

Optimization-based keyboard design



Learning Goals

- Combinatorial optimization as a UI design approach
- Components of optimization
- Example optimizer and application results

Motivation: Fast typing without errors

- Are some layouts better than others?
- If so, how do we find the best one?





QWERTY, by Christopher Sholes, 1873

Dvorak, by August Dvorak, 1936

The set of the last the last test of the	10,5,100	
		Contrast subsection
	Sectors.	國國國黨
	道道是	

By: https://commons.wikimedia.org/wiki/File:KB_United_States.svg, https://commons.wikimedia.org/wiki/File:KB_United_States_Dvorak.svg



Key Assignment Problem



How many layouts are there? $26! = 4 \times 10^{26}$

For comparison - stars in the universe: https://www.esa.int/Science_Exploration/Space_Science/Herschel/How_many_stars_are_there_in_the_Universe

Intelligent Text Entry

4

Daniel Buschek

Examples of Optimised Designs



Zhai et al. 2000, Dunlop and Levine 2012, Oulasvirta et al. 2013, Gong et al. 2018

Intelligent Text Entry

5

What is "optimal"?

Design space: Best among which options?









Design Space, formalised



Design space **D** with **n** design variables

Intelligent Text Entry

Design Space: Set of possible layouts



design space

*

Intelligent Text Entry

26

Objective Function: How to judge a layout?



Objective Function: How to judge a layout?

- Finger movement time (e.g. Fitts' law) $t(k_1, k_2) = a + b \log_2 (\frac{D}{W} + 1)$
- Language properties (e.g. bigram frequencies) e.g. p("n"|"e") = 0.001



Combined: mean time between two key presses

$$f(d) = \sum_{k_1 \in K} \sum_{k_2 \in K} p(d(k_2)|d(k_1))t(k_1, k_2)$$

where the design *d* maps from keys to characters

Optimizer: How to pick layouts?



Design Task

e.g. keyboard layout optimization



A Simple Optimizer

- Can you think of a trivial optimizer?
- Random Search:
 - 1. Generate random design
 - 2. Keep if better than current best design
 - 3. Repeat

Optimization Landscape

Here: objective function (y) across two design parameters (x, z)



Intelligent Text Entry

Daniel Buschek

Random Guessing



Intelligent Text Entry

Daniel Buschek

Optimizers

Heuristic methods (e.g. Simulated Annealing)

+ Flexible

Not guaranteed to find global optimum

- Exact methods (e.g. Integer Programming)
 - + Guarantees
 - Less flexible objectives

Example: Simulated Annealing

- Metaphor: shaping hot metal
- Flexible at beginning (exploration)
- Gradually more rigid as it "cools down" (exploitation)



For i=0 to N:

reduce temperature T
generate neighbor design
if better: go to neighbor
else: still go with chance relative to T

Simulated Annealing



Intelligent Text Entry

Daniel Buschek

Example Results

"boxpum" found with random search	"aero" found with Simulated Annealing
WPM: 31.97	WPM: 36.61
jkzf <u>boxpum</u>	xychtindkz
dnitcyrlv	bl <u>aero</u> fgj
haesgqw	vpmsuwq

Challenge – can you find a better layout than "aero"? Use the provided python notebook as a starting point.

Example Results

With a modified objective function

"chat" found with Simulated Annealing

WPM: 34.56

qjbfreoni

kwg<u>chat</u>

- z x y v p l d u s m What was this layout optimized for?
 - \rightarrow Typing with right thumb, reduce thumb stretching

Potential of Optimization-based Design

- Obtaining information on the design problem and a formal specification
- Exploring a large design space comprehensively
- Improving quality and robustness of designs
- Estimating possible improvements
- Supporting human designers
- Optimization during use, personalised UIs
- Requires: Models of user behaviour, formal problem definition / objective function, computational capacity, …

Questions & Discussion

- Name and explain the key components of optimizationbased UI design.
- How can designers influence obtained designs in this approach?
- Explain Simulated Annealing. Can you consider a design resulting from this method as "optimal"? Why (not)?
- If it is possible to find better designs than QWERTY, why are we not using them widely?
- Beyond keyboard layouts, which other UI design problems could be addressed with this approach? And which are hard to address in this way?

References

- Dunlop, M., & Levine, J. (2012). Multidimensional pareto optimization of touchscreen keyboards for speed, familiarity and improved spell checking. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2669–2678. <u>https://doi.org/10.1145/2207676.2208659</u>
- Gong, J., Xu, Z., Guo, Q., Seyed, T., Chen, X. "Anthony", Bi, X., & Yang, X.-D. (2018). WrisText: One-handed Text Entry on Smartwatch using Wrist Gestures. *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* - CHI '18, 1–14. <u>https://doi.org/10.1145/3173574.3173755</u>
- Oulasvirta, A., Reichel, A., Li, W., Zhang, Y., Bachynskyi, M., Vertanen, K., & Kristensson, P. O. (2013). Improving two-thumb text entry on touchscreen devices. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2765–2774. <u>https://doi.org/10.1145/2470654.2481383</u>
- Zhai, S., Hunter, M., & Smith, B. A. (2000). The metropolis keyboard—An exploration of quantitative techniques for virtual keyboard design. *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology UIST '00*, 119–128. <u>https://doi.org/10.1145/354401.354424</u>

Further Reading:

- Oulasvirta, A., Dayama, N. R., Shiripour, M., John, M., & Karrenbauer, A. (2020). Combinatorial Optimization of Graphical User Interface Designs. *Proceedings of the IEEE*, *108*(3), 434–464. https://doi.org/10.1109/JPROC.2020.2969687
- Simon, H. A. (1969). *The sciences of the artificial.* The MIT Press.

This file is licensed under the Creative Commons Attribution-Share Alike 4.0 (CC BY-SA) license:

https://creativecommons.org/licenses/by-sa/4.0

Attribution: Daniel Buschek

For more content see: https://iui-lecture.org

