpoint the regions of non-determinism.

In this poster we present an updated version of ANACIN-X that expands the tracing functionality of point-to-point communication to incorporate collective communication, and a meta-graph heuristic to accurately align event graphs. These features together enable us to analyze complex applications such as Adaptive Mesh Refinement (AMR) [7].

2 MODELLING MPI COMMUNICATIONS

Figure 2 shows a pictoral representation of different non determinism types that can arise in MPI calls. These are;

- **Non blocking point to point communication** such as *MPI_Isend* and *MPI_Irecv*.
- **Varying neighbors in point to point** through *MPI_ANY_SOURCE*
- **Varying order of MPI calls** through functions such as *MPI_Testany*
- **Non blocking collective communication** such as *MPI_Ibcast*

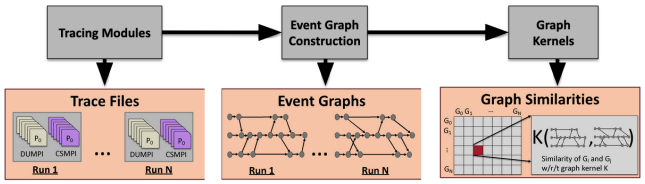

Figure 1: ANACIN-X event graph generation process [4]

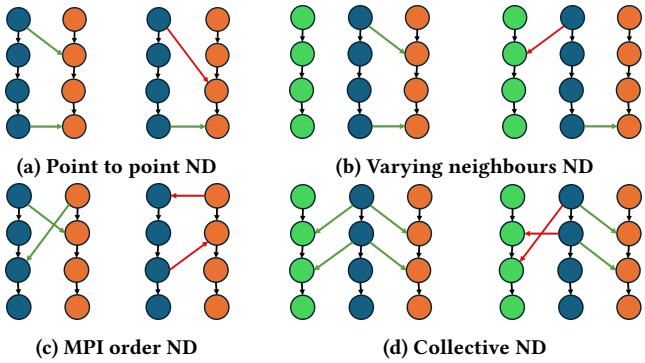

Figure 2: Different ND patterns in message ND

We map collective MPI communication by keeping two separate maps of collective calls in each process trace: one where it is a recipient of information and one where it is a transmitter. Each call is mapped by a key: the collective call channel which is uniquely identified by its root node (-1 for all to all communication), dumpi communicator number and request ID (for non-blocking MPI), and its associated value: a vector of vertex IDs. During event graph construction, we broadcast the transmitter map across all processes, and each process constructs its portion of the event graph from its recipient map.

3 ISSUES WITH EXISTING NETWORK ALIGNMENT METHODS

We address two network alignment methodologies: NetAlign [1] and Graph Auto Encoders [6].

3.1 NetAlign

NetAlign is a scalable network alignment algorithm that utilizes quatratic programming and belief propagation to generate approximate matchings in the L graph. The bipartite L graph is a preliminary mapping which contains the node similarities across graphs.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/10.1145/nnnnnnn.nnnnnnn

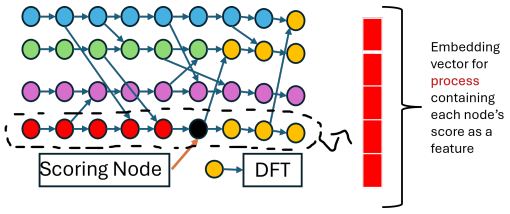

Figure 3: Node scoring heuristic. The black node is the node being scored. The yellow nodes are the vertices in a depth first tree rooted at the black node.

A naive L graph would be a complete bipartite graph with bipartite sets as the nodes of the target graphs. For effective alignment, NetAlign requires sparse L graphs.

Evaluation on Event Graphs We use graphlet degree vector (GDV) [5] similarity to create L. Due to the regular and structurally homogeneous nature of event graphs, L is usually dense and this leads to poor alignment results.

3.2 Graph Auto Encoders

Graph Auto Encoders (GAE) embed graphs into vector spaces via learned representations. The intermediary stage before aggregating a graph embedding involves latent node embeddings that contain structural and feature based information. These node embeddings can be used to compare nodes across the target graphs via vector similarity and generate an alignment.

Evaluation on event graphs We observe that GAE are unable to scale with increasing pattern complexity and size of event graphs. This is due to their ability to aggregate information from a limited radius neighborhood that depends on the size of the GAE neural architecture. For large event graphs, the required radius would yield a computationally infeasible GAE architecture.

Since the existing techniques are insufficient for event graph alignment, we propose a meta-graph heuristic that exploits structural constraints of event graphs to align them.

4 META GRAPH HEURISTIC

We propose a meta graph heuristic that exploits structural constraints of event graphs along with utilizing a message passing scheme that is suited to sparse directed acyclic graphs.

(i) We transform the event graphs into meta graphs where meta nodes represent directed linear subgraphs of the event graph corresponding to the sequence of MPI routines in a process, and the edges represent communication between processes. *(ii)* We encode vector embeddings for meta nodes based on a scalar scoring of their member event graph nodes. *(iii)* We assume an all-to-all mapping across the meta graph nodes and do a preliminary pruning based on a thresholding criterion resulting in candidate matching sets for each node. *(iv)* We align the meta nodes based on vector similarity of their embedding with their candidate set.

The scalar scores for event graph nodes used in step (ii) are generated using a recursive aggregation of the Depth-First-Tree (DFT) rooted at that node as shown in Fig. 3. We mathematically enumerate the scoring scheme as follows:

MPI Application	Point-to-point	varying neighbors	MPI order	Collective
Message Race	✓			
amg2013	✓	✓		
mcb grid	✓	✓	✓	
c-amg2013	✓	✓		✓
c-mcb grid	✓	✓	✓	✓

Table 1: ND types present in our evaluation dataset

$$S(v) = ln[Poly(lvl(v)) + \sum_{c \in Nb(v)} S(c)]$$

where $S(v)$ is the score of v, $lvl(v)$ is the topological level of v, $Poly(i)$ is a high order polynomial of i, and $Nb(v)$ is the set of nodes connected to out-edges of v. Then we embed each meta node into a vector space where the scalar score of each node belonging to that process is a component of the embedding.

5 EVALUATION

We evaluate the alignment accuracy of our meta graph heuristic, netalign and GAE on the simulated non-deterministic HPC applications amg2013, message race, mcb grid and hybridized applications c-amg2013 and c-mcb grid which have collective MPI communication integrated. Table 1 shows the ND properties of our evaluation dataset. We observe in Table 2 that our algorithm is able to outperform the conventional methods.

Table 2: Alignment Accuracy evaluation

MPI Application	#Procs	#Nodes	NetAlign	GAE	Meta Graph Heuristic
AMG 2013	16	1.6k	0.10	0.59	1
AMG 2013	32	2.2k	0.04	0.52	1
AMG 2013	64	330k	0.02	0.28	1
Message Race	16	660	0.14	0.87	1
Message Race	32	1.1k	0.11	0.72	1
Message Race	64	2.8k	0.10	0.70	1
MCB Grid	16	320k	0.005	0.21	1
MCB Grid	32	770k	0.001	0.17	0.875
MCB Grid	64	154k	0.02	0.19	0.938
c-amg2013	16	2.2k	0.08	0.46	1
c-mcb grid	16	270k	0.02	0.15	0.94

Acknowledgement: The work is partially funded by NSF grants #1900765 and #1900888.

REFERENCES

[1] Bayati, M., Gerritsen, M., Gleich, D. F., Saberi, A., and Wang, Y. Algorithms for large, sparse network alignment problems. In *2009 Ninth IEEE International Conference on Data Mining* (2009), pp. 705–710.
[2] Bell, P., Suarez, K., Chapp, D., Tan, N., Bhowmick, S., and Taufer, M. Anacin-x: A software framework for studying non-determinism in mpi applications. *Software Impacts 10* (2021), 100151.
[3] Chapp, D., Tan, N., Bhowmick, S., and Taufer, M. Identifying degree and sources of non-determinism in mpi applications via graph kernels. *IEEE Transactions on Parallel and Distributed Systems PP* (05 2021), 1–1.
[4] Chapp, D., Tan, N., Bhowmick, S., and Taufer, M. Identifying Degree and Sources of Non-Determinism in MPI Applications Via Graph Kernels. *IEEE Trans. Parallel Distributed Syst. (TPDS) 32*, 12 (2021), 2936–2952.
[5] Ho, H., Milenković, T., Memisević, V., Aruri, J., Przulj, N., and Ganesan, A. Protein interaction network topology uncovers melanogenesis regulatory network components within functional genomics datasets. *BMC systems biology 4* (06 2010), 84.
[6] Kipf, T. N., and Welling, M. Variational graph auto-encoders, 2016.
[7] Vaughan, C. T., and Barrett, R. F. Enabling tractable exploration of the performance of adaptive mesh refinement. In *2015 IEEE International Conference on Cluster Computing* (2015), pp. 746–752.