M. Skrodzki,
U. Reitebuch,
K. Polthier
FU Berlin

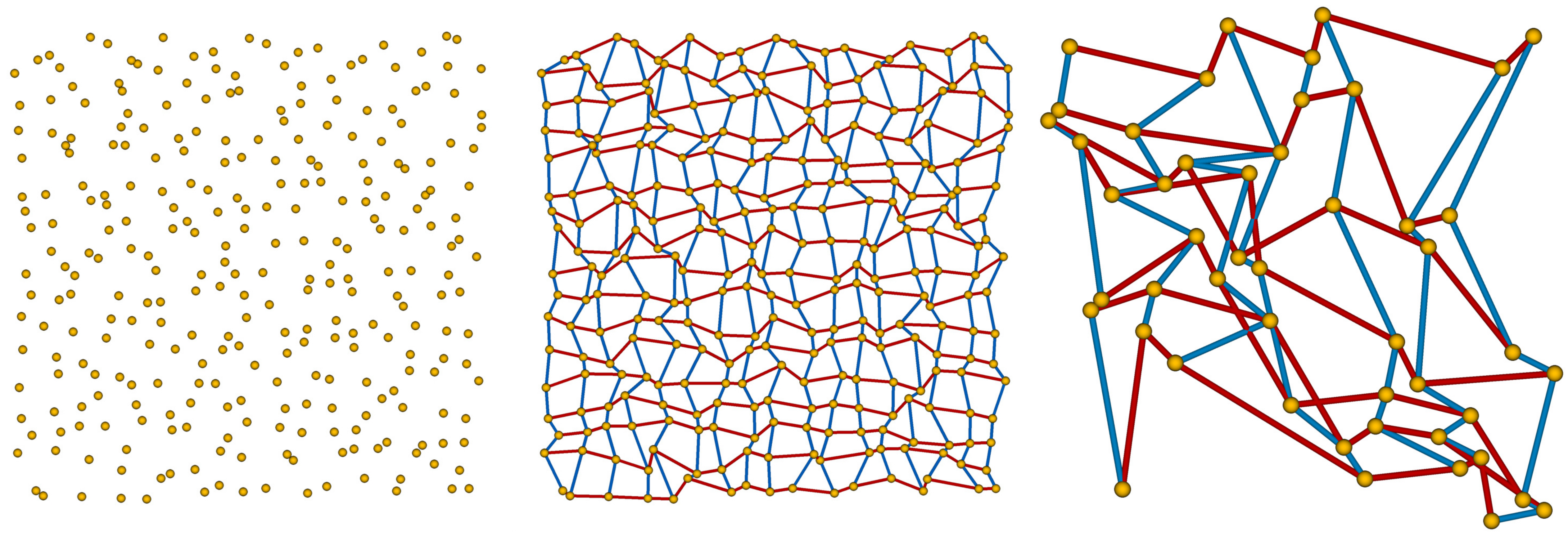
Computational and Structural Aspects of Point Set Surfaces

The Neighborhood Grid

The Neighborhood Grid approximates neighborhood information. A (quadratic) matrix contains the coordinates of the points such that in each row the x-values are increasing while in each column the y-values are increasing.

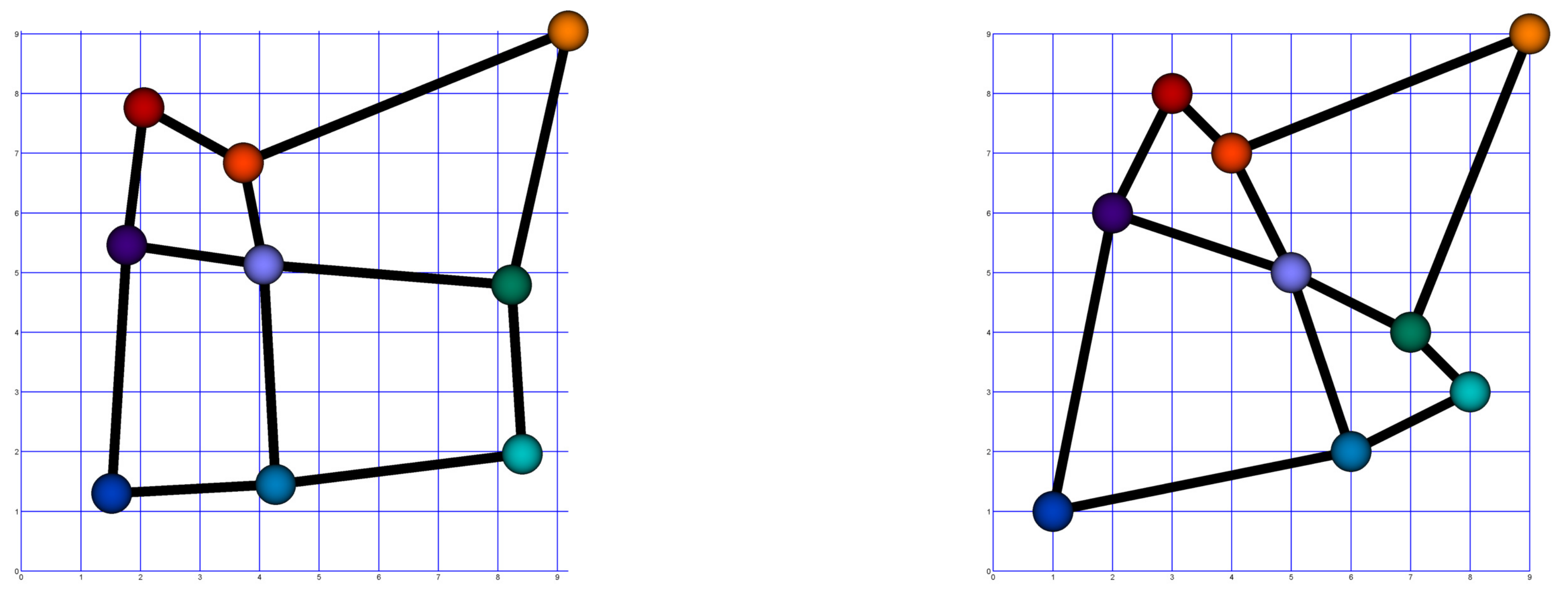
For the algorithm, the order of the points suffices, the exact coordinates are irrelevant. If the above ordering is given, we call it a "stable state".

Illustrating the Neighborhood Grid



From left to right: A raw point cloud, the corresponding structure induced by the grid, and an example where the neighborhood is not faithfully recovered.

Order Preservence



(2,06; 7,76)	(3,73; 6,84)	(9,18; 9,05)
(1,77; 5,46)	(4,07; 5,13)	(8,23; 4,79)
(1,53; 1,30)	(4,27; 1,45)	(8,41; 1,96)

(3; 8)	(4; 7)	(9; 9)
(2; 6)	(5; 5)	(7; 4)
(1; 1)	(6; 2)	(8; 3)

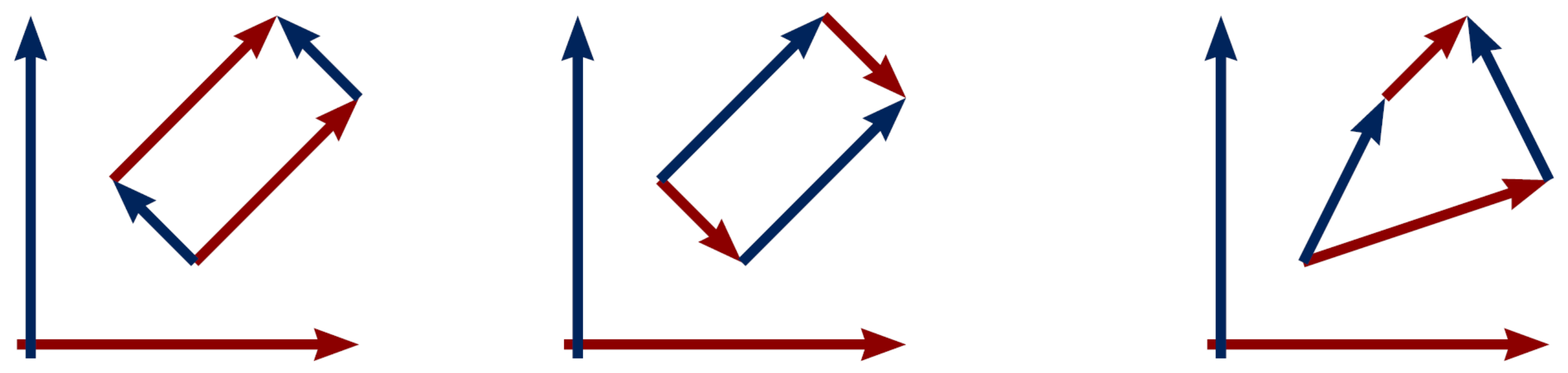
Reducing from rational to integer values keeping the order.

Energy / Uniqueness of Stable States

Summing x-values with column-numbers and y-values with row-numbers, the resulting energy grows when exchanging wrongly sorted pairs.

$$\sum_i i x_i + \sum_j j y_j$$

However, the resulting stable states do not have to be unique.



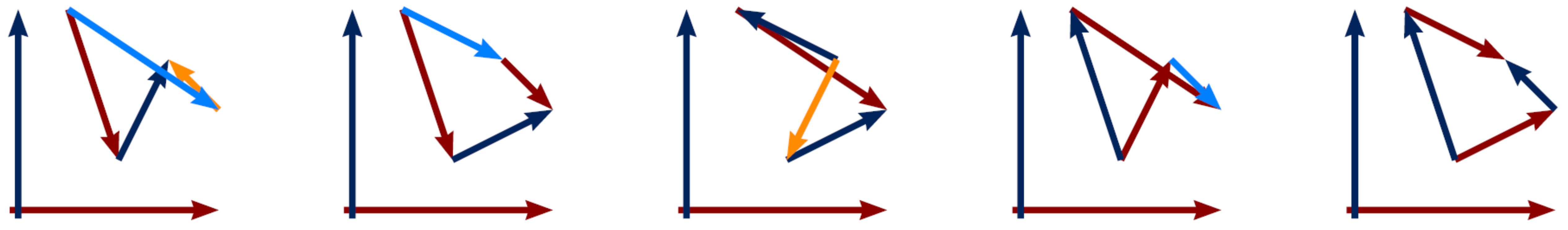
1 2	3 4
2 1	4 3

3 4	4 3
1 2	2 1

2 3	3 4
1 1	4 2

Two different stable states for one point configuration and a 2×2 configuration with a unique stable state.

Circulation of an Element after Exchaning



4 2	3 3
1 4	2 1

3 3	4 2
1 4	2 1

1 4	4 2
3 3	2 1

1 4	4 2
2 1	3 3

1 4	3 3
2 1	4 2

Analysis Difficulty

Violations of the sorting are marked in yellow or light blue. Repeatedly interchanging the element $(3, 3)$ along violated edges ultimately gives a stable state. But $(3, 3)$ circulates through the matrix resulting in a difficult runtime analysis.

Pro/Contra

Benefits:

- Easy to parallelize.
- Const. time approximation.

Disadvantages:

- No lower-dim. points.
- Approx. might be bad.

Results

size n	points n^2	point sets $(n^2)!$	stable states $((n^2)!/n!^n)^2$	unique	min	max	avg.
1	1	1	1	1	1	1	1.0
2	4	24	36	12	1	2	1.5
3	9	362,880	2,822,400	966	1	42	7.777
4	16	20,922,789,888,000	3,976,941,969,000,000	0	≤ 8	$\geq$24,024	190.077

Applications

- Crowd Simulation [1].
- Fluid Animation [2].
- Biological Cell Simulation [3].

References

[1] M. Joselli, E. B. Passos, M. Zamith, E. Clua, A. Montenegro, and B. Feijó. "A Neighborhood Grid Data Structure for Massive 3D Crowd Simulation on GPU", 2009.

[2] M. Joselli, J. R. da S. Junior, E. W. Clua, A. Montenegro, M. Lage, and P. Pagliosa. "Neighborhood grid: A novel data structure for fluids animation with GPU computing", 2015.

[3] M. de Geomensoro Malheiros and M. Walter. "Simple and Efficient Approximate Nearest Neighbor Search using Spatial Sorting", 2015.

Corresponding Author: Martin Skrodzki, Workgroup Geometry Processing, FU Berlin, Germany, martin.skrodzki@fu-berlin.de