Abstract
Smart phone and tablet usage has sharply increased for the last decade. While entering text on these devices, virtual keyboards are generally used instead of conventional hardware keyboards. In this study, a new problem which is two-finger keyboard layout problem and solution approach is presented for increasing user test entrance performance, especially on virtual keyboards. Defined two-finger keyboard layout problem is modeled as Quadratic Assignment Problem. Because of combinatorial structure of the problem a genetic algorithm is developed. Its result is given to mathematical model as initial solution for finding better solutions with mathematical model. Proposed approach is applied on Turkish language. The new two finger keyboard layout for Turkish language is compared with F and QWERTY keyboard layouts based on certain performance measurement techniques.

Keywords: Two-finger keyboard layout problem, quadratic assignment problem, genetic algorithm, virtual keyboard.

1. Introduction
Continuing development of smart phones and tablets has been making communication easier and faster recently. Therefore, usage of smart phone and tablet is increasing rapidly day by day. Generally, virtual keyboards are used to enter text document on tablets and smart phones. Literature search show that there are several type of keyboard designs for different languages such as Opti (Mac Kenzie & Zhang, 1999), Metropolis (Zhai et al., 2000), Ylarof (Li et al., 2006) and Fitaly (Fitaly, 2013) for English, AZERTY for French and F for Turkish. While developing keyboard design, certain criteria are considered, e.g. ergonomic factors or writing performance. Today keyboards, especially virtual ones, are used not only certain type of profession as writers or journalists but also used by adults to primary school students lots of kind of people which are using smartphone or tablet for social sites or blogs on web, chatting each other etc. But we can say that some users are novice and use less than 10 fingers while writing. And lots of them can be use only two finger on virtual keyboard due to dimensional restrictions of smartphones and tablets. Therefore in this study new type of keyboard arrangement problem which is two-finger keyboard layout problem is defined, and methodology presented to solve the problem. Purpose of new design problem is ensuring to enter text fast and with less finger movements. Problem has been modelled as a type of QAP. Nevertheless the optimal solution is difficult by using exact algorithms because of complexity.
of QAP which is NP-hard. Therefore, in literature several heuristic methods such as taboo search algorithm, genetic algorithm, simulated annealing, particle swarm optimization method and ant colony algorithm, are developed to find solution QAPs (Burkard, 2013). In this study, a simple genetic algorithm (GA) is developed to solve the problem. Defined problem is solved for Turkish language and found keyboard is compared with F and QWERTY keyboards.

The paper is arranged as follows; in the second section two finger keyboard layout problem is defined and QAP model is presented. Two finger stylus keyboard (Ç) for Turkish language is designed in the third section. Proposed Ç keyboard, F keyboard and QWERTY keyboard are compared with each other in the fourth section. The last section is conclusion and future studies.

2. Problem Definition

For the two finger keyboard layout problem, standard longitudinal shape keyboard is considered as two part, left and right. When entering text, it is assumed that users use only one the left and right fingers on the left and right parts, respectively (see Figure 1). Objective of problem is minimization of sum of finger movements on two keyboard parts.

![Figure 1. Traditional Longitudinal Keyboard Layout](image)

The problem is modeled as a type of QAP similar to single-finger keyboard problem. Assumptions of model are given below.

- Keys for assigning to locations are known and fixed.
- Keyboard layout and locations on keyboard to assign keys are fixed and known.
- Euclidean distance is taken for distance between locations.
- Key digraph matrix is deterministically given for language.
- Keyboard is separated to two as left and right part.
- While entering text, keyboard user uses left and right hand for keys on left and right parts respectively.

Under these assumptions, the problem is to assign keys to locations such that total movements of fingers will be minimized. Developed QAP model is given below.

\[
\min \sum_{m=1}^{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \sum_{l=1}^{n} C_{ik} \cdot D_{jl} \cdot X_{Y_{imj}} \cdot X_{Y_{kml}}
\]

Subject to

\[
\sum_{m=1}^{2} Y_{im} = 1 \quad \text{for all } i
\]

\[
\sum_{j=1}^{n} X_{Y_{imj}} - Y_{im} = 0 \quad \text{for all } i, m
\]
\[ \sum_{i=1}^{n} XY_{imj} = 1 \quad \text{for all } j, m \]  
(4)

\[ XY_{imj}, Y_{im} \in \{0,1\} \]  
(5)

There are two variables, \( Y_{im} \) and \( XY_{imj} \). \( Y_{im} \) takes 1 if key i is assigned to part m, otherwise takes 0. In a similar way, \( XY_{imj} \) takes 1 if key i is assigned to location j on part m. Each value on key digraph matrix, \( C_{ik} \), shows that number of times of key k comes consecutively after key i in selected texts. \( D_{jl} \) is Euclidean distance between location j and l. Objective function minimizes sum of all movements based on \( C_{ik} \) and \( D_{jl} \) matrixes. Constraint 2 ensures that each letter must be assigned to one part, left or right. If \( Y_{im} \) takes 1, key i assigned to one location on part m with Constraint 3. Constraint 4 provides that one key must be assigned to each location on each part. Last constraint (5) is the the binary constraints.

3. Designing Two Finger Keyboard Layout for Turkish Language

Special keyboard design for Turkish language has been firstly designed by İhsan Sıtkı Yener in 1955 which is called as F keyboard. F keyboard is designed for ten fingers and has been widely used in Turkey with state support. On the other hand, stylus keyboard for Turkish language has been firstly presented by Üssaklı (2004). Then Soydal (2010) presented another design. Two new designs are developed for single finger. In this study, a new two finger keyboard layout (TFKL) for Turkish is designed for novice users to type fast with two fingers. There are five steps to design TFKL for any language. These steps are presented in Figure 2.

![Figure 2. Steps of Designing TFKL for Turkish Language](image)

**Step 1. Data Supplementary**

First step of designing TFKL is data supplementary. Data supplementary has two parts. One part of data supplementary is frequency number of characters. The other part of data supplementary is distance between keys. These two sub-steps are explained below in detail.

**Flow matrix**: Key digraph matrix, \( C_{ik} \), is formed in this step. Turkish language have 29 letters, they are shown below. Letters out of these (q, w, x), space bar and punctuation characters are ignored.

**Set of letters for Turkish language**= \{a, b, c, ç, d, e, f, g, ğ, h, i, j, k, l, m, n, o, ö, p, r, s, ş, t, u, ü, v, y, z\}

Base texts for generating digraph are selected from Turkish web blogs and several recent published books. The result of this step, 29x29 dimensions asymmetric matrix for Turkish language is established and given in Appendix A.
**Distance matrix:** Distance matrix, $D_{ij}$, is calculated by using Euclidian distance. In this study, the geometrical shape of keyboard is standard longitudinal shape. Two zones with 15 and 14 key locations are determined for possible assignment of 29 Turkish language letters as Figure 3. Distance between center points of two horizontal adjacent keys is one unit. Vertical distance of two rows is also one unit. For example, distance between key 1 and key 2 one unit, key 1 and key 6 is $\sqrt{1,25}$. Full distance matrix is given in Appendix B.

![Figure 3. New Proposed Keyboard Layout.](image)

**Step 2. Quadratic Assignment Problem and Solution**

There are two steps to solve TFKL problem. These are genetic algorithm (GA) methodology and exact algorithm. Basic GA structure is used and coded in MATLAB, steps of algorithm and selected parameters can be found in Appendix C. Mathematical model is coded in General Algebraic Modeling System (GAMS). Firstly, TFKL problem is solved with GA. In second stage, best solution of GA is given to the mathematical model as initial solution to find better solution. Mathematical model is solved with GAMS-Cplex 12.3 which is run two week. But solution is not improved due to NP-hard structure of problem. So, GA results are taken as a solution for the design.

**Step 3. Proposed design for two finger stylus keyboard layout**

The best arrangement of TFKL problem for Turkish language is given in Figure 4. The name of proposed model is Ç keyboard. This comes from its first character. All vowel letters are located in right side. The reason of this can be that vowel and consonant letters generally follows each other in Turkish language.

![Figure 4. Proposed design for two finger stylus keyboard layout.](image)
Step 4. Packages and Extra Module

In this step, Microsoft and Android applications of Ç keyboard are developed to apply usability test for participants. There are two screenshots for android application for Ç keyboard. These are vertical and horizontal screenshots in Figure 6 and Figure 7, respectively. Microsoft application screenshot is available in Figure 5.

Figure 5. Screenshot of Microsoft Application of Ç Keyboard

Figure 6. Vertical layout on Android application of Ç keyboard
4. Discussions and Results

Objective function values and usability analysis are used to compare Q, F and Ç keyboards. Values of objective function of mathematical model are calculated using Equation (1) for Ç, F and Q keyboards, which are shown in Table 1.

Table 1. Objective Function Values of Keyboards

|           | Ç Keyboard | F Keyboard | Q Keyboard |
|-----------|------------|------------|------------|
| Objective Values | 202,601    | 310,723    | 825,800    |

Ç keyboard is better than other two keyboards according to objective function values. Q keyboard is the worst because its development process does not include any issue about Turkish language.

Usability is defined as: “the extent to which a product can be used by specified users to achieve specified goals with effectiveness; the extent to which the intended goals of use are achieved, efficiency; the resources that have to be expended to achieve the intended goals and satisfaction; the extent to which the user finds the use of the product acceptable, in a specified context of use” (ISO 9241-11,1994). There are three usability evaluation methods in literature. One of these methods is usability test. The usability test requires sample participant to work on typical tasks using the system. The evaluators use the results to see how the user interface supports the users to do their tasks. In this section, Ç, F and Q keyboards are compared based on efficiency and effectiveness using usability test. The layouts are depicted to the subjects on the test software for Android system. It is said these subjects that “both accuracy and typing speed is important for this experiment”. Also, it is clearly implied to them that the time measurements will not be shown to the other subject, in order to prevent a competition between the subjects. Each participant was given oral instructions explaining the task and the goal of the experiment. They were asked specifically to aim for both entry speed and accuracy. The instructions also stated that if they made an error, that try will be repeated. In other words, a session is repeated if the text typed in that session has missing, extra or different characters when compared with the test material. Hence, after each session, error
rate should be zero. Three tasks are applied to practical usability test. First task is to write specific text which has 31 words and 212 characters. Second task is to enter a tweet which has 20 words and 140 characters. Third task is to write e-mail text which has 29 words and 201 characters. Results are shown in Table 2.

| Participants | Task 1 | Task 2 | Task 3 |
|--------------|--------|--------|--------|
|              | Q      | F      | Ç      | Q      | F      | Ç      | Q      | F      | Ç      |
| Participant 1| 169    | 529    | 402    | 148    | 288    | 201    | 180    | 596    | 459    |
| Participant 2| 176    | 317    | 311    | 164    | 471    | 388    | 189    | 437    | 378    |
| Participant 3| 150    | 446    | 283    | 158    | 428    | 372    | 169    | 574    | 342    |
| Participant 4| 210    | 477    | 384    | 178    | 443    | 309    | 243    | 547    | 442    |
| Participant 5| 162    | 510    | 420    | 152    | 468    | 361    | 175    | 597    | 463    |
| Participant 6| 292    | 403    | 371    | 236    | 378    | 327    | 261    | 422    | 396    |
| Participant 7| 115    | 250    | 205    | 97     | 231    | 140    | 126    | 376    | 260    |
| Participant 8| 152    | 398    | 267    | 141    | 458    | 346    | 168    | 521    | 351    |
| Participant 9| 171    | 365    | 281    | 198    | 434    | 338    | 188    | 466    | 361    |
| **AVERAGE**  | 177,444| 410,555| 324,888| 163,555| 399,888| 309,111| 188,777| 504    | 383,555|

According to Table 2, Q keyboard is best keyboard for all tasks because it is the most known and used keyboard by the users. But when the two unfamiliar keyboards are compared, we see that new design is significantly better than other for three tasks.

5. Conclusion and Future Studies
In this study, two finger keyboard layout problem is defined and mathematically modelled as QAP. Defined problem is solved for Turkish language. The proposed design, Ç keyboard layout, is compared with well-known Q keyboard and F keyboard which is the first keyboard designed for Turkish language, according to QAP model objective value and usability test. Although Ç keyboard has better objective value than two keyboards, it is usability test results are worse than Q keyboard. Because finding inexperienced Q keyboard user is pretty hard. In future studies TFKL problem can be solved for other languages. Also new techniques can be developed to solve the problem efficiently.
REFERENCES

MacKenzie, I.S. & Zhang, S.X., (1999). The design and evaluation of a high-performance soft keyboard. In Proceedings of CHI'99: ACM Conference on Human Factors in Computing Systems. p. 25-31.

Zhai, S., Hunter, M., & Smith, B.A., (2000). The Metropolis Keyboard - an exploration of quantitative techniques for virtual keyboard design. in Proceedings of The 13th Annual ACM Symposium on User Interface Software and Technology (UIST). San Diego, California: ACM. p. 119-218.

Li, Y., Chen, L., & Goonetilleke, R.S., (2006). A heuristic based approach to optimize keyboard design for single-finger keying applications. International Journal of Industrial Ergonomics, 36(8):695-704.

Fitaly, (2013). The fitaly one-finger keyboard. http://www.fitaly.com/fitaly/fitaly.htm (Accessed on October 2013).

Burkard, R.E., (2013). Quadratic Assignment Problems. Handbook of Combinatorial Optimization. Springer New York, pp 2741-2814. DOI: 10.1007/978-1-4419-7997-1_22.

Soydal, M., (2010). Designing Optimum Stylus Keyboard Layout for Turkish Language Using a Decomposition-Based Heuristic. Graduate Program in Industrial Engineering, Boğaziçi University.

Uşşaklı, S., (2004). Performance Optimization of Turkish Virtual Keyboard. A thesis submitted to the department of Computer Engineering and the Institute of Engineering and Science of Bilkent University.

ISO 9241-11, 1994. Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs)—Part 11: Guidance on Usability.
### Character Transition Frequency (CTF) of Flow Matrice for Turkish Language

| A   | B   | C   | D   | E   | F   | G   | H   | I   | J   | K   | L   | M   | N   | O   | Ö   | P   | R   | S   | T   | U   | Ü   | V   | Y   | Z   |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 459 | 272 | 483 | 914 | 4589| 20 | 1048| 305 | 1711| 2591| 4   | 5   | 589 | 689 | 760 | 8095| 3515| 949 | 98  | 2   | 2143| 16610| 4270 | 2972| 4706| 153 | 243 | 514 |
| 229 | 246 | 298 | 2089| 972 | 3205| 148 | 354 | 128 | 2010| 1082 | 17 | 138 | 528 | 409 | 409 | 528 | 609 | 3510| 11952| 195 | 0 | 410 | 16683| 3745 | 1108 | 4207 | 167 | 2 | 1448 | 4105 | 197 |
| 402 | 3 | 4 | 2 | 4 | 953 | 6 | 4 | 3 | 162 | 2968 | 20 | 1282 | 4 | 13 | 346 | 7 | 3 | 7 | 0 | 55 | 7 | 2 | 1499 | 108 | 0 | 10 | 475 | 339 | 377 | 1 |
| 3 | 4 | 586 | 4 | 0 | 586 | 4 | 0 | 0 | 0 | 586 | 2968 | 20 | 1282 | 4 | 13 | 346 | 7 | 3 | 7 | 0 | 55 | 7 | 2 | 1499 | 108 | 0 | 10 | 475 | 339 | 377 | 1 |
| 642 | 4 | 0 | 586 | 4 | 4 | 586 | 4 | 8 | 586 | 3410 | 20 | 1172 | 2 | 3 | 228 | 4 | 12 | 3 | 40 | 7 | 0 | 55 | 7 | 2 | 1499 | 108 | 0 | 10 | 475 | 339 | 377 | 1 |
| 642 | 4 | 0 | 586 | 4 | 4 | 586 | 4 | 8 | 586 | 3410 | 20 | 1172 | 2 | 3 | 228 | 4 | 12 | 3 | 40 | 7 | 0 | 55 | 7 | 2 | 1499 | 108 | 0 | 10 | 475 | 339 | 377 | 1 |
| 642 | 4 | 0 | 586 | 4 | 4 | 586 | 4 | 8 | 586 | 3410 | 20 | 1172 | 2 | 3 | 228 | 4 | 12 | 3 | 40 | 7 | 0 | 55 | 7 | 2 | 1499 | 108 | 0 | 10 | 475 | 339 | 377 | 1 |
| 642 | 4 | 0 | 586 | 4 | 4 | 586 | 4 | 8 | 586 | 3410 | 20 | 1172 | 2 | 3 | 228 | 4 | 12 | 3 | 40 | 7 | 0 | 55 | 7 | 2 | 1499 | 108 | 0 | 10 | 475 | 339 | 377 | 1 |

**Character Transition Frequency (CTF)**

OR FLOW
## Appendix B. Distances Between Two Keys of Two Finger Keyboard Layout

|   | 1   | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  | 11  | 12  | 13  |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 0 | 1   | 2   | 3   | 4   | 1.12| 1.8 | 2.69| 3.2 | 4.61| 2.24| 2.83| 3.61|
| 1 | 0   | 1   | 2   | 3   | 1.12| 1.12| 1.8 | 2.69| 3.2 | 2   | 2.24| 2.83|
| 2 | 1   | 0   | 1   | 2   | 1.8 | 1.12| 1.12| 1.8 | 2.69| 2.24| 2   | 2.24|
| 3 | 2   | 1   | 0   | 1   | 2.69| 1.8 | 1.12| 1.12| 1.8 | 2.83| 2.24| 2   |
| 4 | 3   | 2   | 1   | 0   | 3.2 | 2.69| 1.8 | 1.12| 1.12| 3.61| 2.83| 2.24|
| 12| 1.12| 1.8 | 2.69| 3.2 | 0   | 1   | 2   | 3   | 4   | 1.12| 1.8 | 2.69|
| 1.8|1.12|1.12|1.8 |2.69|1 |0 |1 |2 |3 |1.12|1.12|1.8 |
| 0.69|1.8 |1.12|1.12|1.8 |2 |1 |0 |1 |2 |1.8 |1.12|1.12|
| 3.2|2.69|1.8 |1.12|1.12|3 |2 |1 |0 |1 |2.69|1.8 |1.12|
| 0.61|3.2 |2.69|1.8 |1.12|4 |3 |2 |1 |0 |3.2 |2.69|1.8 |
| 2.4|2.24|2.24|2.83|3.61|1.12|1.12|1.8 |2.69|3.2 |0   |1   |2   |
| 0.83|2.24|2   |2.24|2.83|1.8 |1.12|1.12|1.8 |2.69|1   |0   |1   |
| 0.61|2.83|2.24|2   |2.24|2.69|1.8 |1.12|1.12|1.8 |2   |1   |0   |
| 0.47|3.61|2.83|2.24|2   |3.2 |2.69|1.8 |1.12|1.12|3   |2   |1   |
| 0.39|4.47|3.61|2.83|2.24|4.61|3   |2.69|1.8 |1.12|4   |3   |2   |

Distances between two keys of two fingers keyboard layout.
APPENDIX C. STEPS OF GENETIC ALGORITHM AND SELECTED PARAMETERS

Genetic algorithm is well-known evolutionary heuristic technique, detailed information can be found at Goldberg (1989)\(^1\). Pseudo-code of GA with used parameters is given below.

1. Generate randomly 100 solutions. Select the best 10 as a population.
2. Repeat
3. Select parents with tournament method
4. Apply two-point crossover for generate children with 0.95 crossover probability
5. Apply one point mutation with 0.01 mutation probability
6. Generate the new generation using elitist selection method
7. Until iteration number is two thousand

Also permutation encoding is used for the solution representation. Fitness function of GA is same as mathematical model objective function.

---

\(^1\) Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization, and Machine Learning (1 edition). Reading, Mass: Addison-Wesley Professional.