Algorithms and Data Structures Group

Gerth Stølting Brodal

madalgo
CENTER FOR MASSIVE DATA ALGORITHMICS

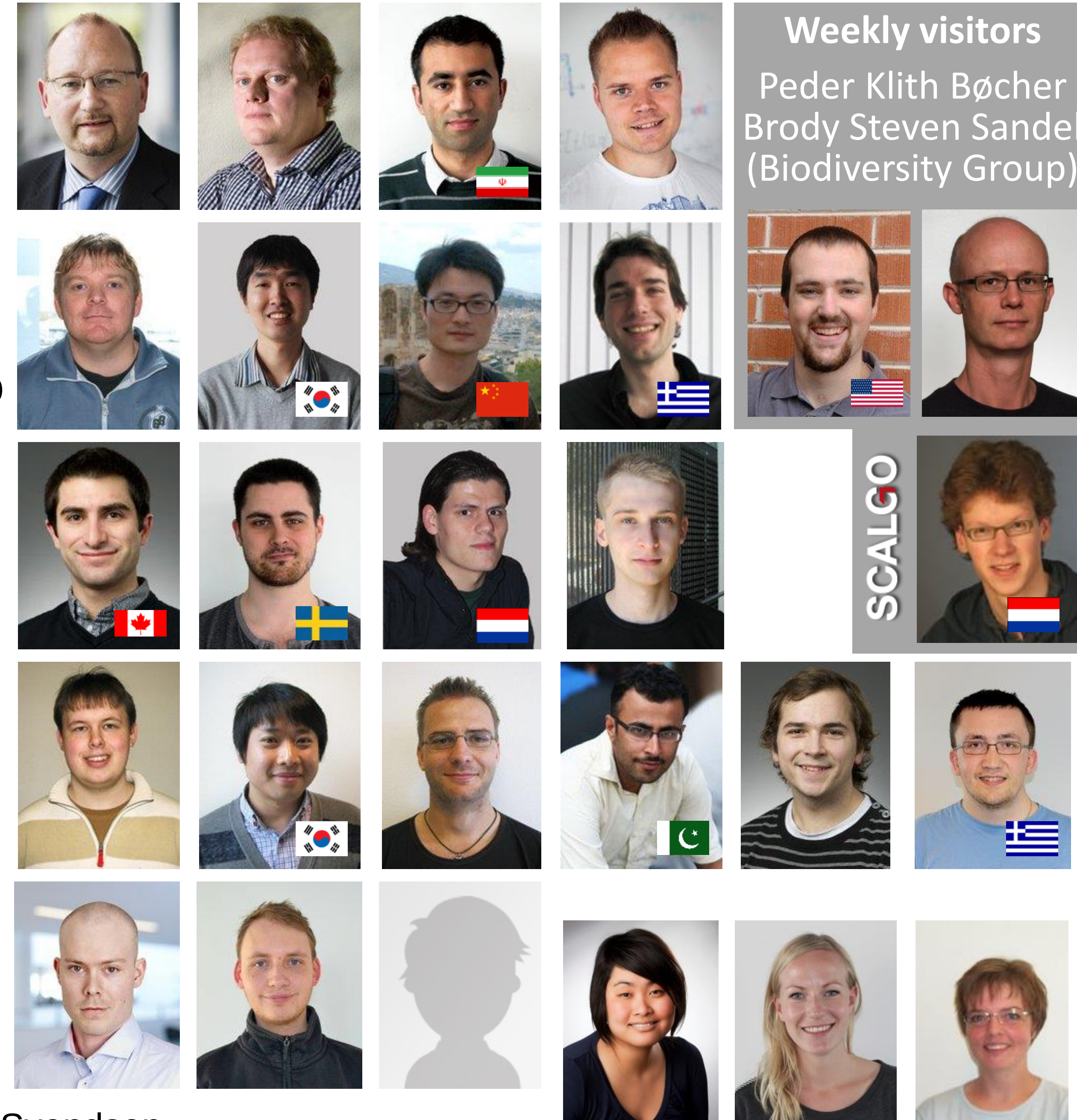
Faculty Meeting, Department of Computer Science, Aarhus University, October 3, 2014

VIP
Lars Arge (Professor)
Gerth Stølting Brodal (Lektor)
Peyman Afshani (Adjunkt)
Kasper Green Larsen (Adjunkt)

PostDocs
Allan Grønlund Jørgensen (HTF)
Wanbin Son
Zengfeng Huang
Constantinos Tsirogiannis (AUFF)

PhD students
Bryan Wilkinson
Edvin Berglin
Ingo van Duijn
Jakob Truelsen
Jesper Asbjørn Sindahl Nielsen
Jungwoo Yang
Morten Revsbæk
Sarfraz Raza
Mathias Rav
Konstantinos Mampentzidis

MSc students
Claus Jespersen
Mikkel Engelbrecht Hougaard
Bo Mortensen



Programmer Svend Christian Svendsen

Administration Trine Ji Holmgaard Jensen, Katrine Østerlund Rasmussen, Ellen Kjemtrup

Events

- Workshop on Massive Data Algorithmis (2009 -)
- Symposium on Computational Geometry (2009)
- European Symposium on Algorithms / ALGO 2016
- MADALGO Summer Schools 2007, 2008, 2010 -
- MADALGO retreat (October)

Research – Efficiency

$O(n^3)$

↓

$O(n^2)$

↓

$O(n \cdot \log^2 n)$

↓

$O(n \cdot \log n)$

↓ ?

$O(n \cdot \frac{\log n}{\log\log n})$

quartet distance between two trees

Less practical ?

Deeper insights

Improved asymptotics

More complicated ?

algorithm engineering ?

$O(\log n)$

↓

$O(\frac{\log n}{\log\log n})$

↓

$O(\log\log n)$

↓

$O(\sqrt{\log\log n})$

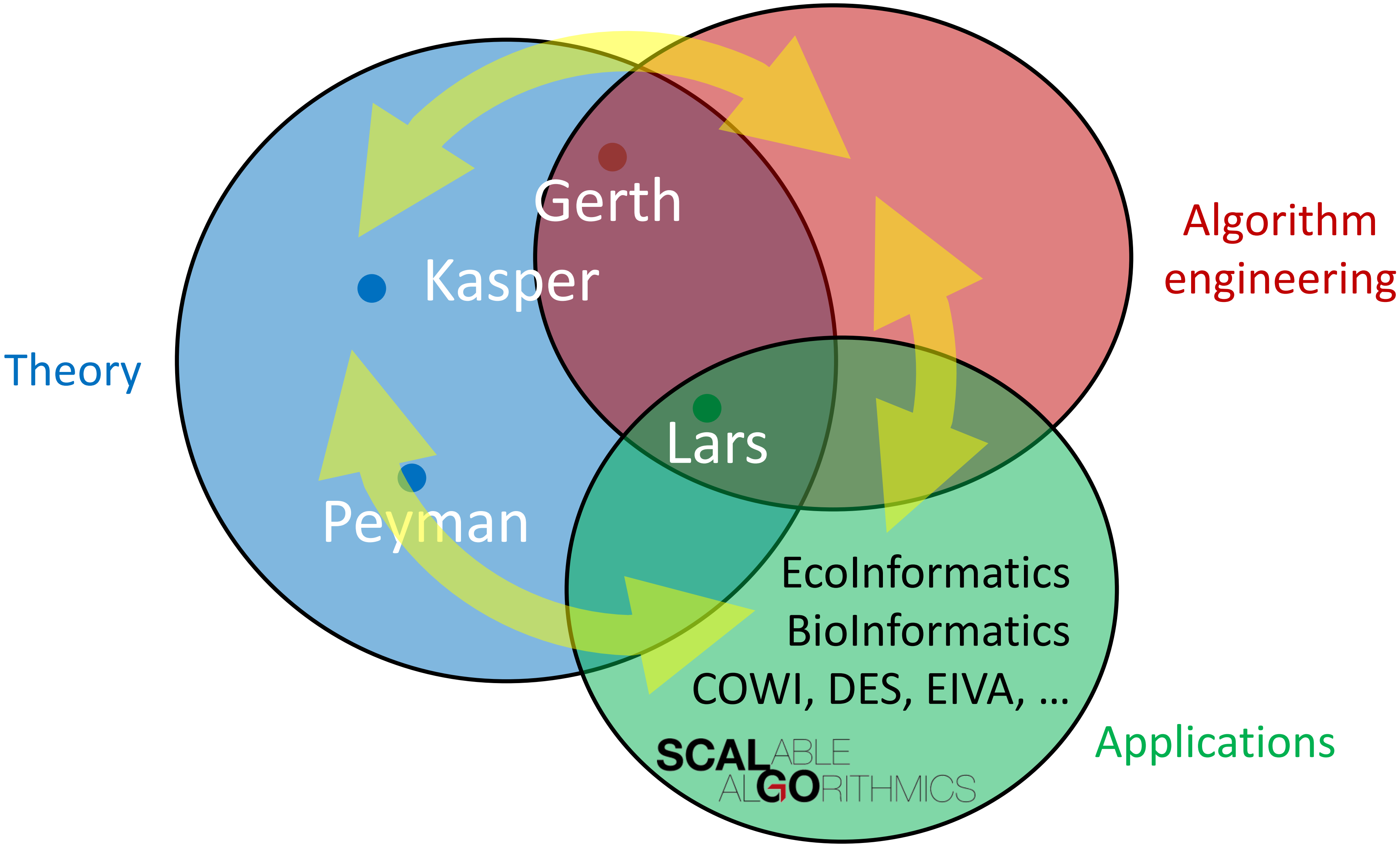
↓ ?

$O(1)$

lower bound ?

integer sorting, cost per element

Algorithms Research

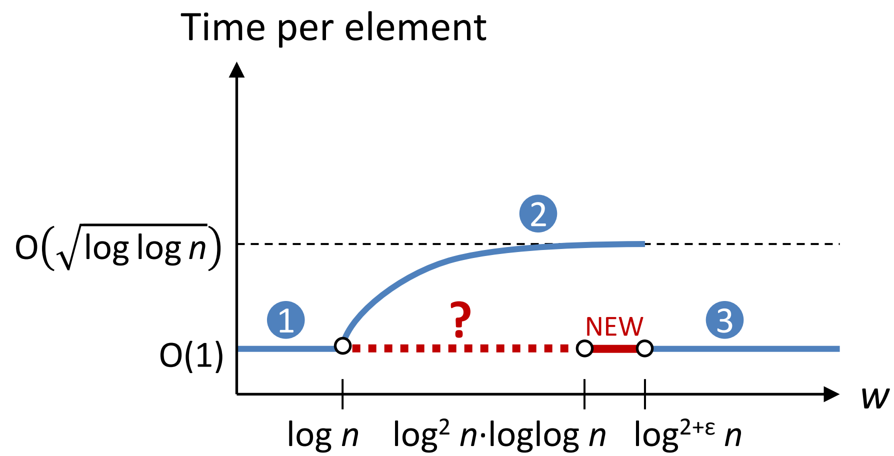


Research – Models of Computation

- **RAM model** memory access and other operations O(1), sometimes * is ω(1)
- **Pointer model** disallow arrays, memory is a graph with O(1) out-degree
- **Functional model** pointer model with no side-effects, implies persistence
- **Comparison model/decision trees** simple lower bounds
- **Cell-probe model** strong lower bounds, applies to RAM model
- **Bit-probe model** fundamental lower bounds, special case of cell-probe
- **Implicit model** O(1) working space, store information as input permutation
- **IO model** focus on number of memory-disk transfers
- **Cache-oblivious model** abstract model to model multiple memory layers
- **Streaming model** limited working space, single or multiple scans of input

Integer Sorting Results

(n words of w bits)



1	Bucket sort	$O(n+2^w)$	
	Radix sort; Hollerith 1887	$O\left(n\frac{w}{\log n}\right)$	
	van Emde Boas 1975 Willard 1983	$O(n \log w)$	superlinear space expected
	Kirkpatrick and Reicsh 1983	$O\left(n \log \frac{w}{\log n}\right)$	
	Merge sort: von Neumann 1945	$O(n \log n)$	comparison based optimal
2	Thorup and Han 2002	$O\left(n\sqrt{\log(w/\log n)}\right)$ $O\left(n\sqrt{\log \log n}\right)$	expected
3	Andersson et al. 1998	$O(n)$	expected, $w \geq \Omega\left(\log^{2+\varepsilon} n\right)$
	Belazzougui, Brodal, Nielsen 2014	$O(n)$	expected, $w \geq \Omega\left(\log^2 n \cdot \log \log n\right)$

Allan Grønlund, Seth Pettie FOCS 2014

Threesomes, Degenerates, and Love Triangles

3SUM problem : Given real numbers $x_1, \ldots, x_n$, does there exist $x_i + x_j + x_k = 0$?

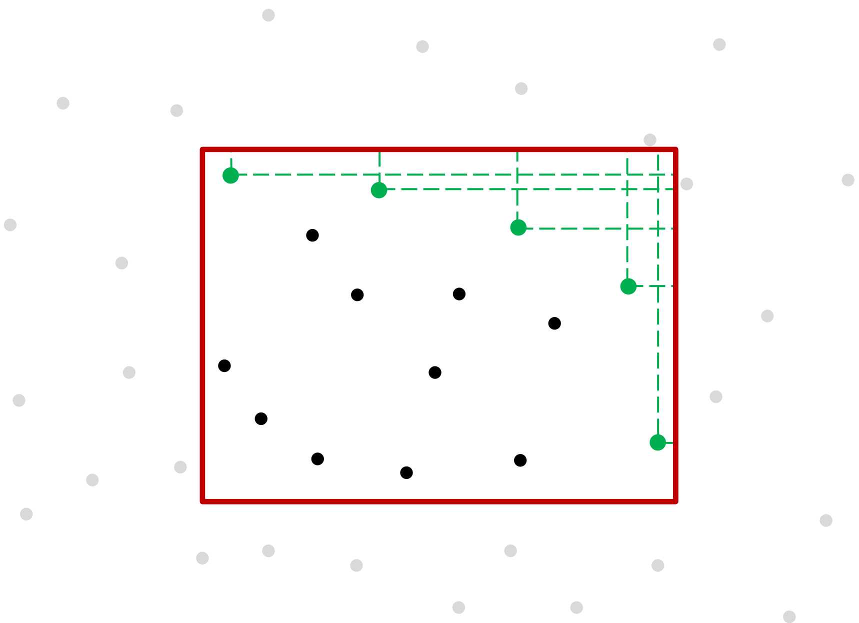
3 2 -9 (11) (-7) 5 9 10 (-4) 6

~~**Conjecture** : 3SUM requires time $O(n^2)$~~

Theorem : 3SUM can be solved in time $O(n^2 / (\log n / \log\log n)^{2/3})$

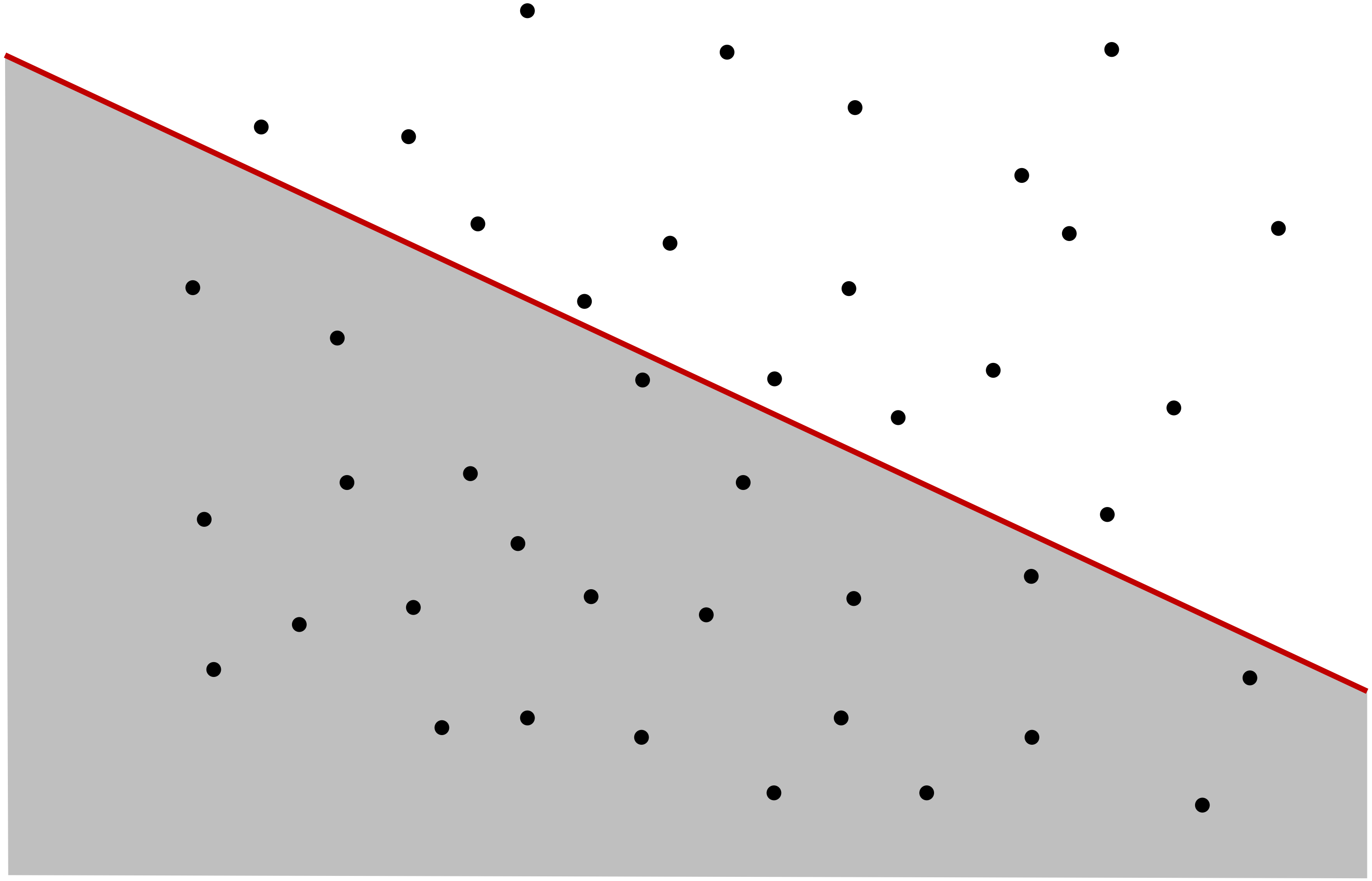
Theorem' : 3SUM has decision tree complexity $O(n^{3/2}\sqrt{\log n})$

Planar Orthogonal Skyline Queries



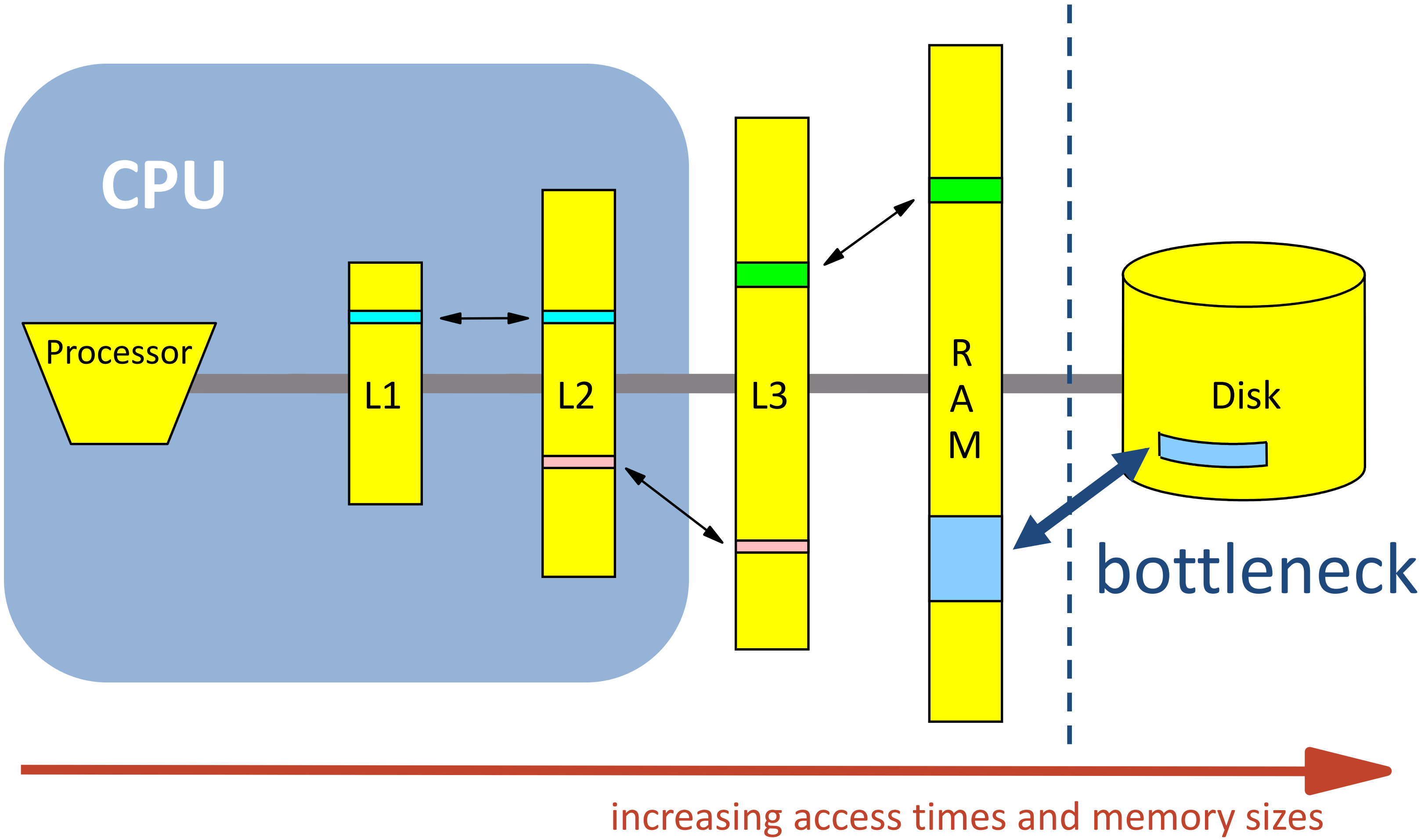
	Space (words)		Query
Reporting	$O(n)$ $O(n \cdot \lg\lg n)$ $O(n \cdot \lg^{\varepsilon} n)$		$O(k \cdot \lg^{\varepsilon} n)$ $O(k \cdot \lg\lg n + \lg n / \lg\lg n)$ $O(k + \lg n / \lg\lg n)$
Counting	$O(n)$ $O(n \cdot \lg^{O(1)} n)$	 $\Rightarrow$	$O(\lg n / \lg\lg n)$ $\Omega(\lg n / \lg\lg n)$

Half-Space Range Reporting



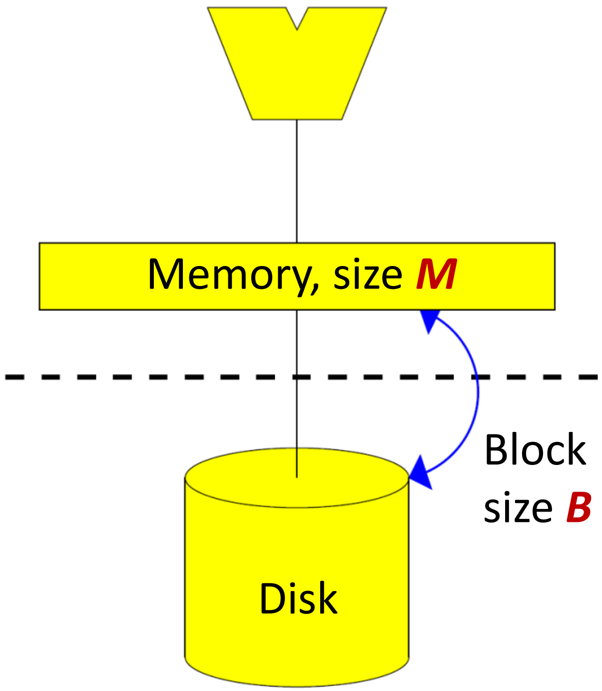
Query time Q(n) $\Rightarrow$ **Space** $\left(\frac{n}{Q(n)}\right)^{\Omega(\sqrt{d})}$

Memory Hierarchies



IO Model

- Cost = # block transfers



Cache-Oblivious Model

- I/O model...but algorithms do not know B and M
- Assume optimal cache replacement strategy
- Optimal on all levels (under some assumptions)

Scanning $O(N/B)$ IOs, Sorting $O(N/B \cdot \log_{M/B} N/B)$ IOs

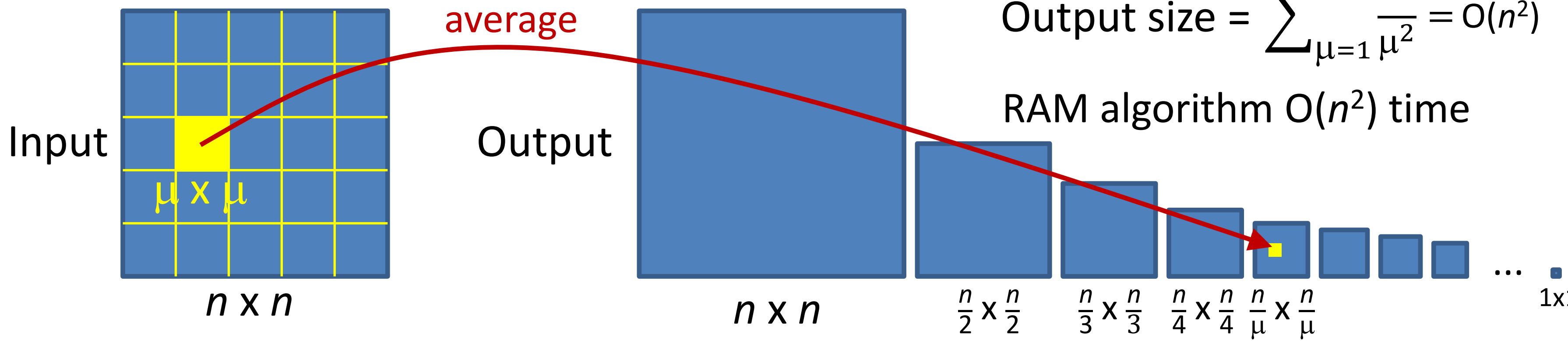
Alok Aggarwal and Jeff Vitter. The Input/Output Complexity of Sorting and Related Problems. Communications of the ACM 31(9) 1988.

Matteo Frigo, Charles E. Leiserson, Harald Prokop, Sridhar Ramachandran. Cache-Oblivious Algorithms. ACM Transactions on Algorithms, 8(1), Article No. 4, 2012.

Computing Multiresolution Rasters

Output size = $\sum_{\mu=1}^{n} \frac{n^2}{\mu^2} = O(n^2)$

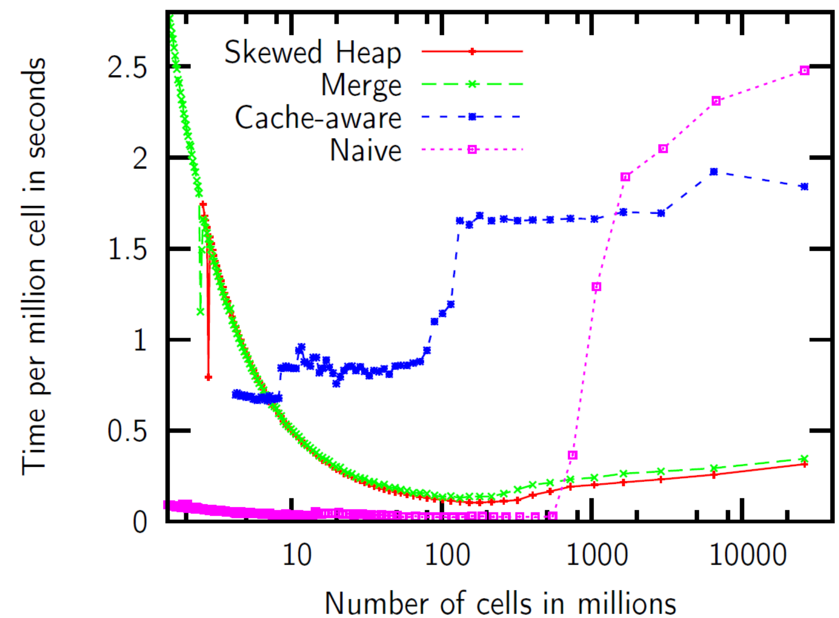
RAM algorithm $O(n^2)$ time



IO Model - O(Sort(n^2)) IOs
Lars Arge, Herman Haverkort and Constantinos Tsirogiannis. ACM SIGSPATIAL 2012.

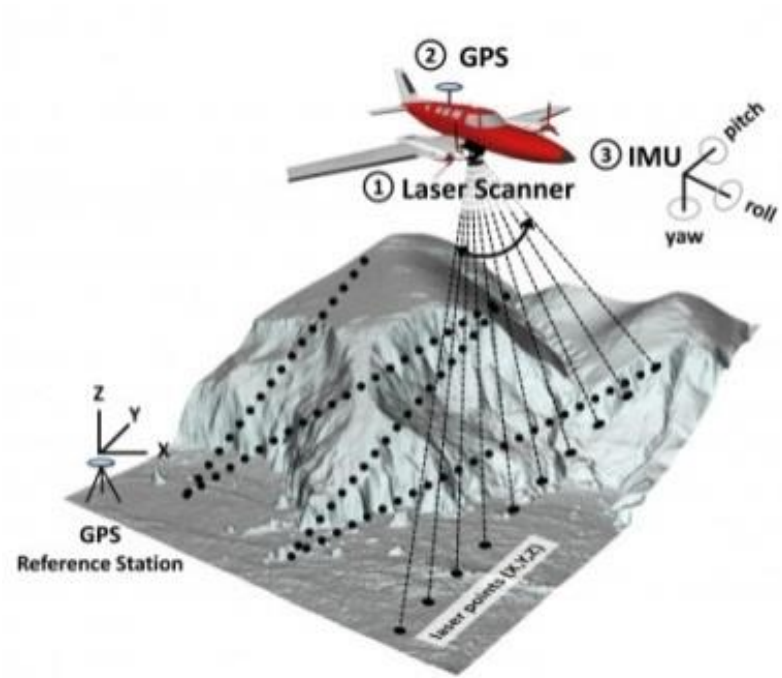
Cache Oblivious Model - O(Scan(n^2)) IOs
Lars Arge, Gerth Stølting Brodal, Jakob Truelsen, and Constantinos Tsirogiannis. ESA 2013.

$$\sum_{\substack{\rho \text{ is prime} \\ \rho \leq x}} \frac{1}{\rho} = O(\log \log x)$$



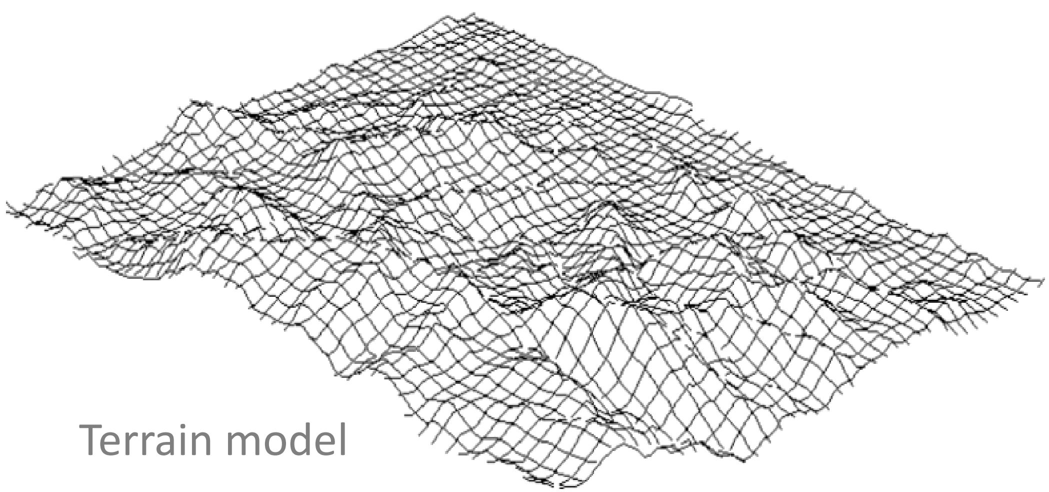

Lars
Water

Terrain Research



3	2	4	3	2	4
7	5	8	7	5	8
7	1	9	7	1	9
3	2	4	3	2	4
7	5	8	7	5	8
7	1	9	7	1	9

Height matrix



Terrain model

TerraSTREAM: Terrain Processing Pipeline

What
TerraSTREAM is a collection of software modules for computation on very large digital terrain models.

Problem
Modern sampling techniques yield datasets in the order of hundreds of gigabytes, which cannot be processed by standard software.

Solution
Use provable efficient algorithms specifically tailored to terrains much larger than the size of the memory. These algorithms tries to minimize the number of disk accesses.

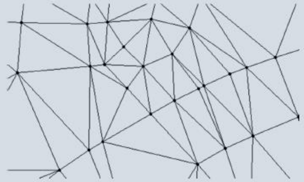
Properties
- Where it makes sense, all modules work on both grid and triangulated terrains.
- Works on GNU/Linux, Mac OS X and Windows.
- Many of the modules export several different parameters that can be tweaked by the user for maximum flexibility.
- Supports reading and writing grid mosaics and LAS files.

Usage
The modules presented on this poster are designed independently of any frontend. Frontends designed for:

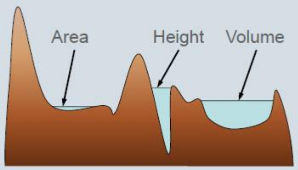
Commercialization
TerraSTREAM is currently used by a range of both governmental and commercial organizations. To secure ongoing service and development the software has been commercialized in the company SCALGO.

SCALABLE ALGORITHMICS

Model Construction
Constructs a terrain model (a grid or a triangulation) from the point cloud representing the terrain. Also, computes quality of a grid by computing the distance to nearest point in input point cloud.

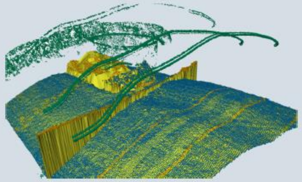


Topological Simplification
Removes insignificant depressions from a terrain model. Significance is user defined in terms of the height, area and volume of the depression. Insignificant depressions are removed by raising them to the height of their surroundings.

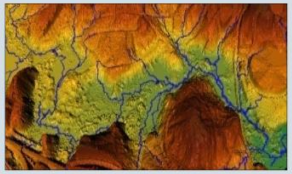


Sea Level Rise Mapping
Computes what part of a terrain model that will be flooded for any given sea level rise. Alternatively the module can compute for each part of the terrain at what sea level this particular part is flooded.

Point Cleaning
Detects outlier points that are due to noise in measurements. Every closely connected component of points are associated with a score that indicates how far this component lies from the terrain surface. A component far away from the surrounding terrain is more likely to be noise.



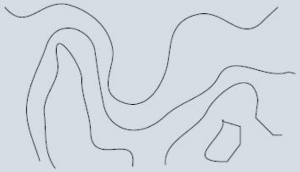
Flow Modelling
Determines the flow direction at each point of the terrain model and then computes upstream area of each point using the flow directions. The module supports several different flow models. Topological simplification prepares a terrain for flow modelling.



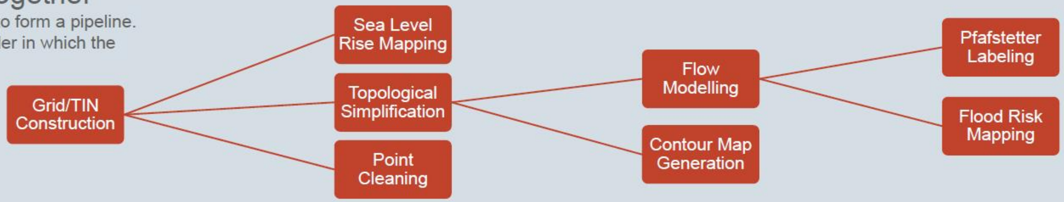
Flash Flood Mapping
In flow modelling, water disappears once it reaches a depression in the terrain. Flood risk mapping models how water fills depressions in the terrain. These depressions eventually spill into neighboring depressions thereby increasing the rate at which this neighbor fills.

Pfafstetter Labeling
Decomposes a terrain model into a hierarchy of watersheds. The pfafstetter labels define a certain hierarchy, which is easy for humans to visualize.

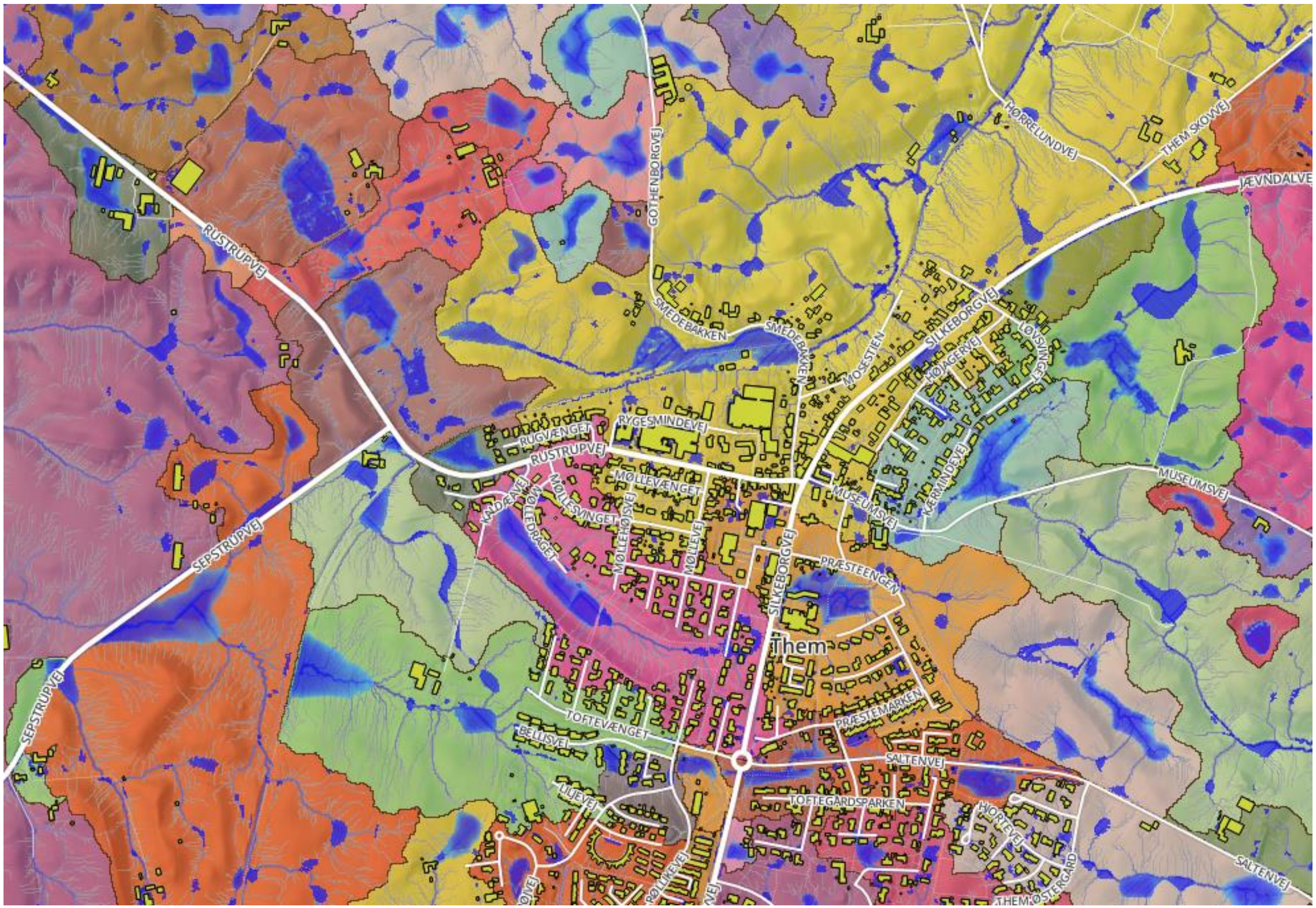
Contour Maps
Constructs and simplifies contour maps of the entire terrain. The contours generated from high resolution terrain models tend to be very jagged and visually unpleasing. Contour maps are simplified while maintaining all significant features and precision.



Fitting Everything Together
The modules can be combined to form a pipeline. This figure shows the typical order in which the modules are invoked



Areas Flooded by 100 mm Rain



SoCG 2010 → MADALGO prototype → SCALGO prodct

Algorithms Research