A Portable Distributed Sparse Grid Density Estimation for Big Data Clustering

Dipl.-Inf. David Pfander, B.Sc. Gregor Daiß, Jun.-Prof. Dirk Pflüger
Simulation of Large Systems, IPVS, University of Stuttgart, Germany

Motivation

- Clustering in Big Data scenarios with millions to billions of data points requires specialized algorithms and efficient implementations
- Sparse grid clustering uses a spatial discretization scheme that is suitable for higher-dimensional problems
- For Big Data: linear complexity of density estimation in data points
- We present first scaling results on Piz Daint and show performance portability across accelerator architectures and processors

Sparse Grids Density Estimation

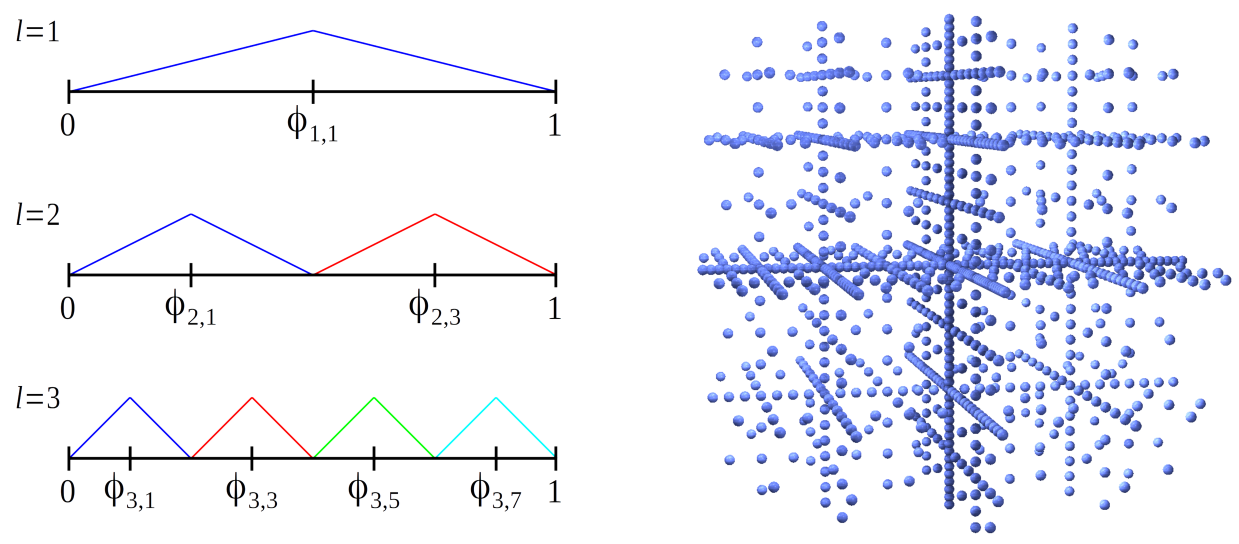
- Spatial discretization with sparse grids mitigates the curse of dimensionality (up to 166 dimensions In practice) [1, 2]
- With M data points and N grid points, the density estimation problem [3]

$$\tilde{f} = \arg\min_{u \in V} \int_{\Omega} (u(x) - f_{\epsilon}(x))^2 dx + \lambda \sum_{i=1}^{N} \alpha_i^2, \qquad f_{\epsilon} = \frac{1}{M} \sum_{i=1}^{M} \delta_{x_i}$$

with sparse grids function space $V = \mathrm{span}\{\phi_i : i \in I\}$ leads to the system of linear equations (SLE) for coefficients α_i

$$(B + \lambda I)\alpha = \vec{b}, \quad B_{ij} = (\phi_i, \phi_j)_{L^2}, \quad \vec{b} = \frac{1}{M} \sum_{j=1}^{M} \phi_i(x_j)$$

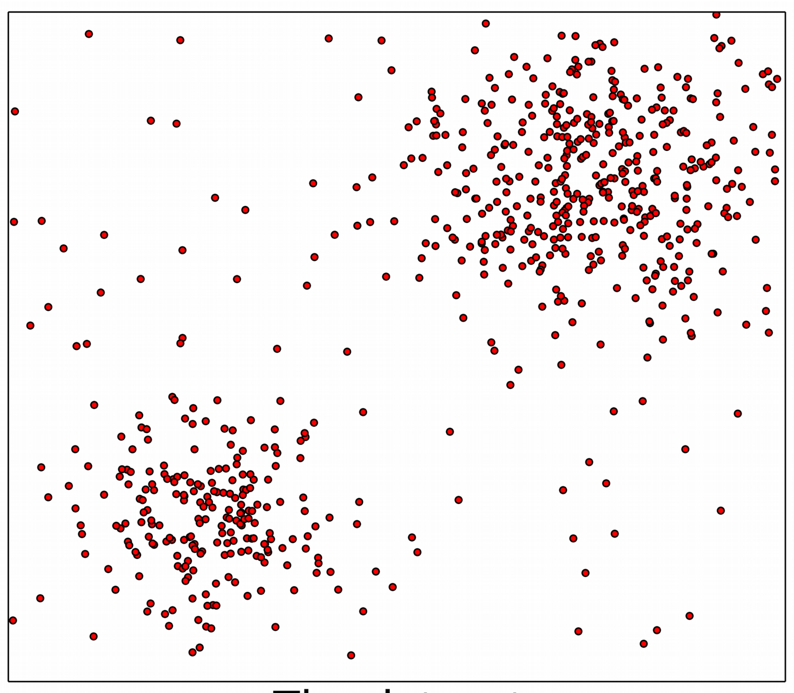
- ϕ_i are hat basis functions at grid points, enumerated through I

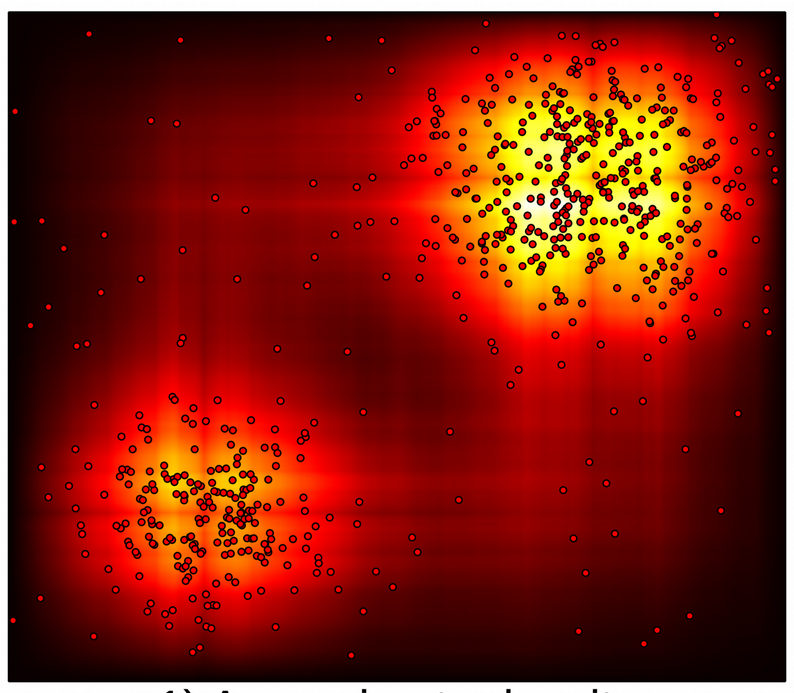


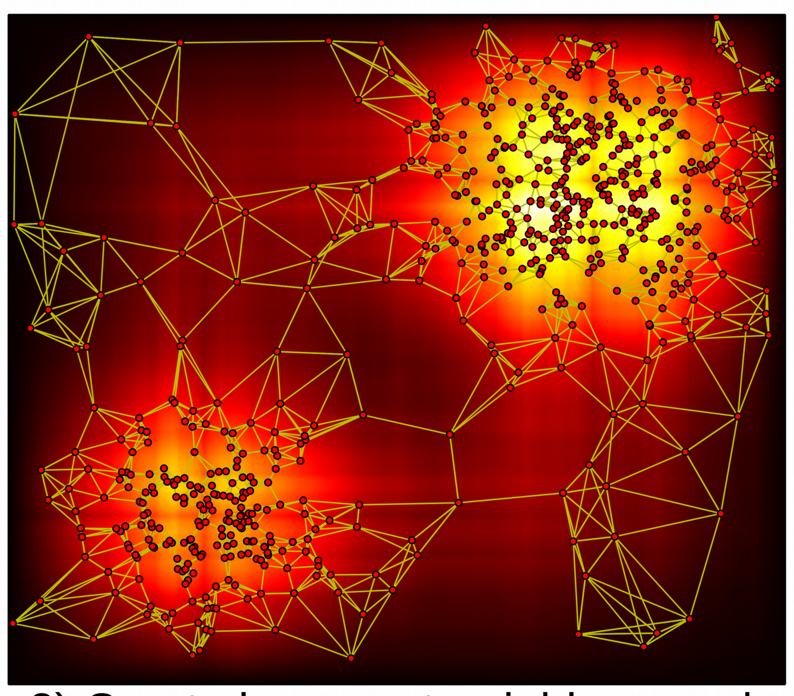
- Hierarchical 1d basis functions (left) and 3d grid (right)
- Linear dataset complexity: Dataset only used to calculate $\vec{b}$

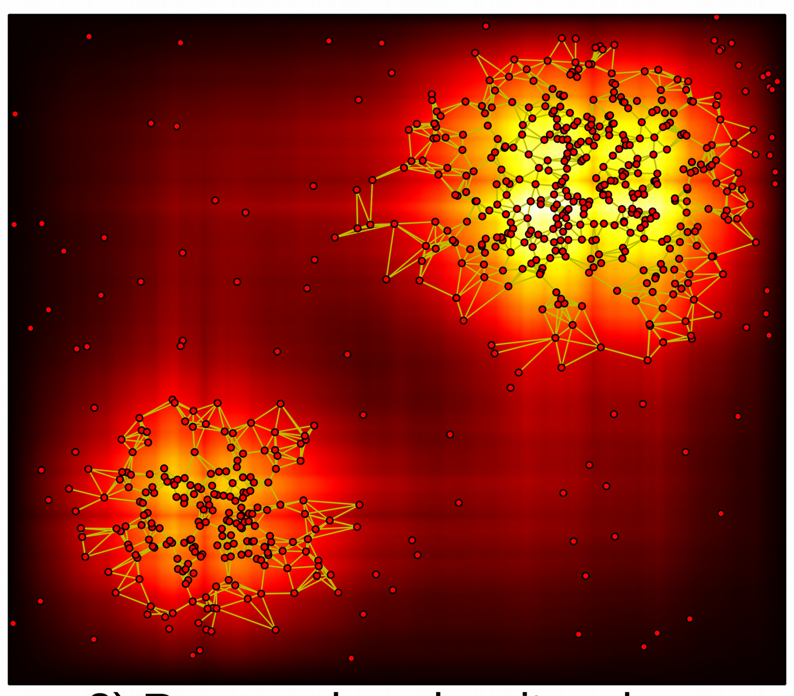
Clustering based on Sparse Grid Density Estimation

- Clustering algorithm proposed by Peherstorfer [4]:


The dataset


1) Approximate density


2) Create k-nearest-neighbor graph

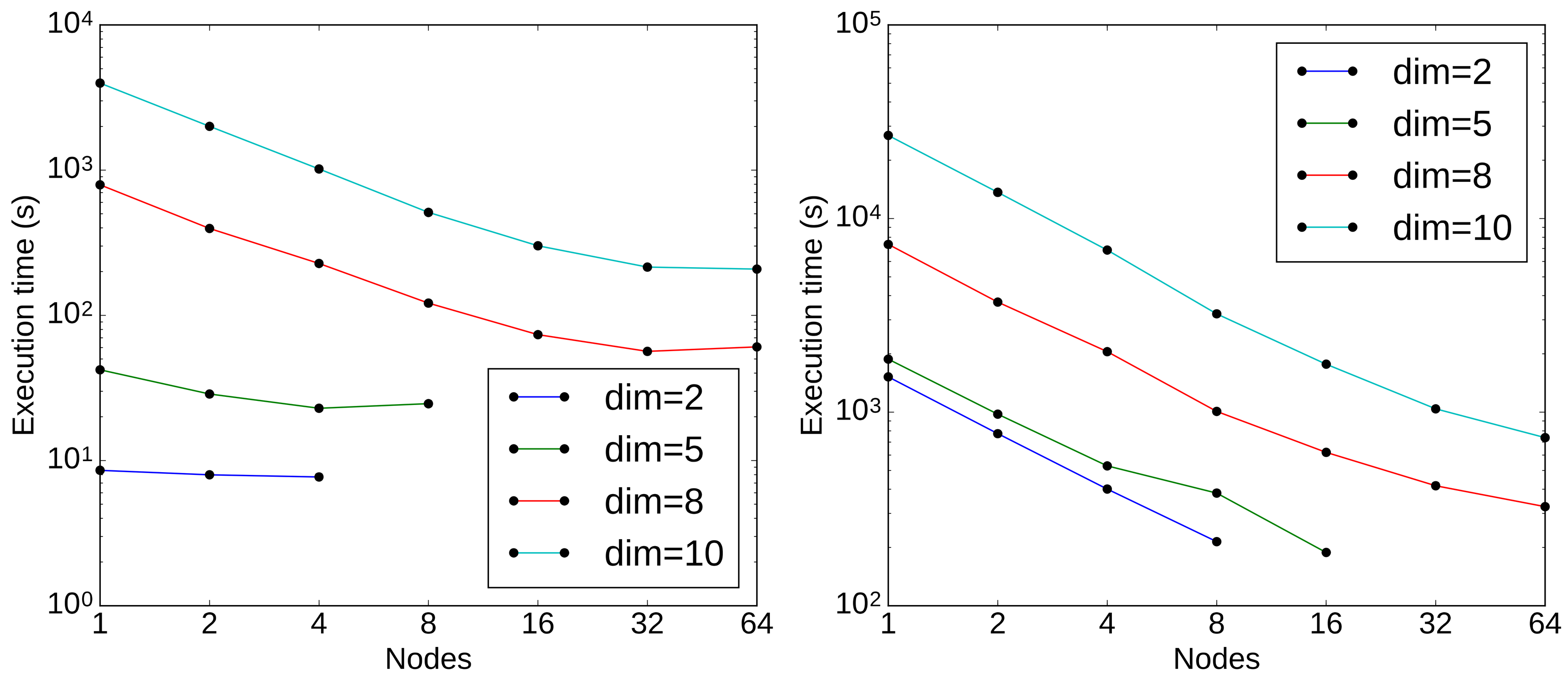

3) Remove low-density edges

- Search and return connected components as detected clusters

Implementation Properties

- System of linear equations solved via iterative solver (conjugate gradient), because matrix B is large: $M \times M$
- Necessitates a highly-optimized implementation of $\vec{w} := B\vec{v}$ and calculation of components $B_{ij} = (\phi_i, \phi_j)_{L^2}$
- Implemented in OpenCL (4 kernels) to enable portability
- Configurable optimizations (blocking, pipelining, local-memory, ...) and code generation for performance portability
- Grid and dataset stored as arrays of d-dimensional tuples
- Manager-worker scheme used for load balancing

Distributed Scaling



- Scenario: 1M data points (left), 10M data points (right), 10 clusters, level 8 regular sparse grid, 1.8M grid points in 10d, strong scaling
- On Cray XC50 Piz Daint, 1xTesla P100 per node
- Linear scaling as long as enough work is available per P100
- Linear complexity in M: observed for calculation of $\vec{b}$ (most expensive step) in 10d scenario with 2 nodes: 361s (1M) vs. 3635s (10M)

Performance Portability of Density Estimation

Device Name	8d (DP)	Peak Fraction	Peak Limit
Tesla P100	1.2TF	26%	39%
Tesla K20X	0.3TF	23%	35%
FirePro W8100	0.5TF	23%	33%
Xeon E7-8880v3, 4xSocket, 72C	1.1TF	50%	76%

- Scenario: 8 dim dataset with 1 million data points in 10 clusters, regular sparse grid with 580k grid points
- Instruction mix limits performance to 66% of peak (69% on FirePro)
- High register pressure limits performance on GPUs

Future Work

- Experiments with adaptive grids for dramatically reduced number of grid points
- Improved grid point encoding for reduced register pressure on GPUs

References

[1] H.-J. Bungartz and M. Griebel, Sparse Grids, Acta Numerica, 2004
[2] D. Pflüger, Spatially Adaptive Sparse Grids for High-Dimensional Problems, Verlag Dr. Hut, 2010
[3] Hegland, M. et al. Finite Element Thin Plate Splines In Density Estimation, ANZIAM Journal, 2000
[4] Peherstorfer, B. et al. Clustering Based on Density Estimation with Sparse Grids, KI 2012: Advances in Artificial Intelligence, 2012