Online Labeling Problem

Martin Babka ╦, Jan Bulánek ╦, Vladimír Čunát ╦, Michal Koucký ╬, Michael Saks ╩

http://ktiml.mff.cuni.cz/~babka
http://ktiml.mff.cuni.cz/~cunat
http://www.math.cas.cz/~koucky
http://www.math.rutgers.edu/~saks

Charles University Prague
faculty of mathematics and physics

The Academy of Sciences of the Czech Republic

Rutgers

Labeling Problem Description

Input:

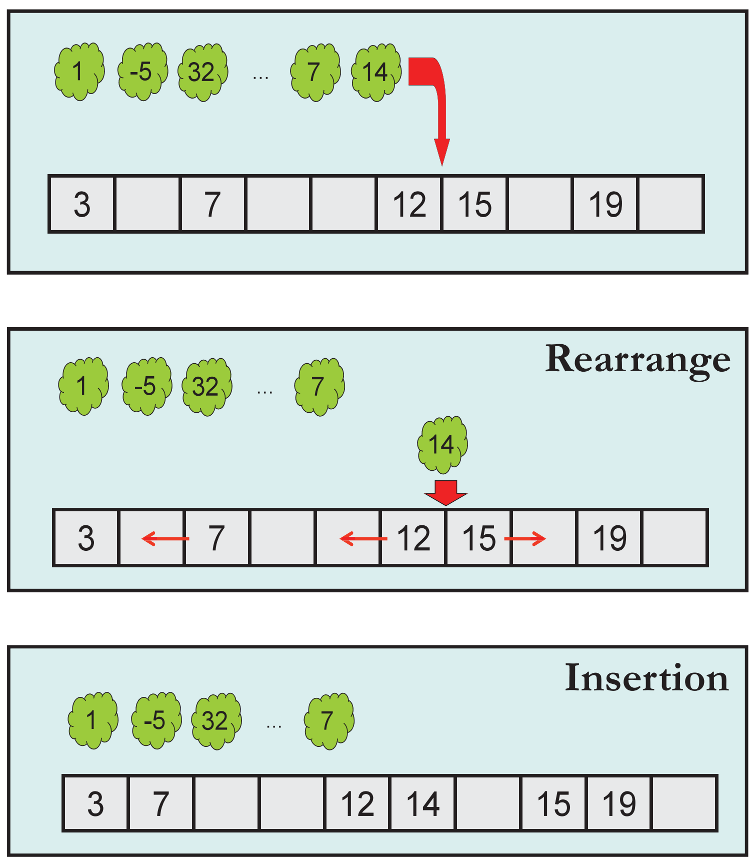
- A stream of n numbers, revealed on-line.
- An array of size $m>n$, initially empty.

Objective:

- Insert the numbers one by one into the array, while maintaining them in correct order after each insertion.
- Rearranging items may be needed.
- We want to minimize the total number of item moves.

Example:

Inserting 14 into the sorted array.



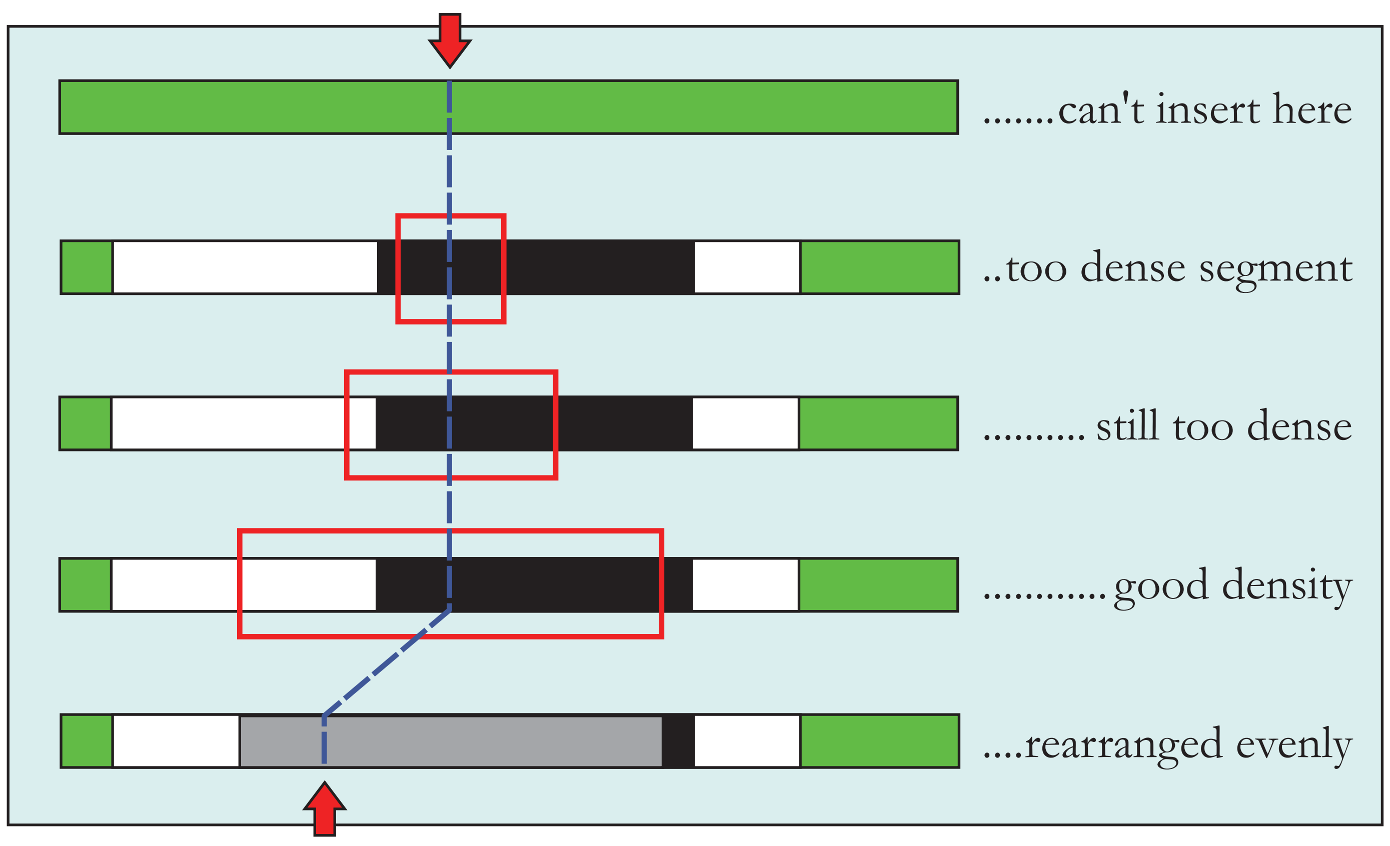
Applications

- Dynamic linear order maintenance. [Bender et al. '02]
  - ↳ Orthogonal graph drawing. [Biedl, Kant '98]
  - ↳ Persistent data structure timestamps. [Driscoll, Sarnak, Sleator, Tarjan '89]
- Cache oblivious dictionaries based on B-trees. [Bender, Demain, Farach-Colton '05]
- Approximate indexed lists, item label used as the index. [Andersson, Petersson '98]
- Distributed controllers. [Emek, Korman '11]

Upper Bound Ideas

- If possible, insert into the middle of free space at the correct position.
- Otherwise find the smallest *acceptable* segment and evenly rearrange all items within. A segment is acceptable iff it contains the insertion point and has a correct density.
- Deletions can be handled by periodically rebuilding the whole array.

Example:



Insert Time Lower Bounds

- Optimal algorithms are mostly known since 80's.
- Our recent work gives **tight** asymptotic lower bounds, building on ideas from Dietz, Seiferas, and Zhang.
- All bounds are amortized, some known upper bounds are even worst-case.

Insertion Time	Array Size
$\Theta\left(\log^2 n \cdot \log \frac{n}{f(n)}\right)$	$m = n + f(n),\ f(n) \geq 1$
Upper bound: [Zhang '93]	
Lower bound: [Bulánek, Koucký, Saks STOC '12]	
$\Theta\left(\log^2 n\right)$	$m = cn,\ c > 1$
Upper bound: [Itai, Konheim, Rodeh '81]; [Itai, Katriel '07] simplified; [Willard '92] worst-case	
Lower bound: [Bulánek, Koucký, Saks STOC '12]	
$\Theta\left(\frac{\log^2 n}{\log f(n)}\right)$	$m = n \cdot f(n),\ f(n) = o(n)$
Upper bound: [Itai, Konheim, Rodeh '81]	
Lower bound: [Babka, Bulánek, Čunát, Koucký, Saks manuscript '12]	
$\Theta\left(\frac{\log n}{\log f(n)}\right)$	$m = n^{f(n)},\ f(n) > 1$
Upper bound: [Bulánek, Koucký, Saks STOC '12]	
Lower bound: [Babka, Bulánek, Čunát, Koucký, Saks ESA'12]	

References

A. Andersson and O. Petersson: *Approximate indexed lists.* Journal of Algorithms, 1998.

M. Babka, J. Bulánek, V. Čunát, M. Koucký, and M. Saks.: *On online labeling with superlinearly many labels.* Manuscript, 2012.

M. Babka, J. Bulánek, V. Čunát, M. Koucký, and M. Saks.: *On online labeling with polynomially many labels.* ESA, 2012.

M. A. Bender, E. D. Demaine, and M. Farach-Colton.: *Cache-oblivious B-trees.* Journal on Computing, 2005.

M. A. Bender, R. Cole, E. D. Demaine, M. Farach-Colton, and J. Zito.: *Two simplified algorithms for maintaining order in a list.* ESA, 2002.

T. Biedl and G. Kant: *A better heuristic for orthogonal graph drawings.* Computational Geometry, 1998.

J. Bulánek, M. Koucký, and M. Saks: *Tight lower bounds for the online labeling problem.* In STOC 2012.

J. R. Driscoll, N. Sarnak, D. D. Sleator, R. E. Tarjan: *Making data structures persistent.* Journal of Computer and System Sciences, 1989.

Y. Emek and A. Korman: *New bounds for the controller problem.* Distributed Computing, 2011.

A. Itai, I. Katriel: *Canonical density control.* Information Processing Letters, 2007.

A. Itai, A. G. Konheim, and M. Rodeh: *A sparse table implementation of priority queues.* In ICALP 1981.

J. Zhang: *Density control and on-line labeling problems.* PhD thesis, University of Rochester, 1993.

P. F. Dietz, J. I. Seiferas, and J. Zhang: *A tight lower bound for online monotonic list labeling.* SIAM J. Discrete Math, 2004.

D. E. Willard: *A density control algorithm for doing insertions and deletions in a sequentially ordered file in a good worst-case time.* Information and Computation, 1992.

╦ Supported by SVV project nr. 265 314 and by projects of Grant Agency of Charles University nr. 265 111 and 344 711.

╬ Partially supported by the Sino-Danish Center CTIC grant 61061130540, by GAČR P202/10/0854, GA AV ČR grant IAA100190902, Center of Excellence CE-ITI (P202/12/G061 of GA ČR), and RVO 67985840.

╩ Also supported in part by NSF under grant CCF-0832787.