Outline

Context
Software
Benchmarks & results
Return on experience
Perspectives
Problem
Blood flow is vital for feeding and cleansing neurons

Recent discovery
Blood flow decrease before measurable cognitive decline in Alzheimer’s Disease (2016)

Novel therapeutic strategy
development of drugs targeting the vascular system

Idea
Use HPC simulation of cerebral blood flow to accelerate vascular targeted drug development
Blood flow study

- Vascular networks have a complex topology
- Blood itself is a complex fluid
Blood flow simulations

Previous Simulations (using V1.0)
14k vessels

Issue
Does not scale up for large vascular networks

PhD IMFT Maxime Berg, Berg et al. 2020

1 mm³ mouse brain
Whole mouse brain imaging

International effort to acquire whole mouse brain vascular networks

2020: publication of a pipeline reconstructing the full vascular network of a mouse brain at *Institut du Cerveau et de la Moelle épinière (ICM)*
Whole mouse brain imaging

Example: Lightsheet microscopy

Context | Software | Result | Experience | Perspectives

visikol.com

Kirst et al. 2020
Simulation goals:

- 5M to 10M vessels
- Pressure and blood flow
- Distribution of red blood cells
- Nutrient and drug delivery
Flow solving

Electrical analogy
- $\Delta U = RI$
- $\Delta P = RQ$

Matrix formulation for full brain
- Inversion of a $\sim 5M \times 5M$ sparse square matrix $M$
- Requires gradient descent like algorithm
- Requires parallel calculation

$Mp = y$
History

- 2014: V1 start
  - Research code in C++
  - Not scalable
  - Completely written on top of PETSc
- 2018: ERC POC grant
  - Scaling for full brain
  - Code industrialization
- 2019: V2 start
  - Hiring dedicated software engineer
  - Codebase mostly rewritten
- 2020: first tests on full brain data (Kirst et al.)
The VITAE software

Design goals

- Fully parallel processing
- CPU efficient
- Modern C++
- Modular API
- User friendly
- Fully documented

Dependencies

- std=c++14 (gcc9, ICC 18 & 19)
- Compilation (cmake-3.10)
- MPI (OpenMPI-2.1.1, MPICH-3.3.2)
- Parallel IO (HDF5-1.10.5)
- Linear algebra (PETSc-3.7.7)
- Graph partitioning (Parmetis-4.0.3)
- JSON Config (nlohman-json-3.7)
- JSON Schema #7 (pboettch-2.0)
- Unit tests (cxxtest-4.4)
Data model

Vascular networks have a graph structure

Three major categories
- **vertex** for bifurcations
- **edge** for vessels
- **point** for vessels path (grouped by edge)

<table>
<thead>
<tr>
<th>Network</th>
<th>Size [vertex]</th>
<th>Memory footprint estimation</th>
</tr>
</thead>
<tbody>
<tr>
<td>[V1] Kleinfeld (mouse)</td>
<td>10 k</td>
<td>11.7 MiB</td>
</tr>
<tr>
<td>[V2] Average mouse brain</td>
<td>5 M</td>
<td>6.1 GiB</td>
</tr>
<tr>
<td>Average Human brain</td>
<td>&gt; 1 G</td>
<td>&gt; 1.2 TiB</td>
</tr>
</tbody>
</table>
Data flow

- **Anatomical Data**
- **Parallel IO**
- **Unit Conversions**
- **Parallel Graph Partitionner**
- **Data Reordering**
- **Replica Builder**
- **Solver**

**MPI communication dominant**
- Run once

**CPU dominant**
- 1 to many iterations
Graph Partitioning

Initial graph

Partitionning
minimize nb of edge cuts

Replica insertion
(ghosts)

process0

process1

...
Efficiency of graph partitionning

Context | Software | Result | Experience | Perspectives

Benchmarks

Lower is better

\[ \text{n.cut} \sim O\left( N_d^{1/3} \times N_v^{2/3} \right) \]

3-regular Synthetic network

1.8M vertex network (Kirst 2020 CU)
Benchmarks

MPI CPU scaling for Flow Solver

<table>
<thead>
<tr>
<th>Context</th>
<th>Software</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Benchmark</td>
<td>Flow calculation</td>
<td>CPU dominated</td>
</tr>
<tr>
<td></td>
<td>File system access</td>
<td>run once</td>
</tr>
<tr>
<td></td>
<td>Partitionning and pre/post processing</td>
<td>run once</td>
</tr>
<tr>
<td></td>
<td>Flow calculation</td>
<td>CPU dominated</td>
</tr>
<tr>
<td></td>
<td></td>
<td>run many</td>
</tr>
</tbody>
</table>

Lower is better
Benchmarks

MPI peak memory scaling for 1 process

Profiling tool: `valgrind --tool=massif`

Context | Software | Result | Experience | Perspectives

Lower is better

Memory excess
Dominated by Parmetis MPI buffers

Minimum Process size
Results

Full brain pressure and flow rate calculation

- Half mouse brain
- Data source:
  - ICM (Kirst et al. 2020 CU)
  - HDF5 File size: 507MB
  - Nb vertices: 3.4M
  - Nb edges: 4.7M
- Calculation
  - Olympe@CALMIP
  - ~10 minutes
  - one process
- Visualization
  - Avizo

Results Table

<table>
<thead>
<tr>
<th>Context</th>
<th>Software</th>
<th>Result</th>
<th>Experience</th>
<th>Perspectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>PhD Marion Giraud IMFT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Issues

Technical

- PETSC installation & config
- Direct use of HDF5 instead of PETSc wrapping
- MPI is not very well suited for graph manipulation
  - Graph structure requires to call various mixtures of `Mpi_Alltoall()` and `Mpi_Alltoallv()`
  - Average nb of connected domains is 13 which may require specific optimizations
- Intel ICC18 compiler (std::shared_ptr<std::array>, etc.), ICC19 easier
- CMake steep learning curve, one has to use modern version
- C++ version of dependencies often not available or incomplete
  Example: HDF5, PETSc, MPI, etc.

Other

- Semantic
  - Example: disambiguate UIDs versus global offset and local offset
Solver teamwork

1) Scientific goal
   Mesh definition

2) Numerical scheme
   Algorithm

3) Code Specifications
   Data, Object Model

Write time: about 2/3 days
Software quality

Continuous integration
- Git versionning
- Wiki
- Runners
- unit tests

Naming convention document

Schema validation

Documentation
- Latex manual
- Rich doxygen code snippets
Scheduled Projects

- **Solver: BrainPulse**
  - Solver for blood pulsatility
  - Project lead: Alexandra VALLET (univ. oslo)
- **Solver: BioGrow**
  - Simulation of bacterial growth
  - Project lead: Jean-Daniel Julien
- **Solver: PhaseSeparation**
  - Calculation of blood cells concentration in plasma
  - Initial author: Maxime Berg
  - Porting to VITAE API V2: Maxime Pigou
- **Solver: MassTransport**
  - Simulation of solute transport into blood
  - Initial author: Maxime Berg
  - Porting to VITAE API V2: Alexandre Sauvé
Evolutions

Technical

- Improve flow solver convergency time (*PETSc* KSP methods)
- Improve partitioning scaling
  (Example: test *PT-SCOTCH*)

Data exploitation

- New vascular networks from various collaborations
- Reduction of full brain results
Questions