Underpass: Directing Regular Path Queries Around the Host in Near-Memory Processing

Bart Hodes, Manil dev Gomony, and Nikolay Yakovets
Eindhoven University of Technology, Netherlands
{b.hodes, m.gomony, n.yakovets}@tue.nl

Regular path queries (RPQs) are fundamentally memory access constrained, making them ideal for near-memory processing (NMP) architectures. NMP trades a few fast cores for many slower DRAM Processing Units (DPUs) that each have high bandwidth to their own DRAM bank. However, prior NMP systems route all cut edges via the host, creating a bottleneck as DPU counts grow [1]. We profiled a multi-source three-hop reachability query and found that scaling from 64 to 2,560 DPUs yielded at most 2.7× speedup (1.5× average) for host-relayed traversal. Performance barely scales with DPU count, defeating the purpose of such architectures.

We introduce Underpass, a redesigned NMP graph traversal system that removes the host from the traversal path. Underpass adds three key mechanisms: (1) an intra-channel DPU network based on PIMnet[2] that enables direct DPU-to-DPU communication within a memory channel, eliminating host-relayed routing for cut edges. (2) Degree-aware graph partitioning that distributes high-degree vertices across DPUs while keeping good work balance. Lastly, (3) one graph copy per memory channel, which enables independent traversal states to execute in parallel without inter-channel communication.

In cycle-level simulation, scaling from 256 to 2,560 DPUs, Underpass achieves 3.2–9.8× speedup on four skewed graphs compared to 0.71–1.73× for host-relayed traversal. At 2,560 DPUs, Underpass is 3.9–26× faster than the best host-relayed design on five skewed graphs while maintaining good mesh graph performance. Host routing overhead drops below 6% of total execution time, with the DPU kernel accounting for 72–86%. The system is now work-bound rather than communication-bound, demonstrating that intra-channel communication is essential for scaling NMP on graph queries.

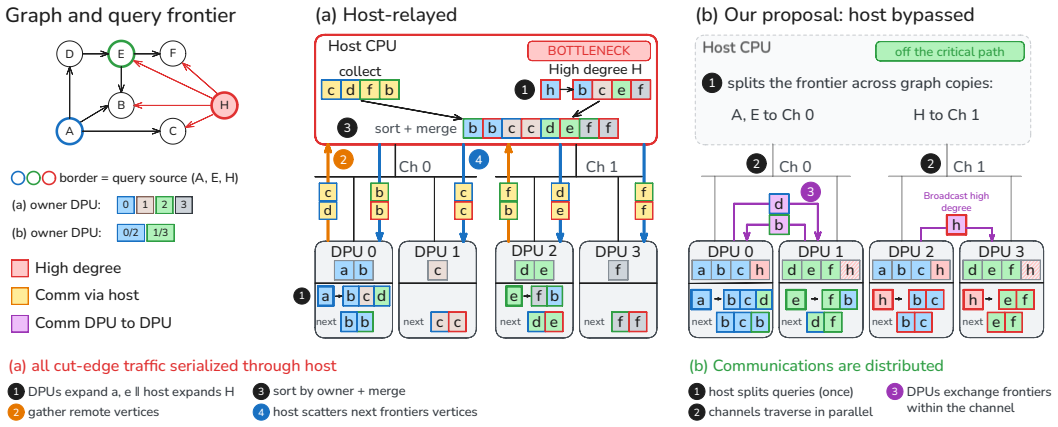


Figure 1: Host-relayed traversal (left) routes all inter-DPU traffic through the host, which limits scaling. Underpass (right) exchanges cut-edge traffic directly between DPUs over an intra-channel network and keeps one graph copy per memory channel, so traversals in different channels run in parallel.

References

[1] W. Kong, S. Zheng, Y. Hua, R. Ma, Y. Wen, G. Wang, C. Zhou, and L. Huang. PimBeam: Efficient Regular Path Queries Over Graph Database Using Processing-in-Memory. *IEEE Transactions on Parallel and Distributed Systems*, 36(5):1042–1057, May 2025.

[2] H. Son, G. Jonatan, X. Wu, H. Cho, K. Shivdikar, J. L. Abellán, A. Joshi, D. Kaeli, and J. Kim. PIMnet: A Domain-Specific Network for Efficient Collective Communication in Scalable PIM. In *2025 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, pages 1557–1572, Las Vegas, NV, USA, Mar. 2025. IEEE.