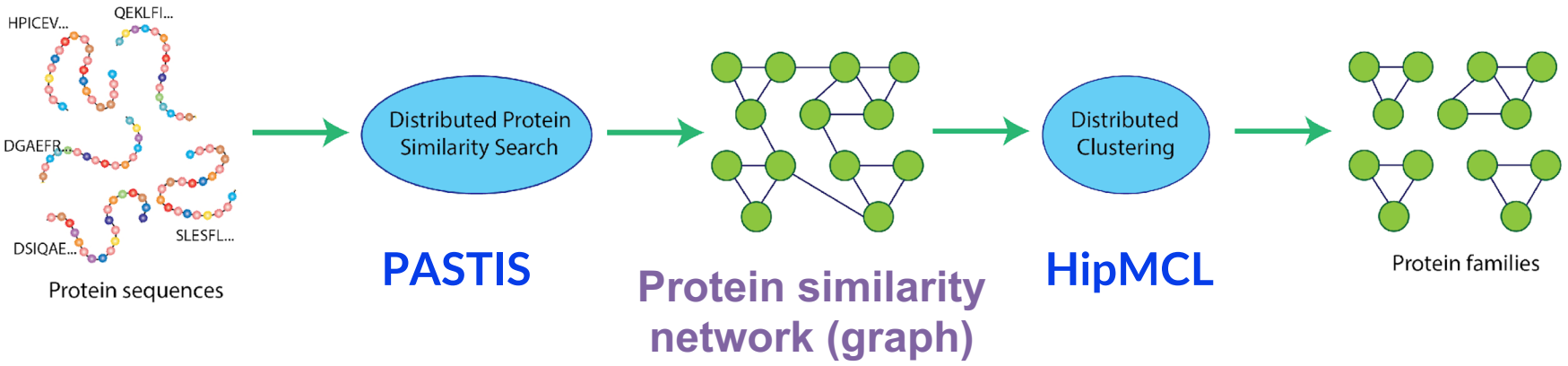
Large Scale Metagenome Protein Family Detection with HipMCL

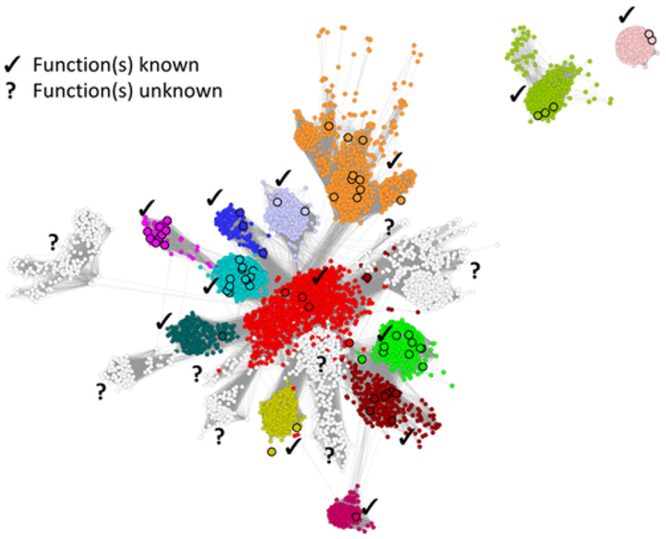
Aydın Buluç
Applied Math & Computational Research Division, LBNL
EECS Department, UC Berkeley

JGI User Meeting, Walnut Creek
Oct 4, 2024

Protein Family Identification



- Problem: Given a large collection of proteins, identify groups of proteins that are homologous (i.e. descended from a common ancestor).
- Homologous proteins often have the same function
- Often, only sequences (and not structure) of the proteins are available, so we infer homology via sequence similarity



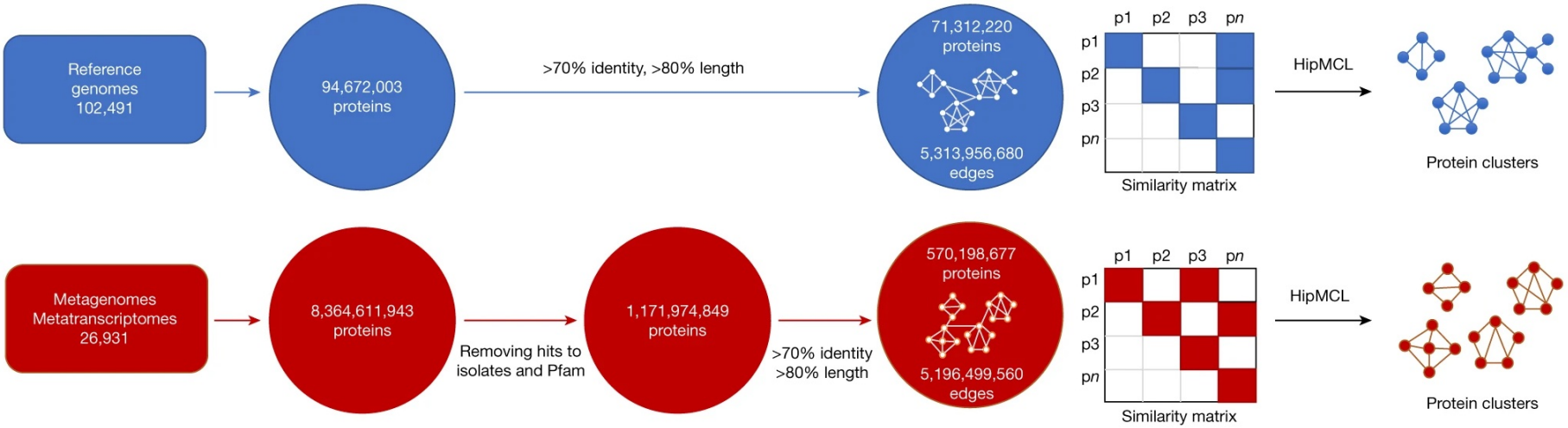
Novel Protein Families in Microbial Dark Matter

Microbial dark matter: novel proteins after removing matches to a database of over 100,000 genomes (including Archaeal, Bacteria, Viral and Eukaryotic)

Unraveling the functional dark matter through global metagenomics

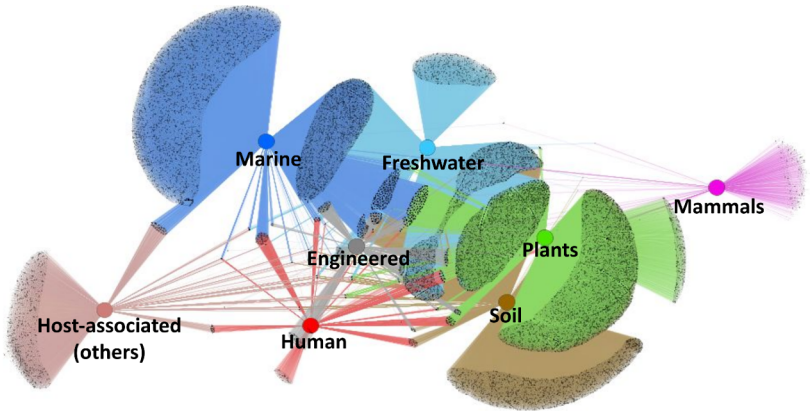
Georgios A. Pavlopoulos ✉, Fotis A. Baltoumas, Sirui Liu, Oguz Selvitopi, Antonio Pedro Camargo, Stephen Nayfach, Ariful Azad, Simon Roux, Lee Call, Natalia N. Ivanova, I. Min Chen, David Paez-Espino, Evangelos Karatzas, Novel Metagenome Protein Families Consortium, Ioannis Iliopoulos, Konstantinos Konstantinidis, James M. Tiedje, Jennifer Pett-Ridge, David Baker, Axel Visel, Christos A. Ouzounis, Sergey Ovchinnikov, Aydin Buluç & Nikos C. Kyrpides ✉

Nature **622**, 594–602 (2023) | Cite this article



Diversity of Novel Protein Families

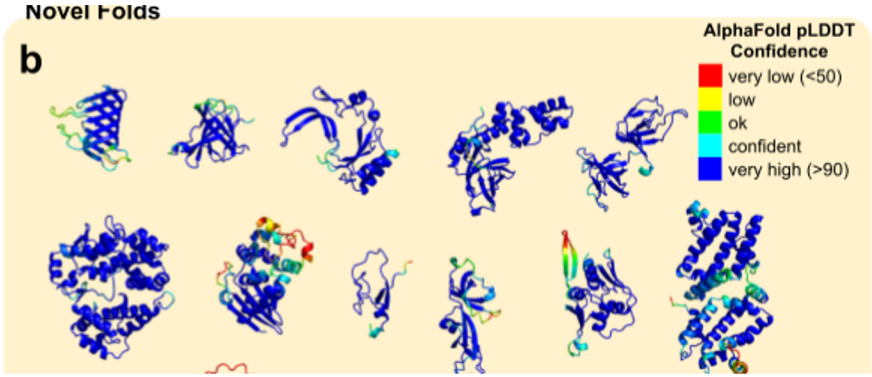
Novel protein clusters are distributed across 8 ecosystem types



Network representation of protein clusters (gray peripheral) and their associated ecosystems (colored central)

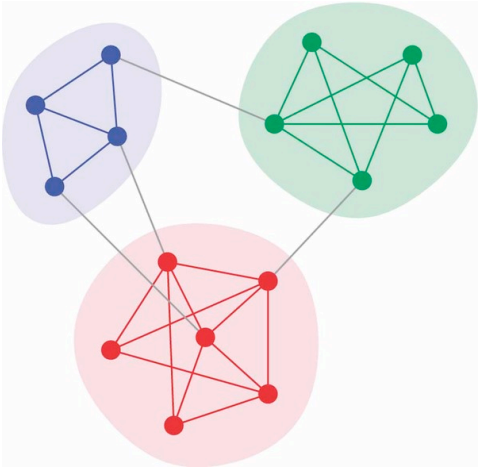
Distribution of protein structures

- 1,215 unique structures were predicted using AlphaFold
- 1,092 structures has hits in Protein Data Bank: their functions can be predicted!
- 141 structures with no database hits are novel folds



The Markov Cluster Algorithm (MCL)

Widely popular and successful algorithm for discovering clusters (e.g. protein families) in protein interaction and protein sequence similarity networks



The number of **edges or higher-length paths** between two arbitrary nodes in a cluster is greater than the number of paths between nodes from different clusters

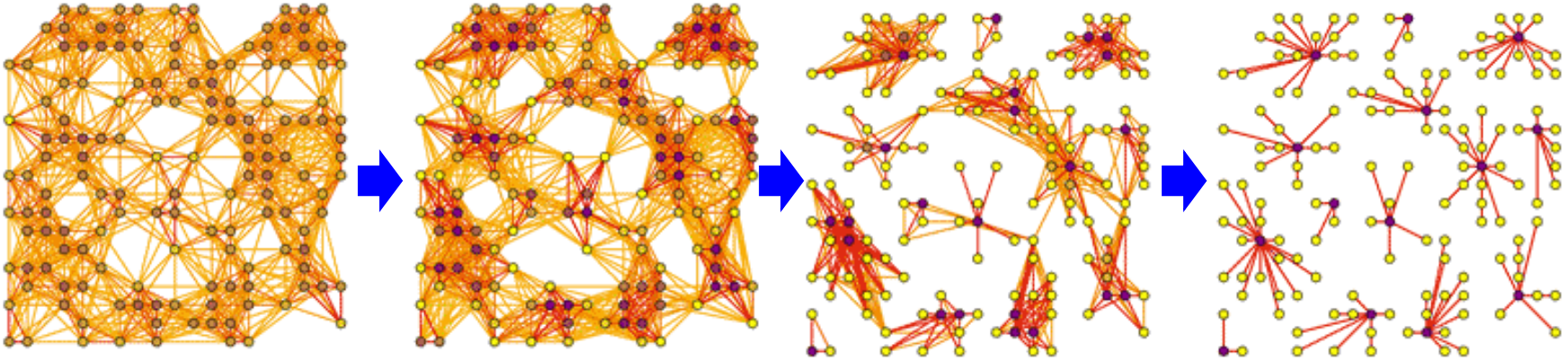
↓

Random walks on the graph will frequently remains within a cluster

↓

The algorithm **computes the probability** of random walks through the graph and **removes lower probability terms** to form clusters

The Markov Cluster Algorithm (MCL)



Initial network | Iteration 1 | Iteration 2 | Iteration 3

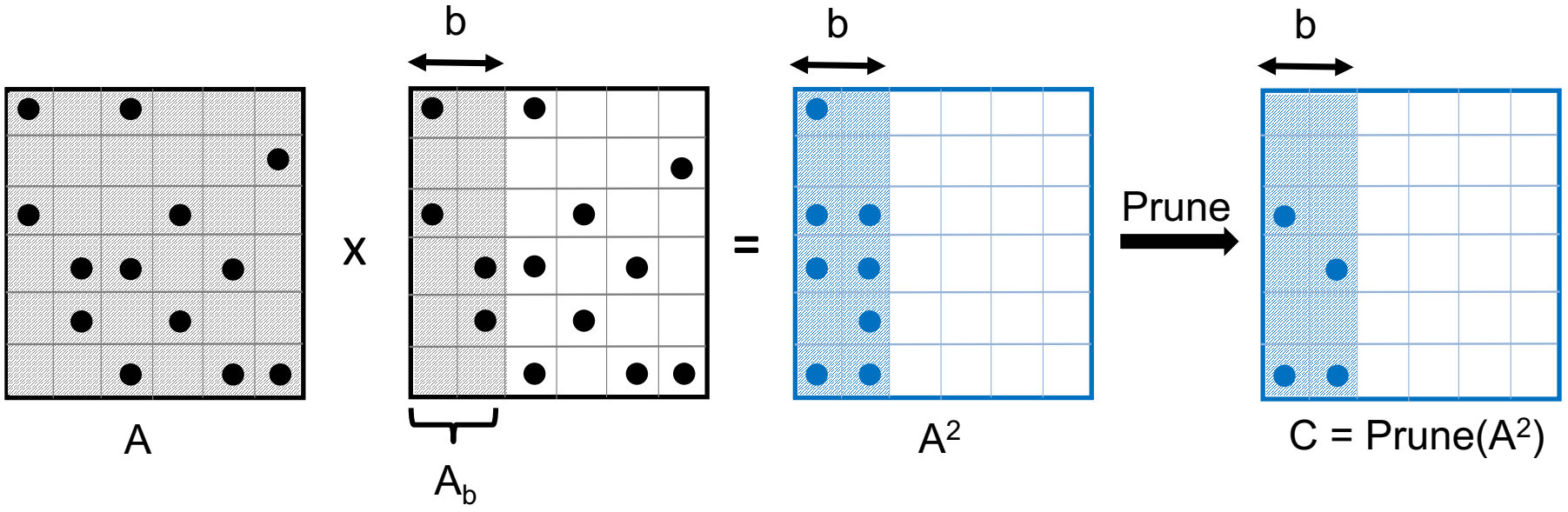
At each iteration:

Step 1 (Expansion): Squaring the matrix while pruning (a) small entries, (b) denser columns

Naïve implementation: sparse matrix-matrix product (SpGEMM), followed by column-wise top-K selection and column-wise pruning

Step 2 (Inflation) : taking powers entry-wise

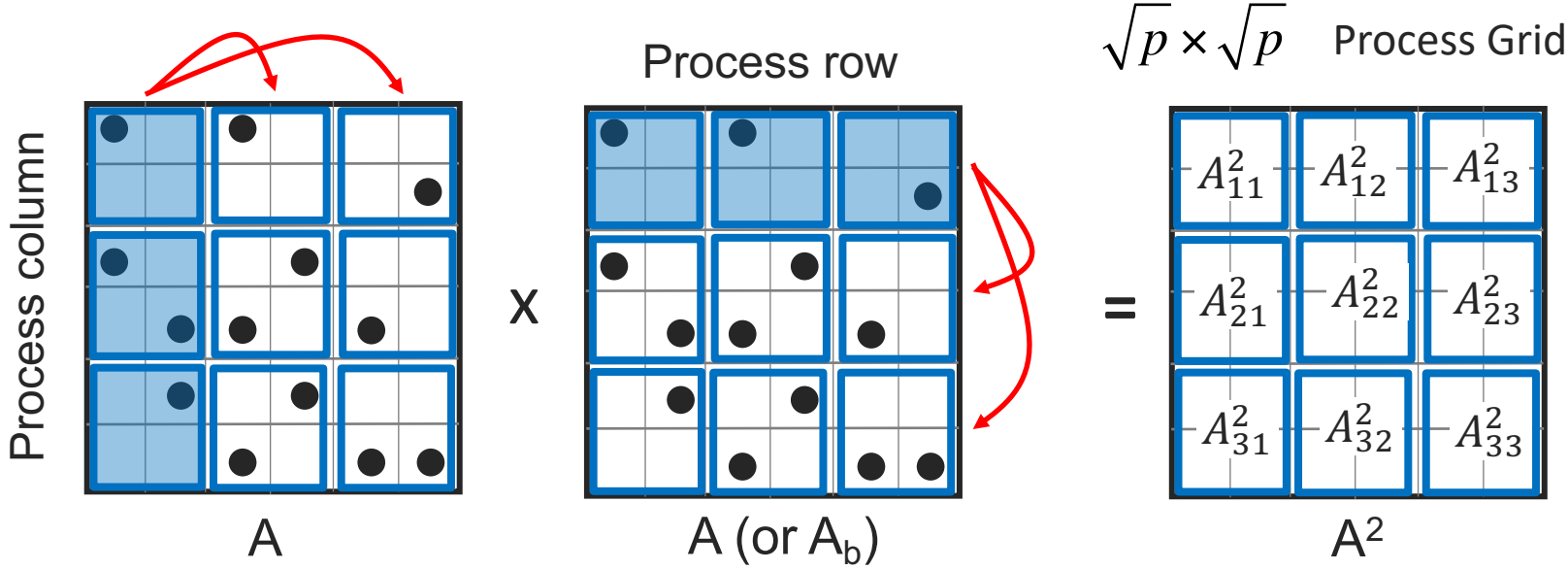
A combined expansion and pruning step



- b: number of columns in the output constructed at once
  - Smaller b: less parallelism, memory efficient (b=1 is equivalent to sparse matrix-sparse vector multiplication used in MCL)
  - Larger b: more parallelism, memory intensive

HipMCL: High-performance MCL

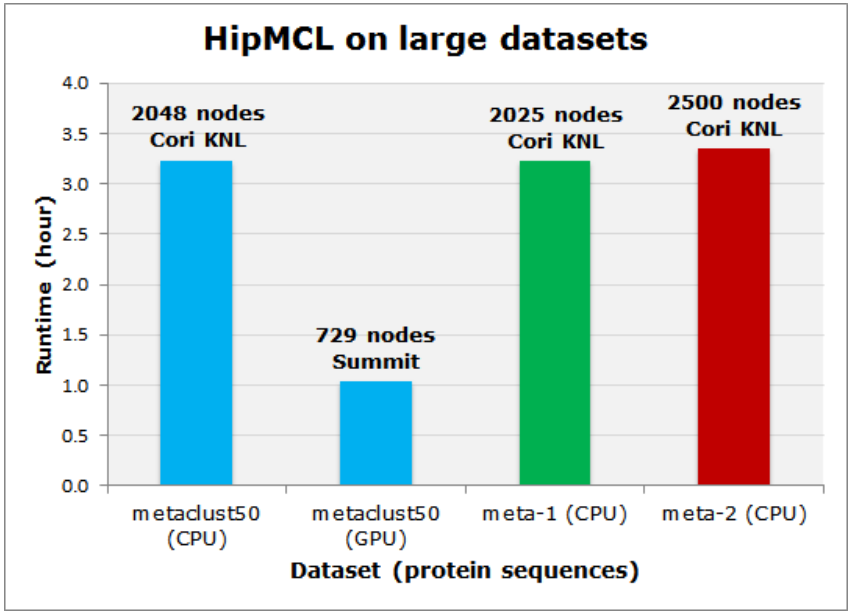
- MCL process is both **computationally expensive** and **memory hungry**, limiting the sizes of networks that can be clustered
- HipMCL overcomes such limitation via **sparse parallel algorithms**.
- **Up to 1000X times faster** than original MCL with same accuracy.



A. Azad, G. Pavlopoulos, C. Ouzounis, N. Kyrpides, A. Buluç; HipMCL: a high-performance parallel implementation of the Markov clustering algorithm for large-scale networks, *Nucleic Acids Research, 2018*

HipMCL on large networks

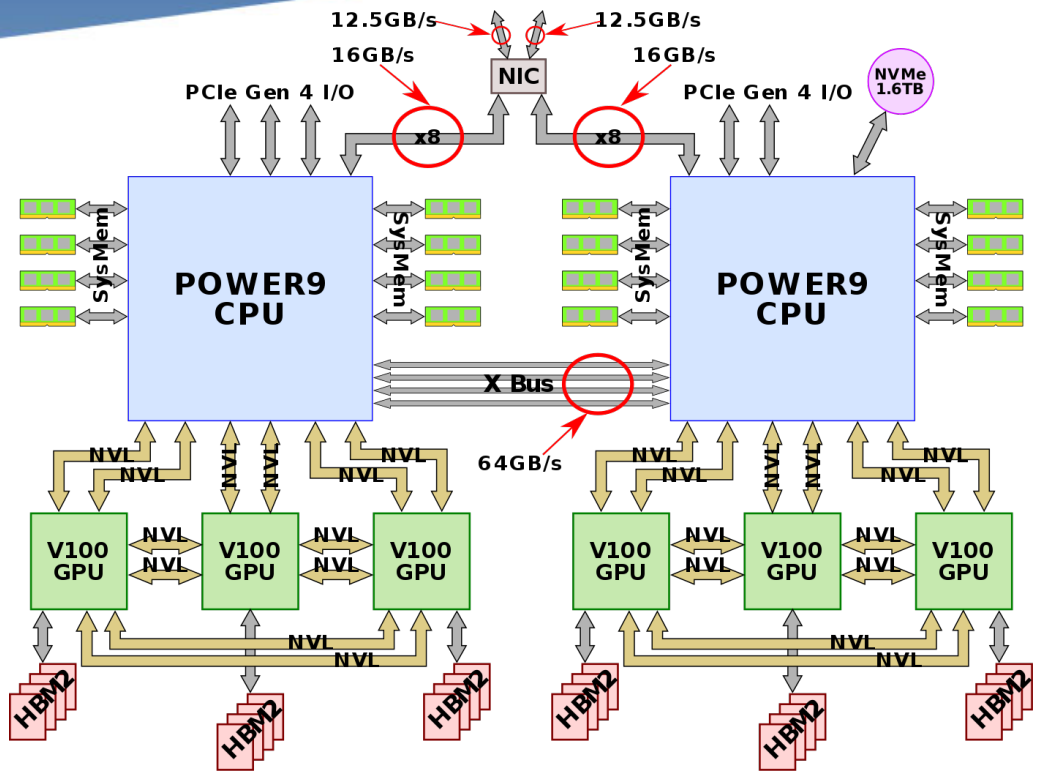
- Clustered protein sequence datasets:
  - **metaclust50** (CPU and GPU)
    - **380 million sequences**
  - **meta-1** (CPU)
    - **662 million sequences**
  - **meta-2** (CPU)
    - **570 million sequences**
- HipMCL (CPU and GPU versions)
  - GPU version up to 12x faster
- Able to cluster millions of sequences within a few hours

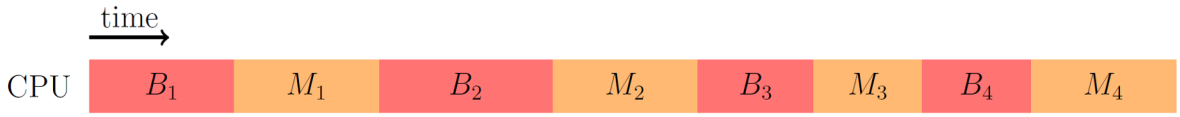


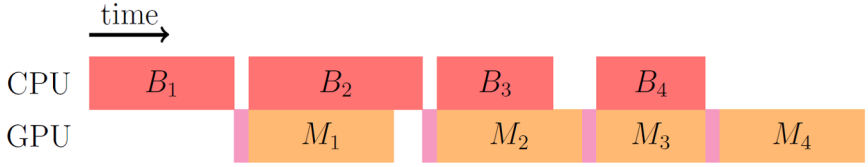
MCL cannot cluster these networks

HipMCL on Supercomputers with accelerators

- Recent top supercomputers are all accelerated (e.g. with GPUs)
- This is what a ORNL Summit node looks like
- There are 4608 such nodes in the system
- Challenges: (1) Utilizing all GPUs, (2) hiding the communication



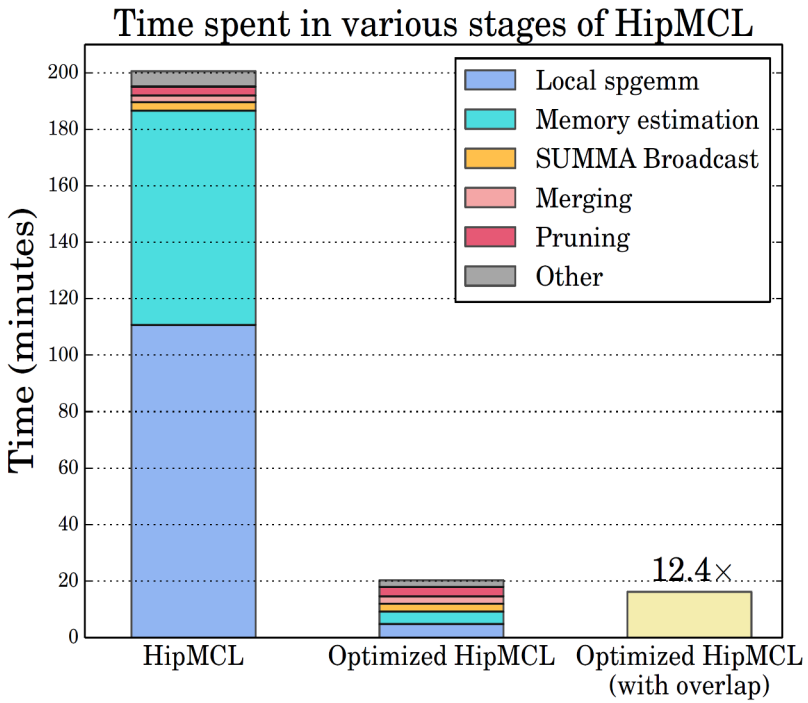


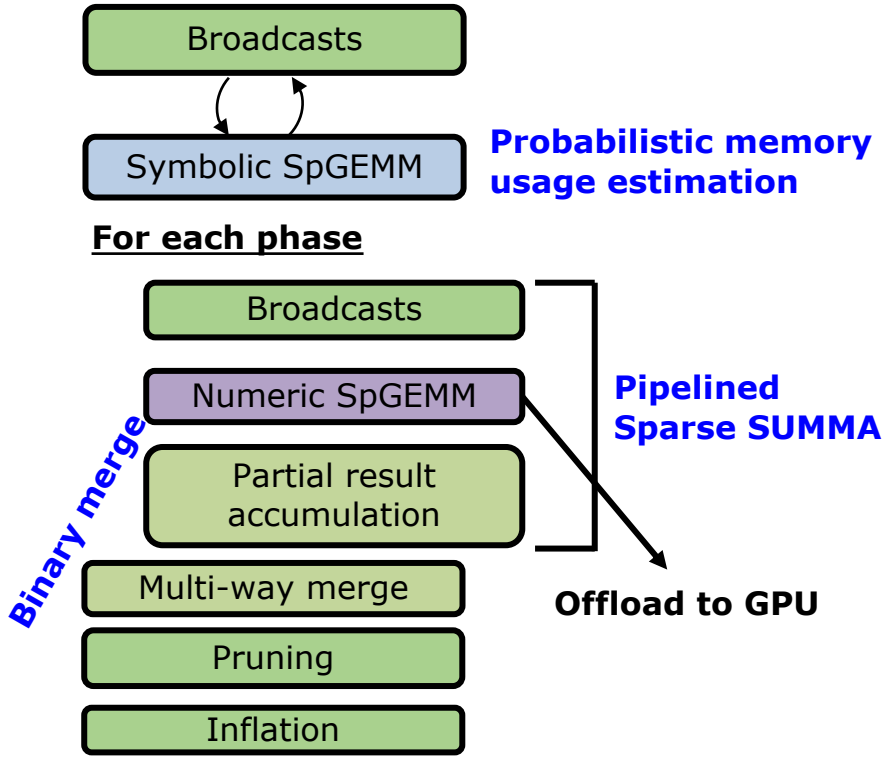


Pipelined Sparse SUMMA

Joint CPU-GPU distributed memory expansion of MCL algorithm

HipMCL on Supercomputers with accelerators





Other changes to HipMCL for the CPU-GPU workflow:

- *Randomized memory estimation algorithm* avoids symbolic phase
- New *eager binary merging* reduces memory footprint
- Integration of a much faster hash-based CPU SpGEMM algorithm

O. Selvitopi, M.T. Hussain, A. Azad, and A. Buluç. Optimizing high performance Markov clustering for pre-exascale architectures. IPDPS, 2020

HipMCL repo and wiki

- Repo: https://bitbucket.org/azadcse/hipmcl/src/master/
- Wiki with installation instructions: https://bitbucket.org/azadcse/hipmcl/wiki/Home

- The CPU version is in the **main branch** and the GPU version is in the **hipmcl-gpu branch.**
- The build instructions for the CPU and GPU version are the same.
- The HipMCL repo code is stable and uses an older (but well tested) version of CombBLAS.
- You can replace CombBLAS folder with the new CombBLAS release from https://github.com/PASSIONLab/CombBLAS for the best performance:

Acknowledgments

Ariful Azad
Indiana U.

Oguz Selvitopi
AMCRD/LBNL

Md Taufique Hussain
Indiana U.

Georgios Pavlopoulos
BSRC Fleming

Nikos Kyrpides
JGI/LBNL

Christos Ouzounis
AUTH and CERTH

Extra Slides

Challenges of similarity search on huge datasets

- **Computational patterns**
  - Search operations on sequences: **memory-bound & irregular**
  - Batch pairwise alignments: **compute-bound & regular**
    - Edit distance computations
- **Memory requirements**
  - All-vs-all search
  - Example: 100 million sequences
    - Candidate pairs ~ **trillions (10^{12})**
    - Alignments ~ **hundreds of billions (10^{11})**
    - Similar pairs ~ **billions (10^{9})**

PASTIS as 2022 Gordon Bell Finalist

Finalist for the 2022 ACM Gordon Bell Prize
https://en.wikipedia.org/wiki/Gordon_Bell_Prize

PASTIS repo: https://github.com/PASSIONLab/PASTIS

Extreme-scale many-against-many protein similarity search

Oguz Selvitopi*, Saliya Ekanayake†, Giulia Guidi‡, Muaaz G. Awan§, Georgios A. Pavlopoulos¶, Ariful Azad‖, Nikos Kyrpides**, Leonid Oliker*, Katherine Yelick‡*, Aydın Buluç*‡

*Applied Mathematics & Computational Research Division, Lawrence Berkeley National Laboratory, USA
†Microsoft Corporation, USA
‡University of California, Berkeley, USA
§NERSC, Lawrence Berkeley National Laboratory, USA
¶Institute for Fundamental Biomedical Research, BSRC "Alexander Fleming", 34 Fleming Street, 16672, Vari, Greece
‖Indiana University, USA
**Joint Genome Institute, Lawrence Berkeley National Laboratory, USA
roselvitopi@lbl.gov

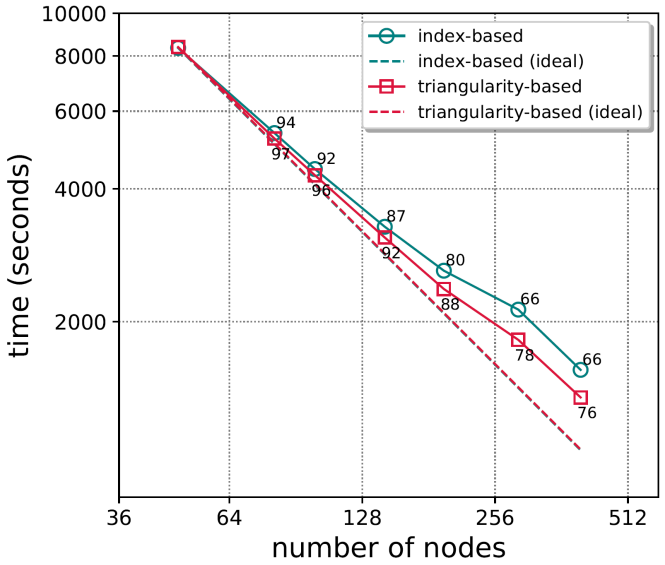
Abstract-- … We unleash the power of over 20,000 GPUs on the Summit system to perform all-vs-all protein similarity search on one of the largest publicly available datasets with 405 million proteins, in less than 3.5 hours, cutting the time-to-solution for many use cases from weeks. The variability of protein sequence lengths, as well as the sparsity of the space of pairwise comparisons, make this a challenging problem in distributed memory …

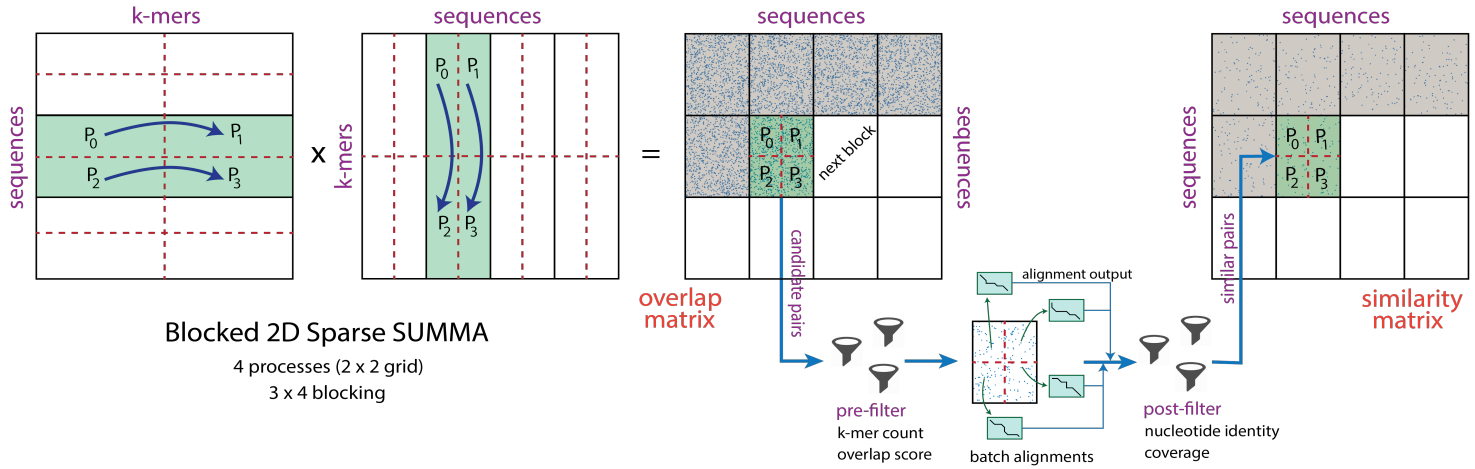
Extreme-scale many-against-many protein similarity search

Advances: memory-consumption optimizations, new parallel algorithms taking advantage of the symmetry in the sequence similarity matrix, GPU acceleration, the ability to address load imbalance issues

Result: many-against-many protein search on 405 million proteins with PASTIS on 3364 compute nodes of ORNL Summit in 3.4 hours, sustaining a rate of 691 million alignments/sec and attaining ~176 TCUPs (Tera Cell Updates/sec).

The output protein sequence similarity graph is 27 TB.





Similarity search at scale: Advancements

- Discovered candidates: **96T**
- Performed alignments: **8.6T**
- Similar pairs: **1.1T**
- Runtime: **3.44 hours**