Tag-Cloud Drawing

Algorithms for Cloud Visualization

Owen Kaser[1] Daniel Lemire[2]

[1]University of New Brunswick, Saint John campus

[2]Université du Québec à Montréal

WWW2007 Workshop on Tagging and Metadata for Social Information Organization

Introduction

- After the tagging . . . the tag cloud.

- Visualize aggregate information, promote interaction.
- What is a good cloud layout?
- . . . and how can we compute it?

(optional slide) Do. . .

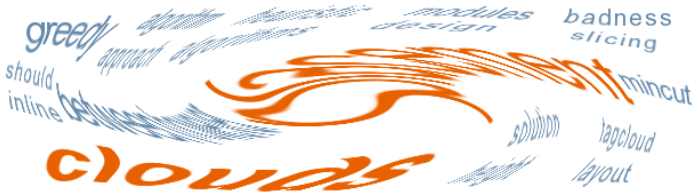
Do tell me lots. Use

- font $\mathcal{FAMILY}$
- point size
- colour
- location and position. Cluster related tags.

HCI people can help here.

(optional slide) Don't...

pda and busy page with cloudugly



page impatient person, clipart?

- Don't waste space. This favours rectangles.
- Don't be ugly. Banish whitespace blobs.
- Don't be weird. Use simple HTML+CSS.
- Don't require horizontal scrolling. Limit width.
- Don't be tardy. Consider compute power and bandwidth.

(alternative slide) Setup

- Tag clouds are fixed-width rectangles.
- Tags are displayed in a range of font sizes. For layout, only tags' bounding boxes matter.
- A small gap (at least) is required between tags.
- (alternative 1) Related tags should be in close proximity.
- (alternative 2) Tags should be organized into lines.
- Colour, font family etc. are ignored — except for their effects on bounding boxes.

Related work

Hassan-Montero and Herreno-Solana : clustering, 8×12 display (?)
Millen et. al: auxiliary index, find tag in large cloud
Bielenberg(sp)

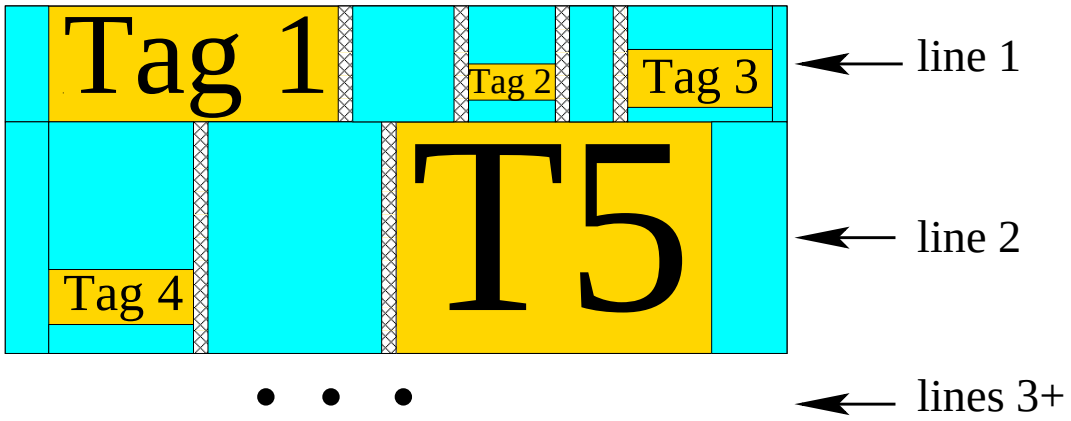
Inline text

Inline text uses only "inline" HTML: eg `span`, `font`, `b` or `br`.
Not tables.
A tag cloud with inline text consists of a sequence of rows of text.
Most tag clouds on the Web are currently like this. . .
. . . unlike most visualizations of topic/concept maps.

We have the power to insert $\langle br \rangle$

tag1 tag2 tag3
tag4

Badness model



whitespace (here blue!) is bad. It's left/right of tags + above/below short tags. Required inter-tag space isn't bad.

Line k's badness b_k depends only on the tags on line k (and cloud width). Independent of spacing.
Overall badness: the L_p norm of line badnesses: $\sqrt[p]{\sum_k |b|^p}$
except L_∞ is "max".

Cannot reorder tags

First problem variant: tags given some desired order: often alphabetic.
Only ability we have is to insert `br` elements.
Cloud (with/without `br`'s) will be rendered normally by the browser's greedy line-breaking algorithm.
Goal: insert them so as to minimize total badness.
Almost like the text-justification problem for TEX. So we modify the Knuth-Plass (K-P) algorithm [Knuth and Plass, 1982].

Knuth-Plass

K-P uses dynamic programming.
It tabulates c_i, the minimum total (L_2) badness of justifying the first i words.
K-P considers hyphenation + handles last line in paragraph specially
But we consider varying font sizes instead; allow L_1, L_2 or L_∞.

Implementation and Experiments

For n tags, our Knuth-Plass variant runs in $O(n^2)$ time with small constants: 140 tags takes less than 1 ms.
Range of font sizes: ???
Test data:

1. 65 ZoomClouds (obtained from a REST API they publish).
2. 80 tag sets obtained from Project Gutenberg e-Books. (Every e-Book is tagged by each long word occurring in it. Then take the most frequent k tags for the e-Book.)

Badness vs Area

- Within the cloud, we see only tags, (bad) whitespace, mandatory (small) inter-tag gaps.
- The L_1 norm combines line badnesses by summing them.
- Ignoring gaps, we see minimizing badness also minimizes area.
- The L_2 norms could lead to more area than greedy line breaking. Beauty vs. area trade-off (?) Not really: $< 1\%$ area increase over L_1.
- L_∞ disastrous for area: 200% increase!

Measured Badnesses

2 custom barcharts, for L1 and L2. Show opt, greedy x Gut, Zoom x Alpha, weight.
Note how well sorted-by-weight does, even with greedy line breaking.

Reorderable tags

- We gain additional flexibility to reduce badness/area if we can rearrange tags. How much? . . . experiments. . .
- Sorting by weight looks promising
- If inter-tag gap = 0, we have the (level version) Strip-Packing Problem (SPP) [Lodi et al., 2002].
- SPP is NP-hard but good approximation algorithms are known.

Strip-Packing Problem (SPP) - extra slide

pictures showing a bunch of boxes with tags
picture showing level-oriented and non-level oriented packings

Approximation Algorithms for SPP / Cloud Layout

Although we have non-zero inter-tag gap, we look to SPP for inspiration. See [Coffman, Jr. et al., 1980]. We find

- First-Fit Decreasing Height (FFDH)
- Next-Fit Decreasing Height (NFDH)
- (our variant) Next-Fit Decreasing Height, Weight (NFDH).
- We also tried 10 random orderings, followed by our Knuth-Plass variant

Add pictures!

Results

Pictures

For L_1 or L_2, NFDHW is good.

For L_2, simply sorting by weight and using our K-P variant is also good.

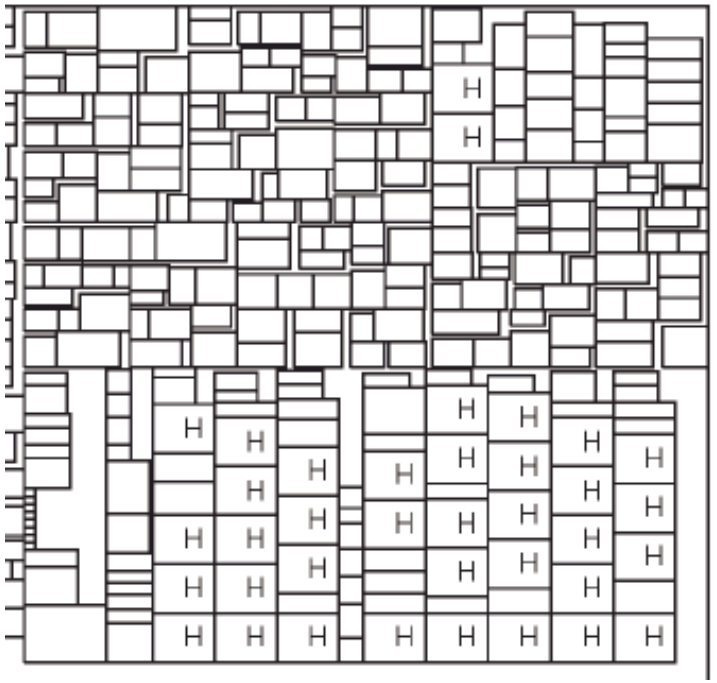
Clouds with arbitrary placement

Let's

- relax level-oriented packing (full strip packing)
- consider clustering related tags eg tags t_1, t_2 commonly on the same e-Book lines $\Rightarrow$ t_1 placed near t_2 in the cloud

Something seems very familiar about this problem...

Tag clouds and VLSI placement



(from [Cong et al., 2006])

Issue comparison

Similarities:

- place oriented rectangles in the plane
- cluster related rectangles
- (sometimes) gaps needed between rectangles
- (for "fast floorplanners") use interactively (speed required)
- (optional) slightly resize rectangles

Differences:

- electrical connectivity is transitive
- tags cannot be rotated
- cloud "legibility gaps" differ from VLSI wiring space
- clouds with 1M tags — unlikely

Our thesis

Thesis, part I: "Fast floorplanner" techniques **can** be adapted for tag-cloud layout.
In particular, *min-cut layout* [Bruer, 1977] and *floorplan sizing* [Stockmeyer, 1983] are mature.

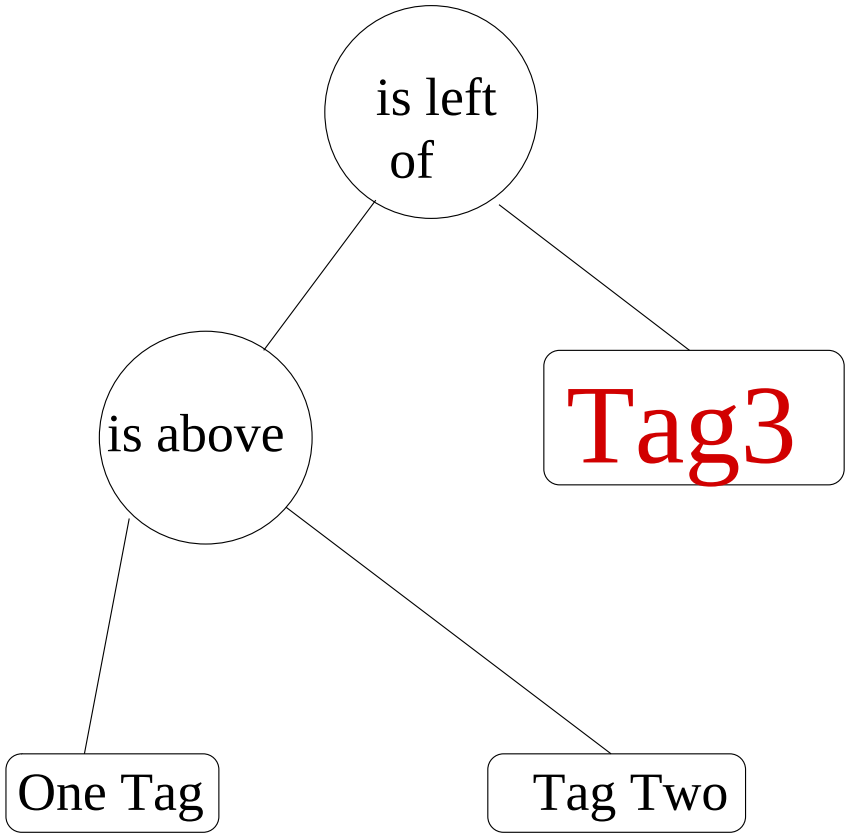
Thesis, part II: Existing full-scale VLSI floorplanners/placers **cannot** be easily adapted.
Part II is largely untested.

Mincut placement of tags

Key idea is recursive bipartitioning, a divide-and-conquer approach.

1. Divide tags into 2 balanced groups, keeping strongly related tags together.
   (Mincut refers to (broken) relationships from two tags being in different groups)
2. Recursively lay out each group.
   Terminal propagation [Dunlop and Kernighan, 1985] considers tags' relationships outside the current subproblem.
3. Compose the two layouts (horizontally or vertically). Tricky to enforce "nearly width w" constraint.

Slicing tree

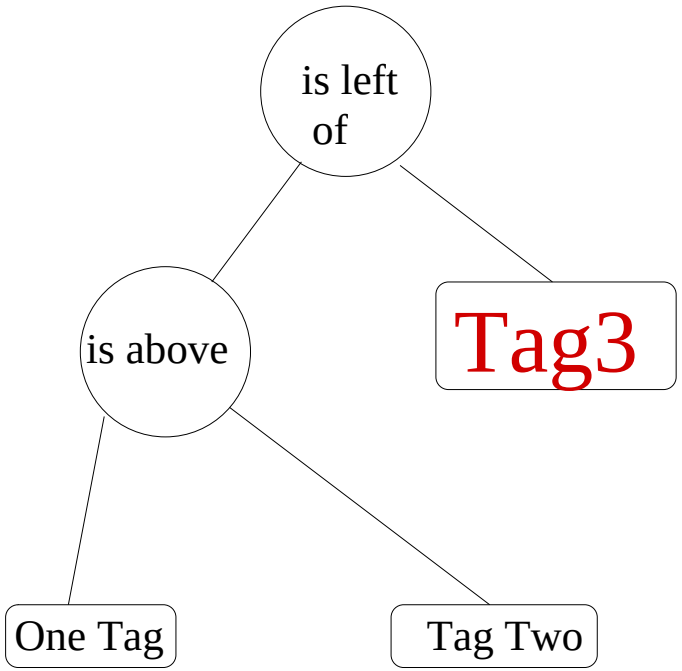


Mincut approach generates a slicing tree.

- Leaves: individual tags
- Internal nodes: record whether two layouts are composed vertically or horizontally

From slicing tree to HTML

To transform a slicing tree to HTML, use nested tables.
1×2 or 2×1, depending on vertical or horizontal composition.




```
<table><tr>
  <td><table>
    <tr><td>One Tag</td></tr>
    <tr><td>Tag Two</td></tr>
    </table></td>
  <td>Tag3</td>
</tr></table>
```

"Floorplan" Resizing

- Gain better packings from adjusting a tag's aspect ratio (while holding tag area near constant).
- One extreme adjustment is 90° rotation.
- That's Bad for tags!
- Given s total shape alternatives, in $O(s \log s)$, we can find the optimal size for each tag [Shi, 1995].
- Note: Remarkably hard to play with `font-size`, `font-family`, `letter-spacing`, `font-weight` to get alternatives.
- If only browsers really supported `font-stretch`!

Results measured

Results consider

- layout area and
- weighted distance $= \sum_{\text{tags } i,\ j} \text{dist}(i,j) * \text{weight}(i,j)$, where dist($i$,$j$) is Euclidean distance between the bounding-box origins of tags i and j. Weight is the strength of relationship.

We compared mincut placement against greedy-sorted-by-height, greedy-random-order and a block packer, COMPASS.

Results

No. Tags	Min-cut	gr-sorted	gr-random	COMPASS
Area (pixels^2)				
20	31	37	46	29
50	63	62	85	59
100	111	99	139	98
200	192	165	231	170
Weighted distance (kilopixels)				
20	61	124	120	65
50	166	282	271	180
100	296	465	482	382
200	438	693	765	654

Min-cut placed even the largest clouds within 100 ms.

Client vs. server

Who has what information? Client (browser) or server (tag database)

- Client: display width
- Client: bounding box sizes.
- Server: tags, requested font sizes
- Server: semantic tag relationships

- Tag relationships may be large; bandwidth problems?.
- Layout may require significant processor horsepower.

Conclusions and Future Work

- Even simple heuristics can reduce badness and area substantially. (A factor of 3 in some cases.)
- Simply sorting tags by weight and using K-P (or greedy) is effective for area minimization.
- Min-cut placement is fast, clusters tags well, and is not bad for area.
- Future: take other "beauty metrics" (eg symmetry) from graph drawing and HCI tag-cloud usability studies. Do layout for them.
- Adopt compute-intensive methods from VLSI design: create beautiful clouds of "static content" (cloud-of-the-month).

ZoomClouds, Jan. 24, 2007

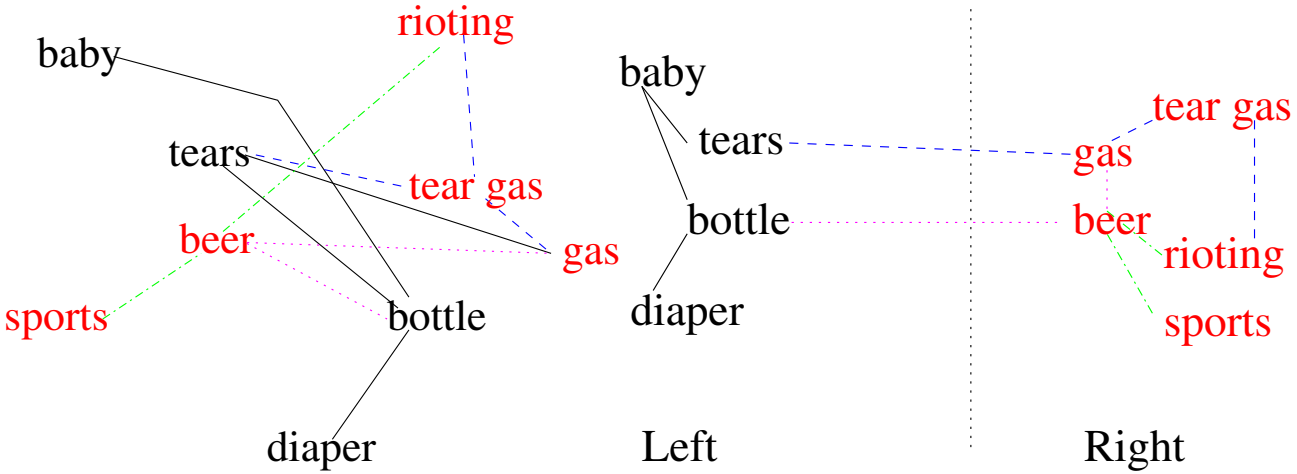
Animation of mincut placement

to do, can be optional

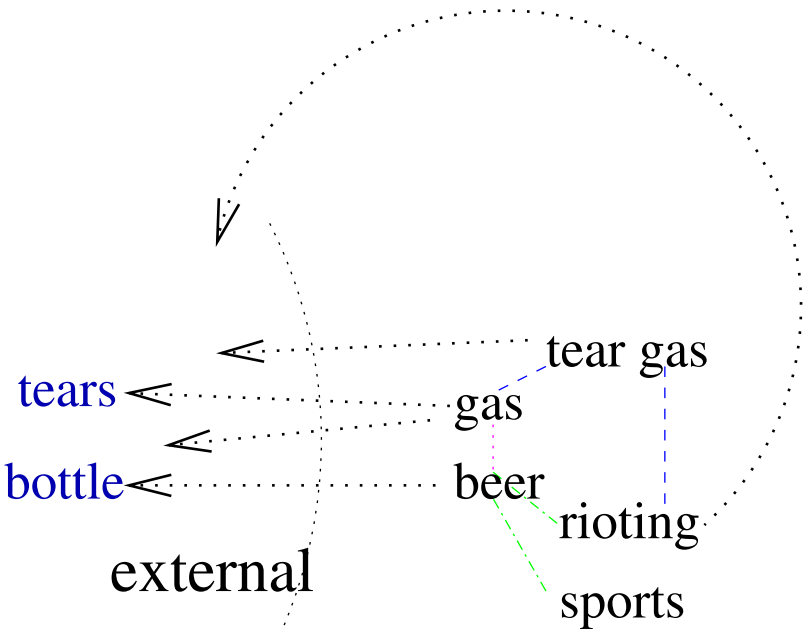
Sizing of slicing floorplan

to do, can be optional

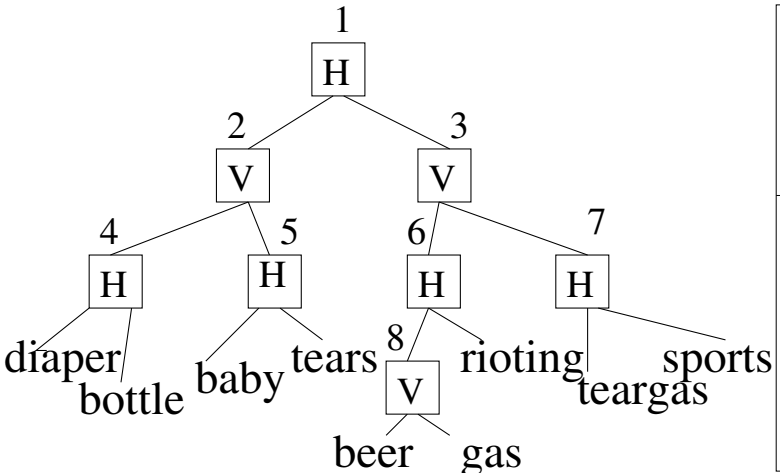
pict

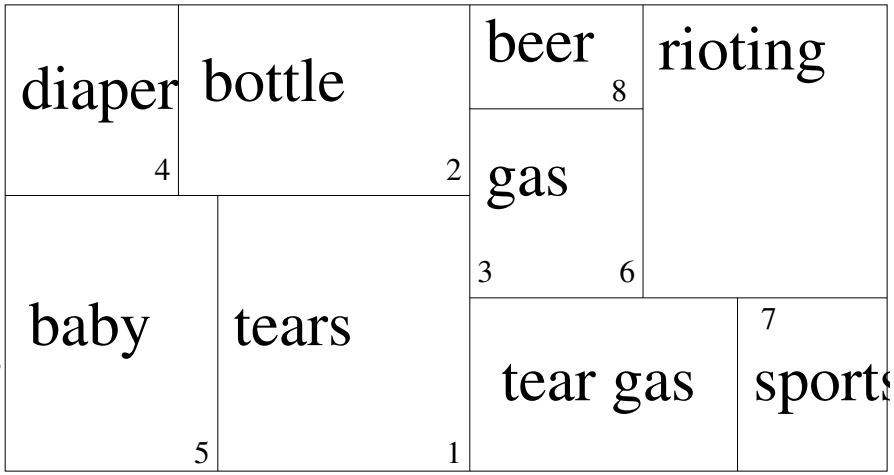

baby
rioting
tears
tear gas
beer
gas
sports
bottle
diaper
baby
tears
bottle
diaper
Left
tear gas
gas
beer
rioting
sports
Right

pict

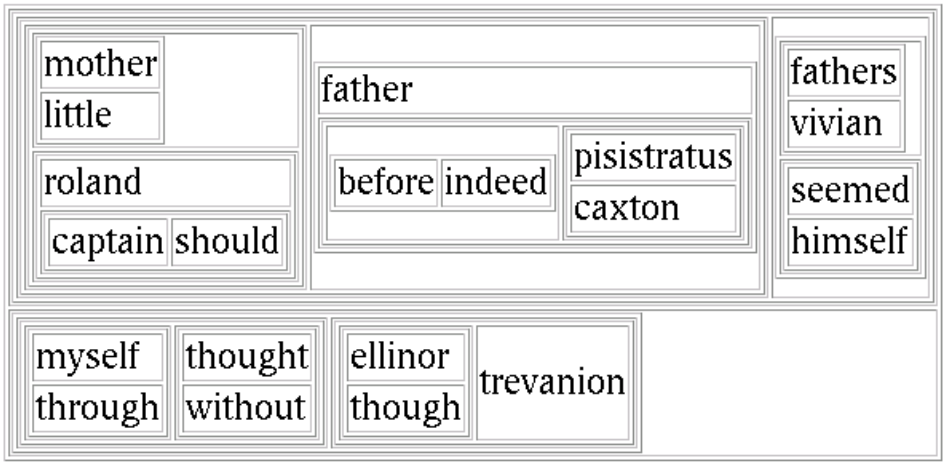


pict


1
H
2
V
3
V
4
H
5
H
6
H
7
H
diaper
bottle
baby
tears
8
V
rioting
teargas
sports
beer
gas


diaper
4
bottle
2
beer
8
rioting
gas
3
6
baby
5
tears
1
tear gas
7
sports

pict


mother
little
father
fathers
vivian
roland
captain
should
before
indeed
pisistratus
caxton
seemed
himself
myself
through
thought
without
ellinor
though
trevanion


mother
little
roland
captain
should
before
indeed
myself
through
thought
without
ellinor

pict


against almost already although always another answered anything appeared around arrived august autumn beautiful became because become before behind beneath besides better between beyond breeze breton brittany brother brought called captain carried chapel chapter cheeks children chinese closed clouds coming continued cottage country course covered depths different direction distance distant during enough entered etexts evening evenings everything expression fallen falling family father feeling filled fisherman fishermen fishers fishing flowers followed france french gaoses giving gloomy grandmother grandson granite granny gutenberg having herself himself horizon iceland icelanders immense impression indeed itself length leopoldine letter letters listened little lonely longer looked looking mademoiselle making marriage midnight moment monsieur morning mother movement mysterious narrow nature nothing notwithstanding oclock others outside paimpol particular passed passing peculiar people perhaps pierre placed ploubazlanec porseven present pretty project quickly rapidly rather really received remained remembered respect return returned returning rising rushed sailor sailors saying scarcely season second seeing seemed shoulders silence simple sitting slowly smiled something sometimes spread standing stopped straight strange strength strong struck suddenly summer sweetheart sylvestre sylvestres tender themselves things thinking though thought thoughts thousand through thrown together towards turned understand virgin waiting walked waters weather wedding window winter wished without written yellow yvonne

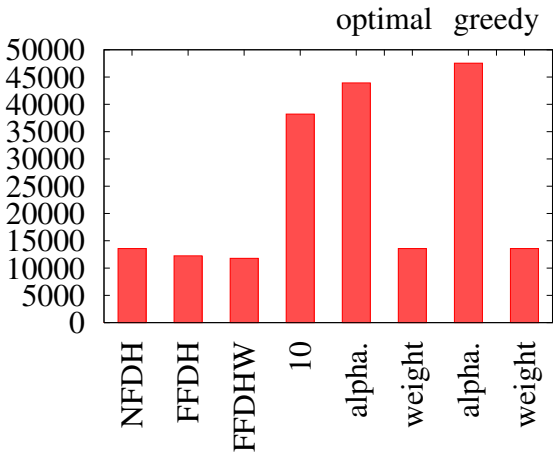
against almost already although always another answered anything appeared around arrived august autumn beautiful became because become before behind beneath besides better between beyond breeze breton brittany brother brought called captain carried chapel chapter cheeks children chinese closed clouds coming continued cottage country course covered depths different direction distance distant during enough entered etexts evening evenings everything expression fallen falling family father feeling filled fisherman fishermen fishers fishing flowers followed france french gaoses giving gloomy grandmother grandson granite granny gutenberg having herself himself horizon iceland icelanders immense impression indeed itself length leopoldine letter letters listened little lonely longer looked looking mademoiselle making marriage midnight moment monsieur morning mother movement mysterious narrow nature nothing notwithstanding oclock others outside paimpol particular passed passing peculiar people perhaps pierre placed ploubazlanec porseven present pretty project quickly rapidly rather really received remained remembered respect return returned returning rising rushed sailor sailors saying scarcely season second seeing seemed shoulders silence simple sitting slowly smiled something sometimes spread standing stopped straight strange strength strong struck suddenly summer sweetheart sylvestre sylvestres tender themselves things thinking though thought thoughts thousand through thrown together towards turned understand virgin waiting walked waters weather wedding window winter wished without written yellow yvonne

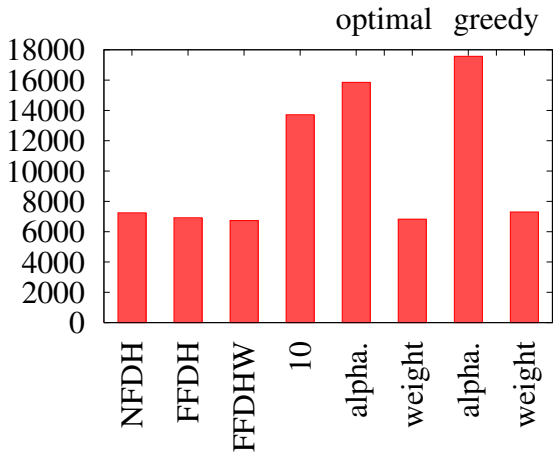
---

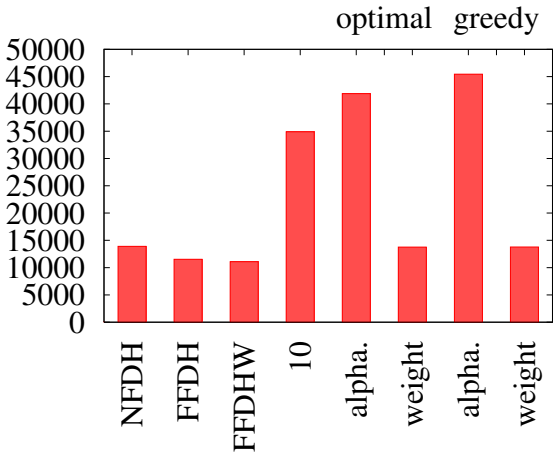

before little seemed through sylvestre another iceland paimpol without almost sailors things always chapter looked nothing others remained thought together against return towards already appeared around become between country evening father granny having herself morning passed project rather returned suddenly though became because behind brought course distance during enough falling feeling fishers fishing himself horizon icelanders looking moment perhaps ploubazlanec present slowly something strength themselves turned weather although answered anything appear arrived august autumn beautiful beneath besides better beyond breeze breton brittany brother called captain carried chapel cheeks children chinese closed clouds coming continued cottage covered depths different direction distant entered etexts evenings everything expression fallen family filled fisherman fishermen flowers followed france french gaoses giving gloomy grandmother grandson granite gutenberg immense impression indeed itself length leopoldine letter letters listened lonely longer mademoiselle making marriage midnight monsieur movement mysterious narrow nature notwithstanding oclock outside particular passing peculiar people pierre placed porseven pretty quickly rapidly really received remembered respect returning rising rushed sailor saying scarcely season second seeing shoulders silence simple sitting smiled sometimes spread standing stopped straight strange strong struck summer sweetheart sylvestres tender thinking thoughts thousand thrown understand virgin waiting walked waters wedding window winter wished written yellow yvonne

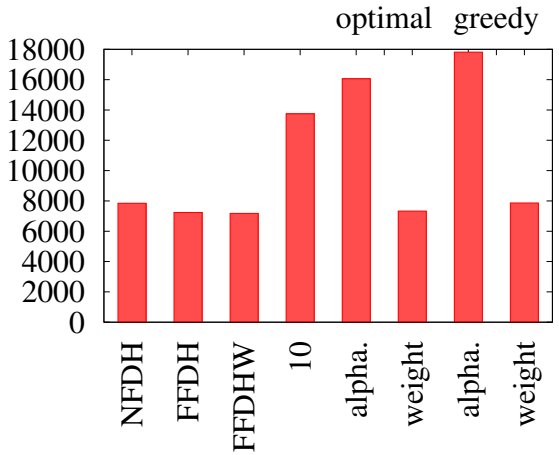

against already around between looked nothing remained towards almost iceland paimpol return sylvestre without although brought course falling fishers himself moment ploubazlanec slowly strength turned always chapter others rather sailors things thought together another before little seemed through answered breeze carried continued outside quickly silence straight strong summer waters window yvonne anything brother chapel coming different etexts lonely midnight oclock passing stopped thousand wished appear arrived beneath impression leopoldine letter making monsieur particular pierre pretty second yellow appeared become breton country evening father granny herself morning passed august captain children covered fishermen itself length listened peculiar returning sitting sweetheart virgin autumn besides closed depths expression filled giving grandson immense rushed sylvestres thrown walked beautiful brittany called direction distant gaoses mother movement respect shoulders thoughts written became behind during enough fishing having horizon project returned suddenly though because distance feeling icelanders looking perhaps present something themselves weather better entered everything gloomy granite gutenberg marriage porseven really sailor sometimes understand beyond cheeks clouds fallen family flowers mademoiselle mysterious narrow scarcely spread tender waiting chinese cottage evenings fisherman grandmother indeed placed received rising simple wedding winter followed france rapidly seeing standing thinking french letters longer nature notwithstanding people remembered saying season smiled strange struck

pict


optimal
greedy
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
NFDH
FFDH
FFDHW
10
alpha.
weight
alpha.
weight


optimal
greedy
18000
16000
14000
12000
10000
8000
6000
4000
2000
0
NFDH
FFDH
FFDHW
10
alpha.
weight
alpha.
weight


optimal
greedy
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
NFDH
FFDH
FFDHW
10
alpha.
weight
alpha.
weight


optimal
greedy
18000
16000
14000
12000
10000
8000
6000
4000
2000
0
NFDH
FFDH
FFDHW
10
alpha.
weight
alpha.
weight


Bruer, M. A. (1977).
Min-cut placement.
Journal of Design Automation and Fault-Tolerant Computing, 1(4):343–362.


Coffman, Jr., E. G., Garey, M. R., Johnson, D. S., and Tarjan, R. E. (1980).
Performance bounds for level-oriented two-dimensional packing algorithms.
SIAM J. Comput., 9(4):808–826.


Cong, J., Romesis, M., and Shinnerl, J. R. (2006).
Fast floorplanning by look-ahead enabled recursive bipartitioning.
IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 25(9):1719–1732.


Dunlop, A. E. and Kernighan, B. W. (1985).
A procedure for placement of standard-cell VLSI circuits.
IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 4:92–98.


Knuth, D. E. and Plass, M. F. (1982).
Breaking paragraphs into lines.
Software — Practice & Experience, 11(11):1119–1184.


Lodi, A., Martello, S., and Monaci, M. (2002).
Two-dimensional packing problems: A survey.
European Journal of Operational Research, 141(2):241–252.


Shi, W. (1995).
An optimal algorithm for area minimization of slicing floorplans.

In *ACM/IEEE International Conference on Computer-Aided Design (ICCAD)*, pages 480–484.

Stockmeyer, L. (1983).
Optimal orientation of cells in slicing floorplan designs.
Information and Control, 57(2–3).