

Meta Dynamic Programming

Paul O'Neil
Juneki Hong

Dynamic Programming across Dynamic Programming problems

- Local Alignment problems are frequently queried in large batches.
- Individual Local Alignment problems can be efficiently solved with Dynamic Programming.
- Many Local Alignment problems may be similar in nature. Share similar structure, or even have identical regions where the genes are the same.

Existing Approaches

- SIMD (Tyler and Jen)
- Static Work Allocation, MPI-type parallelism

Existing Approaches

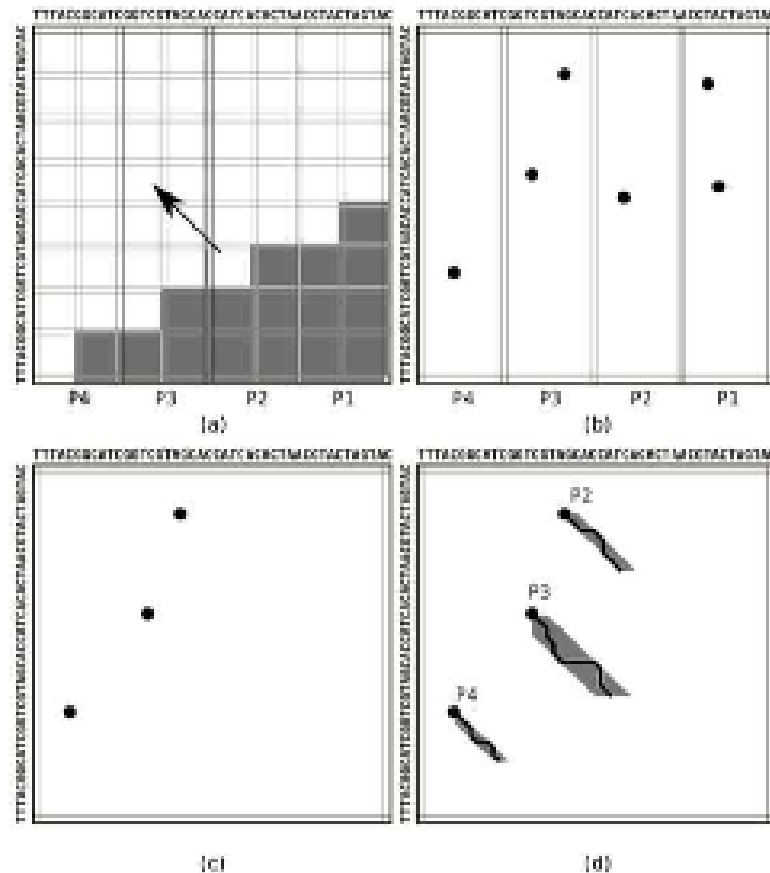


Fig. 5. General view of the Z-align strategy.

Our Goals

- We want operate a server for Local Alignment problems.
- ... that will solve many problems over time.
- We want to reuse solutions to previous problems to help solve current ones.
- We want to run many of these problems concurrently.

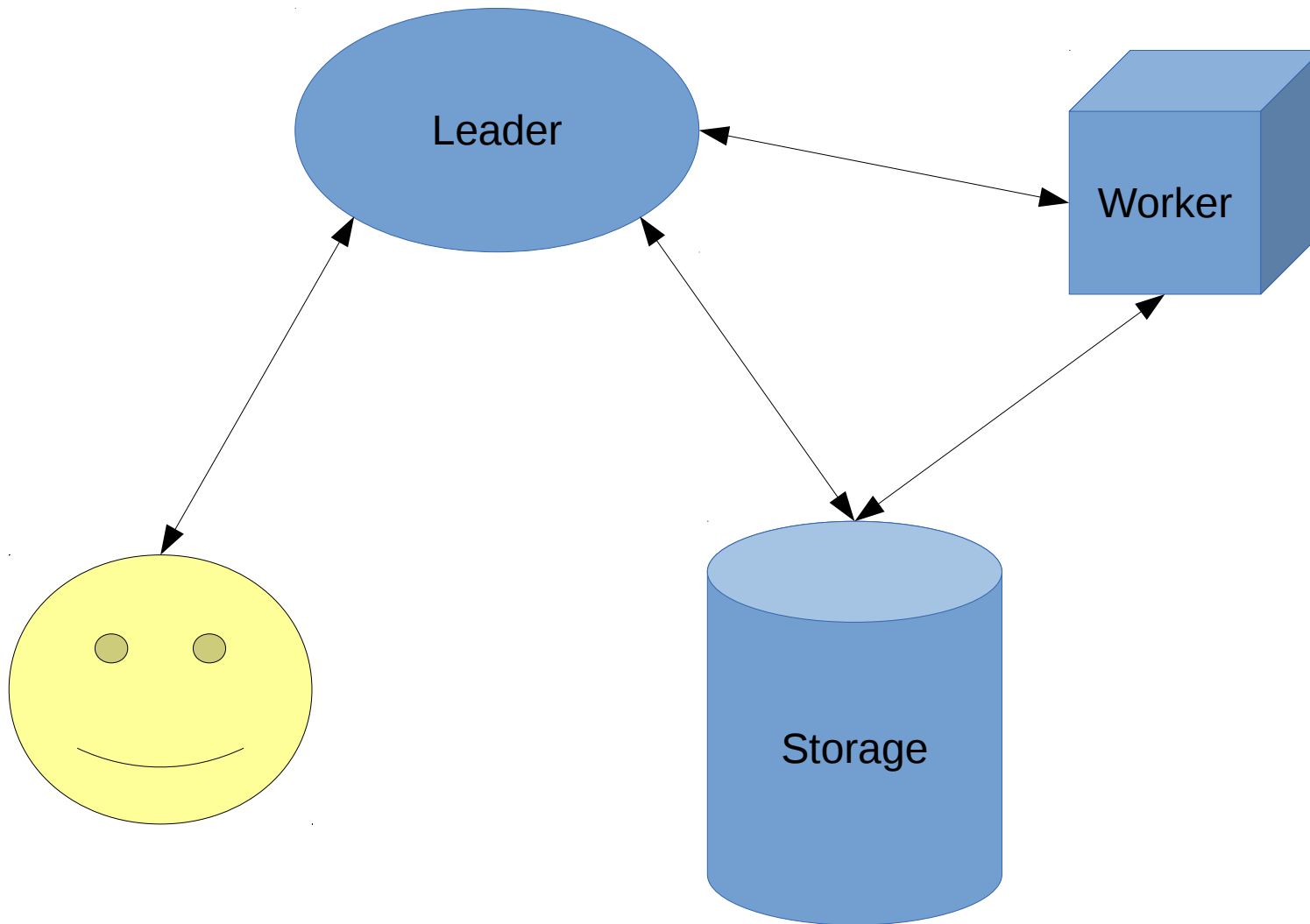
Our Approach - Architecture

- Client
- Leader
 - Allocates work and puts together solutions.
- Worker
 - Computes the matrix.
- Storage
 - Stores answers and problems.

Leader Cont.

- Advertises a list of problems to the workers
- Workers attempt to claim problems
 - Leader rules on the claim
- Upon completion, worker sends a “certificate” to the leader
- Leader informs the client of the solution

Here is a System Diagram



What's a Problem?

		A	C	G	T	A	C	G
	1	2	3	4	5	6	7	8
A	2							
C	3							
G	4							
T	5							
A	6							
C	7							
G	8							

Answer is $O(n^2)$ in the size of the problem

What's a Problem?

A C G T G C G							
1 2 3 4 5 6 7 8							
A	2						
C	3						
G	4						
T	5						
A	6						
C	7						
G	8						

Issues with our Approach

- We have to be faster than naively computing the whole matrix (on average).
- Message cost overhead. The cache hits have to be worth it.
 - Network latency
- Computing cache hits themselves have to be worth it.
- Total data size.

Features yet to Come

- Partial matching
- Subdivision
 - Intelligent subdivision
 - Genome metadata
- Distributed storage
 - Local RAM cache on the Worker.
 - Intelligent problem choices by the Worker.
 - PostgreSQL, Cassandra, Mongo, or similar