UNIVERSITÉ DE GENÈVE

FACULTY OF SCIENCE
Department of Informatics

Platelets GPU Team

University of Geneva

Members: Christophe Charpilloz*, Christos Kotsalos*, Pierre Künzli*

Mentors: Shoshana Jakobovits**, Marcel Schoengens**

Principal Investigators: Bastien Chopard*, Ritabrata Dutta***

*University of Geneva
** CSCS
*** University of Warwick

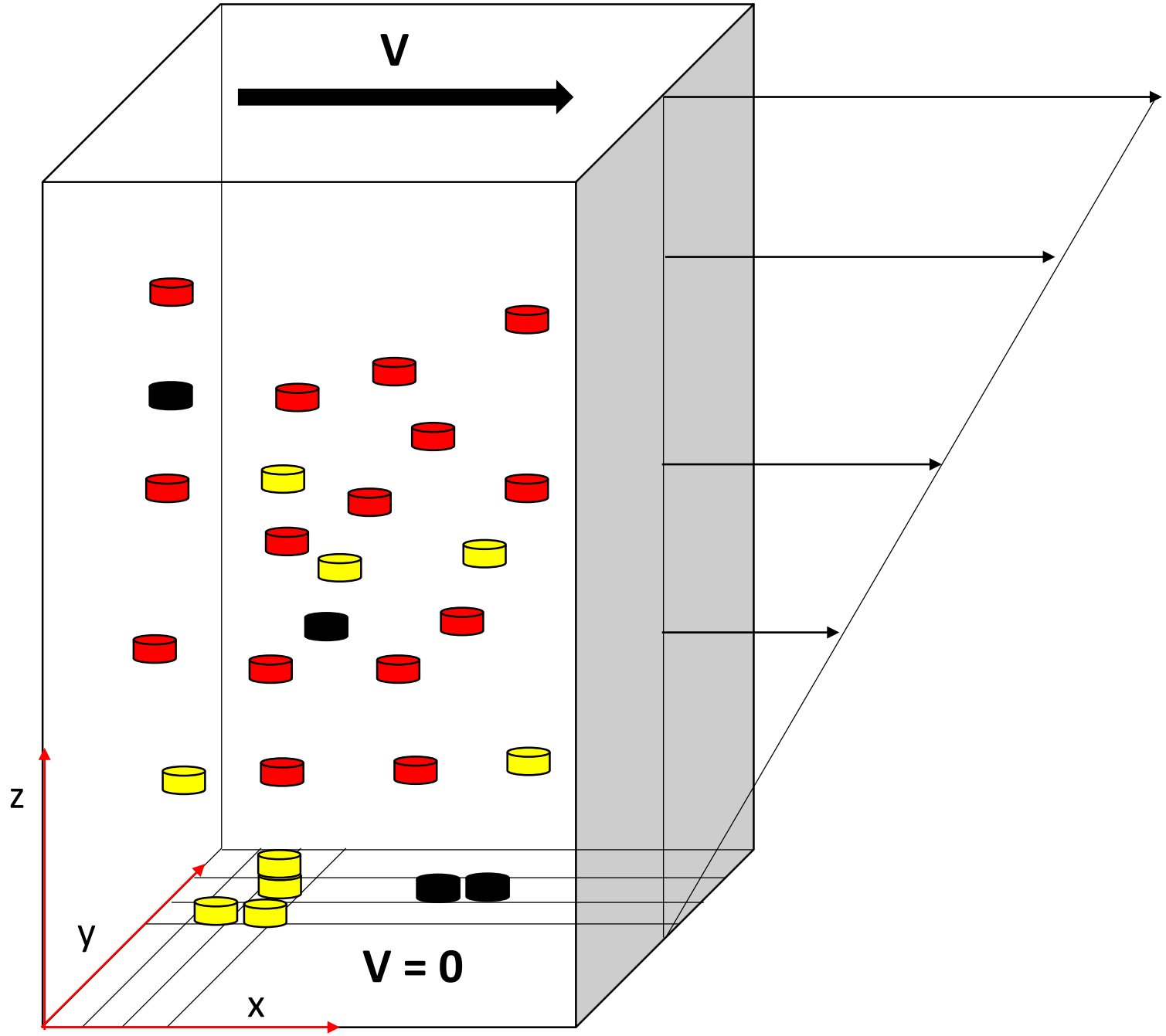
Model

- Red Blood Cell
- Platelet
  (Activated/ Non-Activated)
- Albumine

XY Plane
Homogeneous Field

Z-Direction: 1D
Diffusion equation

XY-Plane: 2D
Stochastic Problem



Pseudocode – **Sequential**

C++ code

```
while (t < tmax )
    Solve(1D Diffusion Eqn)

    2D Stochastic Part:
      Solve(Platelets Adhesion)
      Solve(Albumine Adhesion)
      Solve(Platelets Aggregation)

    if ( checkpoint)
        count Clusters/ Aggregates
```

Pseudocode – **Parallel**

CUDA C++

```
RNG (Curand Host API, async)
while (t < tmax )
    Solve(1D Diffusion Eqn)

    2D Stochastic Part:
      Solve(Platelets Adhesion)
      Memcpy: GPU2CPU (1 int) - Sync
      RNG (Curand Host API, async)
      Solve(Albumine Adhesion)
      Memcpy: GPU2CPU (1 real) - Sync
      Solve(Platelets Aggregation)
      Memcpy: GPU2CPU (1 int) - Sync
      RNG (Curand Host API, async)

    if ( checkpoint)
        count Clusters/ Aggregates
```

Goals

- ✓ Parallelize the 2D Stochastic Problem using CUDA
- ✓ Generate efficiently Random Number using CuRand
- ✗ Run multiple instances of the problem in order to explore the parameter space
- ✓ Desired SpeedUp : **x36** as we could execute monothread simulations on an XC40
  - ✓ Speedup achieved on one P100: **x45**
  - ✓ GPU competes in terms of node utilization, 25% efficiency gain
  - ✓ 5min 45sec monothread → 7.6 sec GPU version

Machine used for monothread: Piz Daint - XC40 Compute Nodes, specs Two Intel® Xeon® E5-2695 v4 @ 2.10GHz (2 x 18 cores, 64/128 GB RAM)

GPU used for parallel: NVIDIA® Tesla® P100 16GB

Methods

Failure

- Unified memory
- Kernel collapse
- 1D LBM on GPU
- Thrust reduction instead of atomic add

Success

- 2D deposition/aggregation solving on GPU
- CUDA streams, asynchronous RNG and simulation
- Home made reduction instead atomic add