Keldysh Institute of Applied Mathematics (Russian Academy of Sciences)

Plekhanov Russian University of Economics

Implementing the Graph Model of the Spread of a Pandemic on GPUs

Vladimir Sudakov – Leader Researcher, Head of the Applied Modeling Laboratory, Doctor of Technical Sciences

Nikita Yashin - Assistant Researcher of the Applied Modeling Laboratory

The study was funded by RFBR and CNPq, FASIE, DBT, DST, MOST, NSFC, SAMRC according to the research project No. 20-51-80002

Problem definition

Input:

- Probability on infection
- Everyday behavior of people

Output:

- The number of people in the states "Susceptible - Exposed - Infectious – Recovered" at each time
- Assessment of the economic consequences of the situation

Problem:

- How to change the everyday behavior of people in order to reduce the number of infections and not lead to significant economic consequences?

Approaches to the solution

Macro models

- SIR model
- SIRD model
- SEIR model

Micro models

- Discrete Event Simulation model
- Agent-based model

Micro models

They allow the assessment of structural changes.

But they work very slowly with a large number of people.

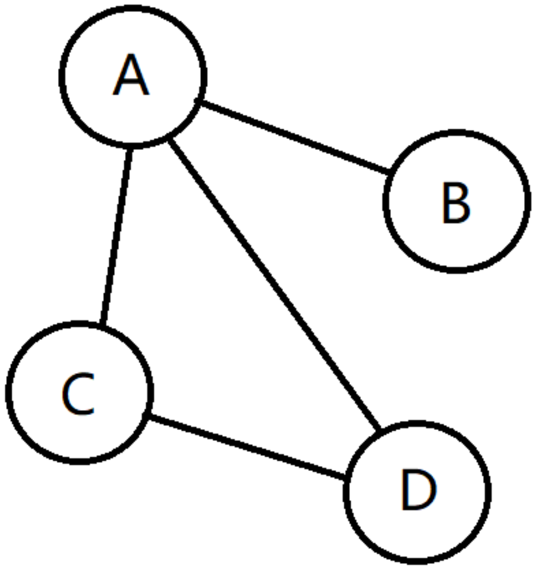
Parallel Discrete Event Simulation model - Conflict resolution is required when scheduling events and later events are already being processed.

Solution

Building micromodels on small populations and their further study and scaling on graph models

- A set of graphs for common interactions are defined.
  - Separate graphs of contacts are built between family members, relatives, relatives, friends, work colleagues.
  - Separate graphs for random interactions in public places.
- Processing of graphs on the GPU
  - Union of graphs.
  - Determination of random interactions by the Monte Carlo method.
  - Rebuilding graphs of some types of interactions

Implementing micro model of interaction using graph model



	A	B	C	D
A		1	1	1
B	1		0	0
C	1	0		1
D	1	0	1	

Development tools

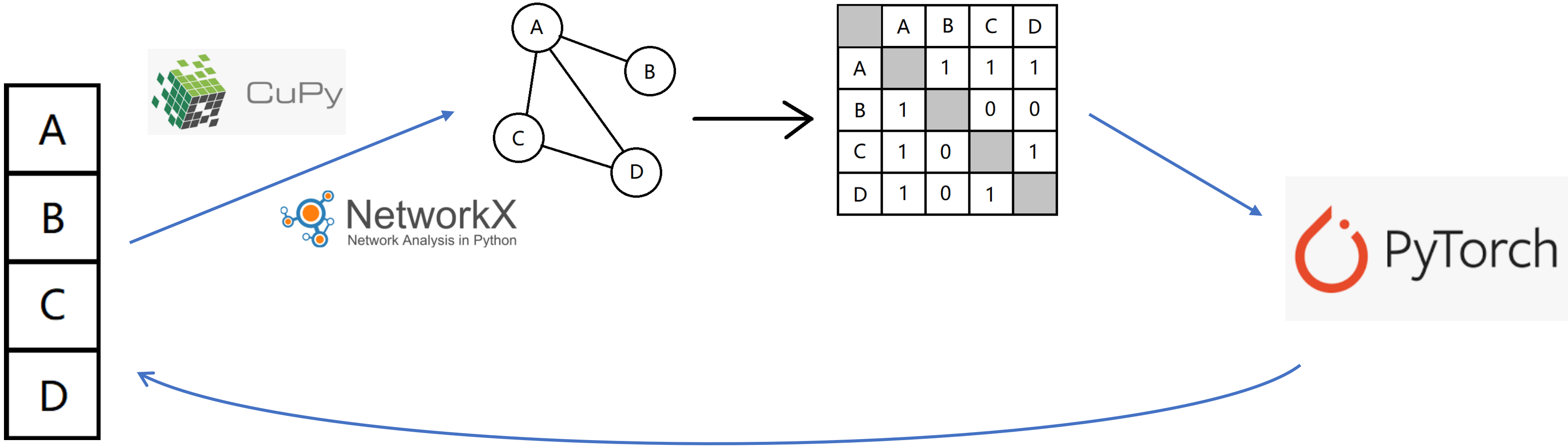
python™

NetworkX
Network Analysis in Python

CuPy

PyTorch

Algorithm of implementing virus spreading with graph



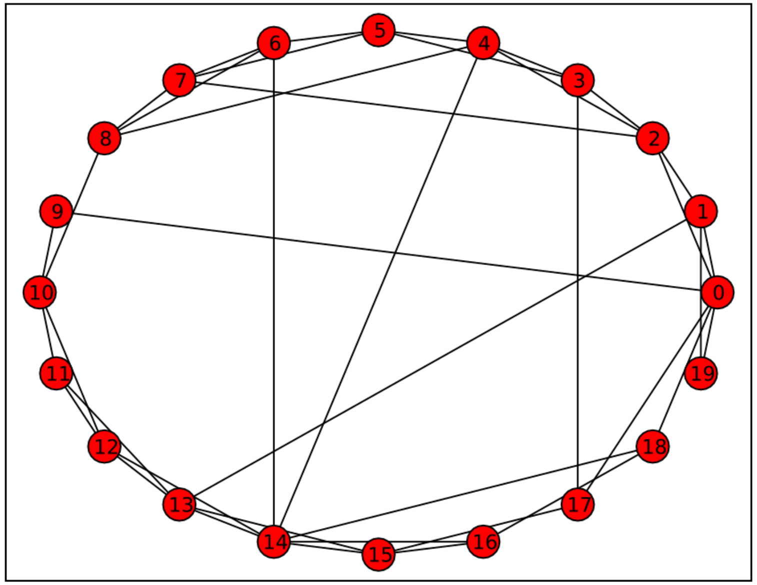
	A	B	C	D
A		1	1	1
B	1		0	0
C	1	0		1
D	1	0	1	

Building graph for constant interactions

NetworkX
Network Analysis in Python

watts_strogatz_graph

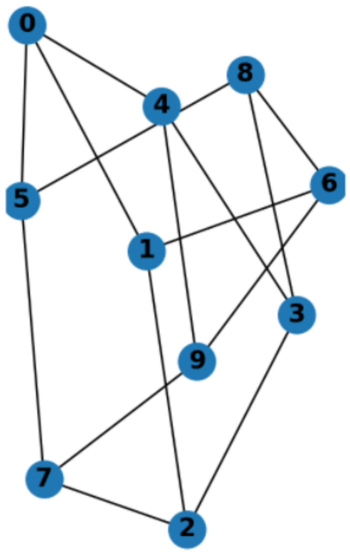
Watts-Strogatz model N=20, K=4, β=0.2



Building graph for dynamic interactions without using GPU

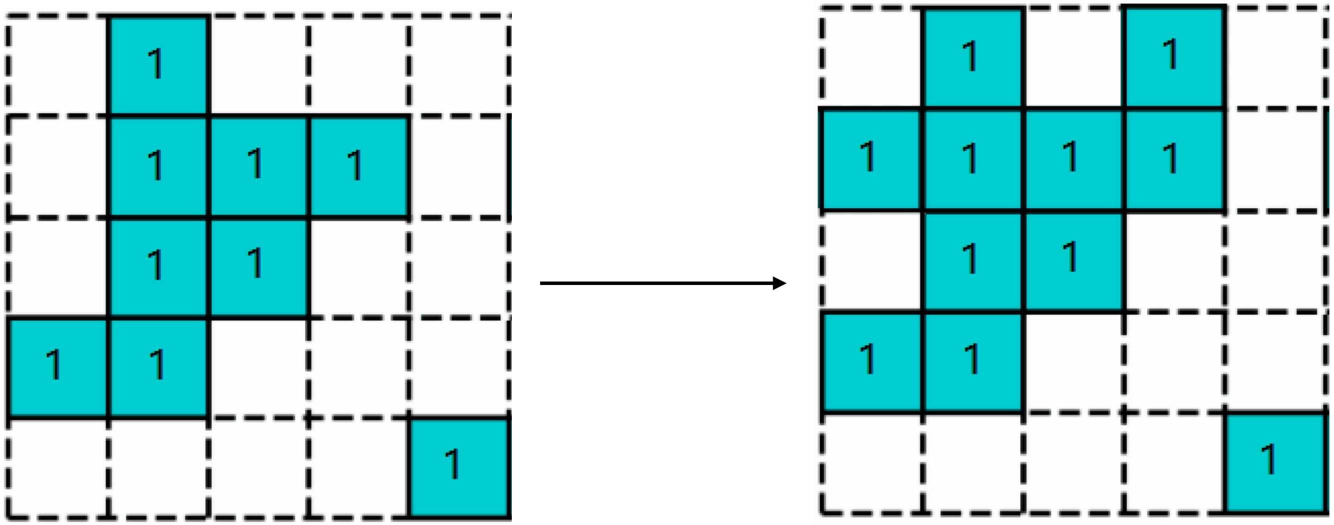
NetworkX
Network Analysis in Python

fast_gnp_random_graph

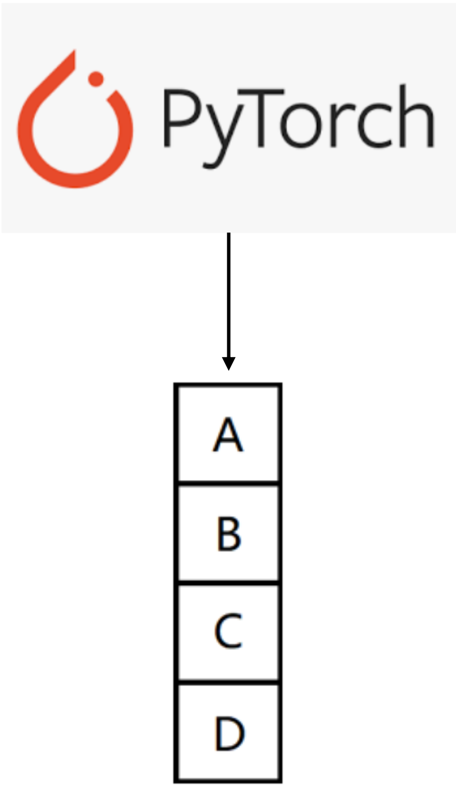


Building graph for dynamic interactions using GPU

sparse.rand



Adjacency matrix traversal using PyTorch




```
# CPU:
state = torch.zeros(pop)

# GPU:
cuda = torch.device('cuda')
state = torch.zeros(pop, device=cuda)
```

Functions to compare efficiency

no_cuda

CPU + CPU

torch_cuda

CPU + GPU

graph_cuda

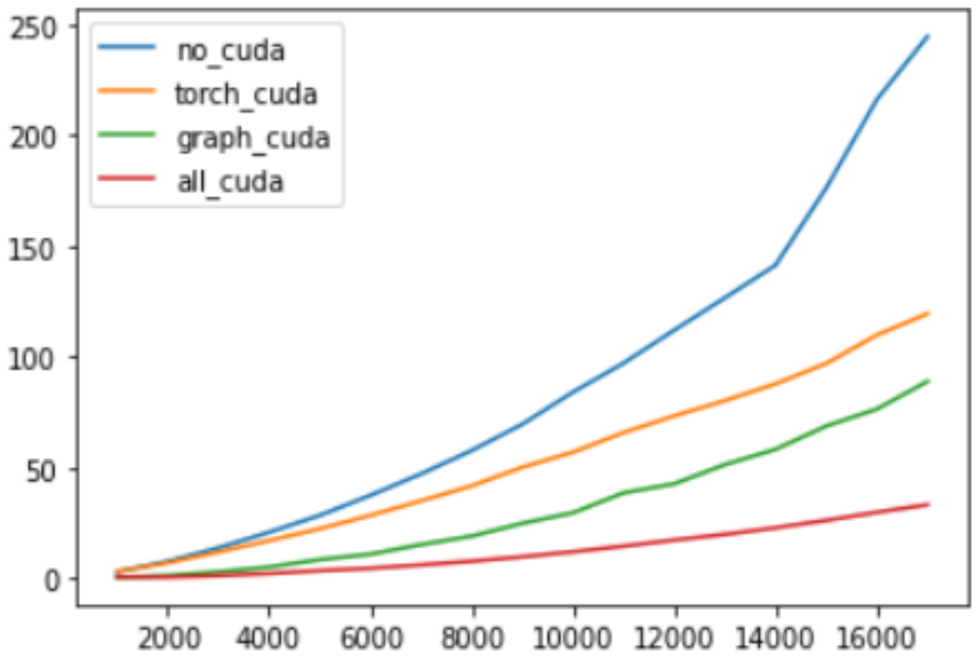
GPU + CPU

all_cuda

GPU + GPU

Results

	no_cuda	torch_cuda	graph_cuda	all_cuda
1000	3.200957	3.351706	0.710170	0.912640
2000	7.914960	7.242812	1.510238	0.848995
3000	13.984233	12.124245	3.181422	1.496793
4000	21.029639	17.249926	5.338851	2.455780
5000	28.600931	22.572831	8.708607	3.728163
6000	37.634567	28.611021	11.137490	4.775039
7000	47.314849	35.154100	15.447932	6.286974
8000	57.854738	41.903814	19.260485	7.980263
9000	69.675968	50.306944	25.033169	9.963441
10000	84.130254	57.087363	29.785555	12.199988
11000	97.206411	65.925612	38.764165	14.762600
12000	112.067288	73.430345	42.834090	17.492042
13000	126.610486	80.314935	51.497435	20.050080
14000	141.328987	87.860316	58.368204	23.021266
15000	176.068139	97.052003	68.845642	26.358311
16000	215.929255	109.848109	76.498892	29.992050
17000	244.363839	119.475280	88.965665	33.416545



1000	3.507361
2000	9.322746
3000	9.342799
4000	8.563324
5000	7.671588
6000	7.881520
7000	7.525854
8000	7.249728
9000	6.993163
10000	6.895929
11000	6.584640
12000	6.406758
13000	6.314712
14000	6.139062
15000	6.679796
16000	7.199550
17000	7.312660

7.152423040425422

Conclusion

- The use of the matrix form of representation of the "small world" graphs made it possible to simulate the spread of pandemics
- Required matrix computations have been implemented using libraries that allow efficient use of GPUs
- Achieved 7-fold increase in computing speed for a population of 17 thousand people
- The constructed model makes it possible to analyze the influence of management decisions, for example, a lockdown on the rate of spread of pandemics.

Prospects

Write as: sudakov@ws-dss.com

Thanks for attention!