International Journal of Modern Manufacturing Technologies
ISSN 2067–3604, Vol. XV, No. 3 / 2023
https://doi.org/ 10.54684/ijmmt.2023.15.3.67

A SYSTEM FOR AUTOMATIC DIMENSIONING OF MECHANICAL DRAWINGS

Konstantinos G. Kakoulis, Petros Konstantinidis, Athanasios Manavis, Panagiotis Kyratsis

University of Western Macedonia, Department of Product and Systems Design engineering,
Koila Kozani GR50100, Greece

Corresponding author: Konstantinos G. Kakoulis, kkakoulis@uowm.gr

Abstract: One essential aspect of any CAD system, which has not been addressed adequately, is the capability to support the automatic placement of dimensions of a mechanical drawing. Today, even the most sophisticated CAD systems automatically place dimensions of mechanical drawings without strictly adhering to standard dimensioning rules (e.g., there are usually overlaps, repetitions and misplacement of dimensions). Therefore, the automatically generated dimensions are the starting point for a manual improvement of the dimension positions. In this paper we present a software system for the automatic placement of dimensions in mechanical drawings, incorporated into a commercial CAD system, which is efficient and places dimensions that comply with the basic drawing standards, assigns dimensions to all elements of a drawing, and does not produce redundant dimensions.

Key words: CAD, automatic dimensioning, automatic label placement, technical drawings, mechanical drawings.

1. INTRODUCTION

Mechanical drawings are essential parts of an international technical language that allows communication between people for the construction, quality control and assembly of products. In many CAD systems it is possible to create 2D Views from 3D models automatically. Even though dimensioning of Mechanical drawings is an essential component of mechanical product design, these systems automatically place dimensions, however, without strictly adhering to standard dimensioning rules. There are usually overlaps, repetitions and misplacement of dimensions. This fact forces the user to be involved in correcting and completing the dimensioning, a process that is even more prone to errors.

The text or symbol annotation of information is an important task in information visualization [8], Cartography [10], GIS [16], Graph Drawing [11, 30], and Engineering [9].

The research addressing the annotation of information problem has been primarily directed towards the automatic placement of labels to elements of geographical maps [4, 13, 22, 31] and graph drawings [12, 23, 24]. The placement of labels on the boundary of a drawing or a diagram, where each element is connected to its associated label through a line segment, called leader, has been studied in [3, 6, 25].

For the automatic placement of dimensions to mechanical drawings most of the research has been focused on placing dimensions for 2D mechanical drawings that are created automatically from their 3D models. We will refer to this as the *Automatic Dimensioning Placement* (ADP) problem.

Requicha [29] introduced an algorithm for dimensioning drawings corresponding to 3D models created with the Constructive Solid Geometry (CSG). Its application is limited because in CSG the geometric objects must be defined procedurally and built from set-theoretic combinations of primitive 3D models. Additional work on dimensioning objects created with CSG can be found in [14, 15, 34]. Hashimoto et al. [18] presented an algorithm for automatic dimensioning of 2D drawings. Only linear and radial dimensions were covered and no adequate checking of overlapping dimensions was provided. Bond and Ahmed [7] introduced a rule-based system based on domain knowledge. Early attempts in automatic dimensioning have been summarized in [33].

Chen et al. [9] introduced a technique that uses a rule-based expert system approach. Dimensions are placed one at a time. Each dimension that is placed is associated with a region. Each subsequent dimension cannot be placed in a region associated with an already placed dimension. The limitation of this method is due to its sequential nature.

Ren et al. [28] presented a method for the automatic dimensioning of shear walls in construction drawings. This method places dimensions only for simple rectangular shapes.

Goyal et al. [17] introduced an algorithm that places the dimensions on an engineering drawing, after removing unwanted lines, which usually result when the drawing is scanned. Raffaeli et al. [27] developed a method that generates automatic dimensioning of engineering drawings. This method identifies the geometric entities from the dimensions used to design the 3D object, which are retrieved through dedicated search algorithms based on the recognition of the geometric features of the input model. Both techniques place dimensions to all elements of the drawings, however, they do not optimize the positions of the dimensions.
Kakoulis and Siozos [20] presented techniques for automatically placing dimensions in 2D technical drawings. These techniques can be used with any type of 2D technical drawings (e.g., architectural, engineering, or topographical drawings). Because they do not derive the actual dimensions from 3D objects, they can be used to dimension technical drawings produced by scanning technical drawings printed on paper. Additionally, their approach is fast, efficient, follows basic drawing standards, does not produce redundant dimensions, and guarantees that all graphical elements in the technical drawings will have dimensions.
In this paper we present a software system for the automatic placement of dimensions in mechanical drawings, incorporated into a commercial CAD system, which is efficient and places dimensions that comply with the basic drawing standards, assigns dimensions to all elements of a drawing, and does not produce redundant dimensions.
Our paper is organized as follows: In Section 2 we define the ADP problem and we describe a software prototype for the automatic dimensioning of mechanical drawings. In Section 3 we present and compare the results of our system with other systems that offer automatic placement of dimensions, and we conclude in Section 4.

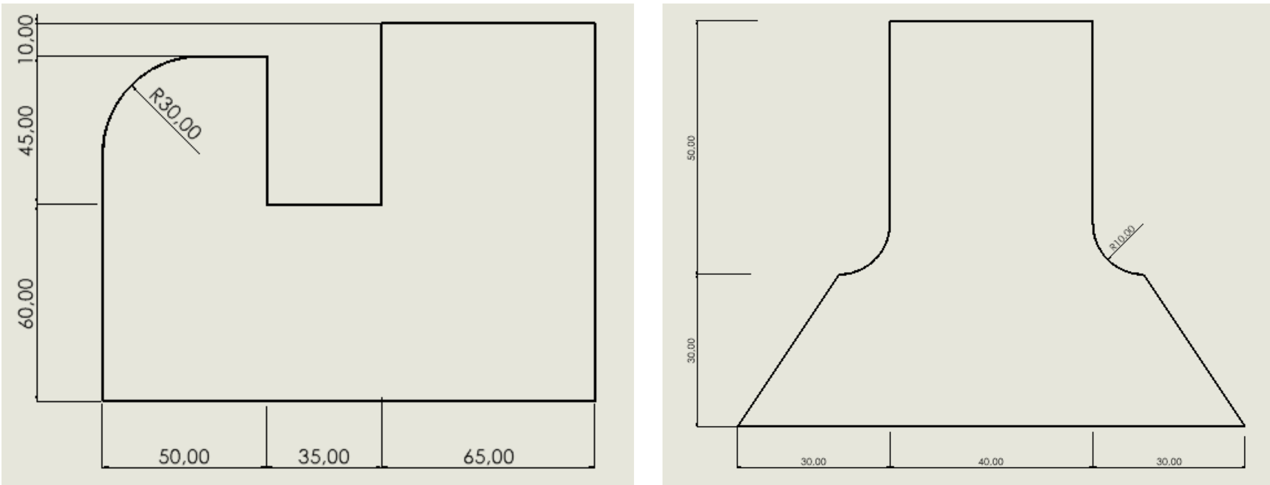


Fig. 1. Drawings with dimensions produced by our Software System

2. MATERIALS AND METHODS

Dimensioning drawings is about communicating in a clear and accurate fashion shape and size of the objects being drawn. Since dimensions are the means for describing the exact geometry of a part to be manufactured, the rules for placement of dimensions are defined by thorough international drawing standards [1, 2]. Some of the basic dimensioning principles are:

- Dimensions should be placed as close to their element as possible.
- Dimensions should not be duplicated, or the same information given in two different ways.
- Dimensions should not overlap.
- Avoid dimensioning over or through the object.
- Dimension lines should not cross.
- Dimension lines and extension lines should not cross.
- It is acceptable for extension lines to cross one another.
- Extension lines do not touch the feature of the drawing they refer to.
- Dimension figures should be approximately centered between the arrowheads.
- Group dimensions together to make them easy to read.
- For interior elements place the dimensions on the periphery.
- Dimensions should be spaced uniformly throughout the drawing.

For the ADP problem the goal is to place the dimensions of all the graphical features of a drawing in the best possible way. What constitutes a good label assignment has its roots in the art of cartography. For many centuries cartographers have perfected the art of placing labels on maps. It is difficult to quantify all the characteristics of a good label placement, because they reflect human visual perception and intuition. In a successful label assignment, labels must be positioned such that they are legible and follow basic aesthetic quality criteria [19, 32]: Clarity: A label must not overlap other labels or graphical features; Readability: A label must have legible size; Associativity: A label must be easily identified with exactly one graphical feature and Quality: A label must be placed in the most preferred position, among all legible positions.

In the context of technical drawings, dimensioning includes the placement of dimension text, dimension lines, and extension lines customized for a single viewpoint. Dimension lines are line segments that indicate the extent of a feature, while extension lines visually connect the ends of a dimension line to the relevant feature in the model (Figure 2).

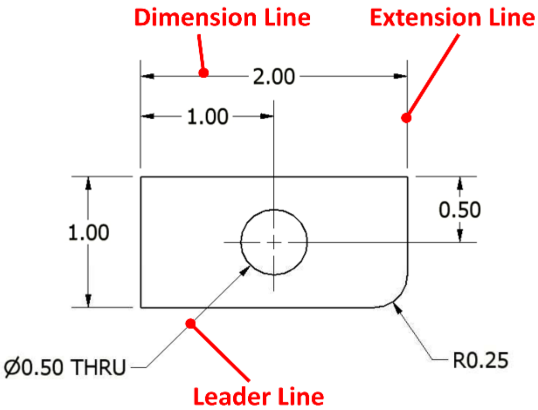


Fig. 2. Types of dimension lines.

The ADP problem is an optimization problem where the objective is to find a dimensioning assignment of minimum total cost. Each dimension that is part of a final assignment may be associated with a cost which reflects the severity of the violation of the basic placement rules established mainly by field experts. In addition, for the placement of dimensions we may apply some optimization criteria, such as [20]: Minimize the total number of dimensions on each side; Minimize the total length of the main dimension lines and Minimize the total length of the extension dimension lines.

The ADP problem is a computationally hard problem, and its computational complexity has been studied in [20]. The problem of automatically placing labels to graphical features is typically NP-hard [4, 5, 21, 24, 26]. Even the problem of deciding on which sides of the bounding rectangle of a drawing the dimensions must be placed, which corresponds to the Partition problem, is an NP-complete problem. Because the automation of the placement of labels is a very difficult problem, we rely on heuristics to provide practical solutions for real world problems.

Next, we describe a polynomial time algorithm for the ADP problem that was introduced in [20].

The basic idea of the algorithm is to: (i) Project the end vertices of the line segments to the bounding rectangle *R* of the input technical drawing *G*, (ii) eliminate redundant dimensions, and (iii) place the dimensions around *R* in a well-balanced way and as close to their annotated features as possible.

In a preprocessing step the bounding rectangle R and the center of gravity *P* of a drawing *G* are calculated. Next, the side of the bounding rectangle *R* where the dimension of a line segment *s* will be placed is decided by comparing the *x* and *y* coordinates of the midpoint of *s* with the *x* and *y* coordinates of a point *P* that corresponds to the center of gravity of *G*. For example, if $s_x < P_x$ *and* $s_y < P_y$, then the end vertices of *s* will be dimensioned to the left and bottom sides of the bounding rectangle of *G*. Then, four projection lists are defined that correspond to the four sides of *R*. The projection points of the end vertices of the line segments are stored in these lists. Next, redundant projection points are removed from the lists to ensure that no duplicate dimensions are present. Finally, circles and arcs are dimensioned. In [20] it is shown that this algorithm assigns dimensions to each graphical feature, without duplications and overlaps, of the input mechanical drawing *G* in $O(n \log n)$ running time.

2.1. A System for the ADP problem

In this section we present a software system for the automatic placement of dimensions in mechanical drawings. The main idea of this software system is to provide the user with a simple and easy procedure to dimension automatically mechanical drawings. The user selects a 3D mechanical object, which has been modeled with

SolidWorks™ and automatically creates its corresponding 2D views along with their dimensions. The dimensions produced follow the basic drawing standards. In addition, all graphical features in the 2D views will be dimensioned and there will be no redundant dimensions. Furthermore, dimensions are placed in all standard 2D views without duplication of dimensions in different views.
This software system has been developed in Visual Basic™ and incorporated into a commercial CAD system (SolidWorks™) through its Application Programming Interface (API).
The core engine of this software system is based on an algorithm presented in [20]. The software dimensions lines, arcs, chamfers, ellipses, and slots.
This software system has the following main steps:

- The user selects a 3D object, which has been modeled with SolidWorks™.
- The standard 2D views of the input 3D object are created automatically in a Drawing Sheet.
- Each 2D view automatically in a Drawing Sheet is dimensioned automatically.

First, when the software system starts a user form appears, which allows the user to select the 3D object and the size of the Drawing Sheet (see Figure 3).
Then the standard 2D views of the input 3D object are created in a Drawing Sheet and the dimensions are placed (see Figure 4).
The dimensions of the mechanical drawings in Figures 1 and 3-10 have been placed by the implemented software. Special care was taken for the correct dimensioning of arcs, circles (see Figures 5, 7, 9 and 10) and chamfers (see Figure 6). For arcs, the dimensions are placed between the center point and the midpoint of the arc if there is enough space. Otherwise, the dimensions are placed above the midpoint of the arc.

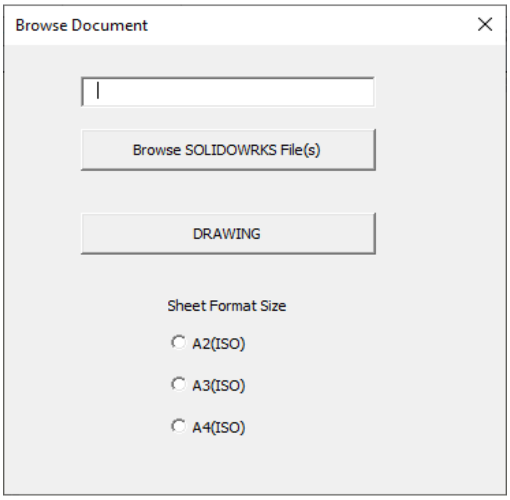


Fig. 3. A user form to select a 3D object

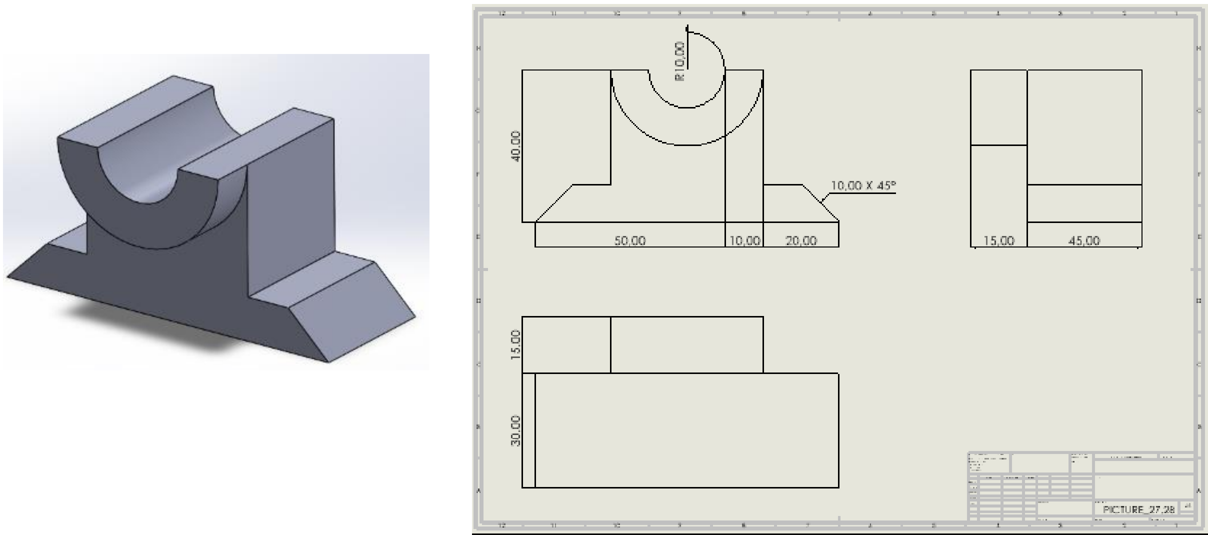
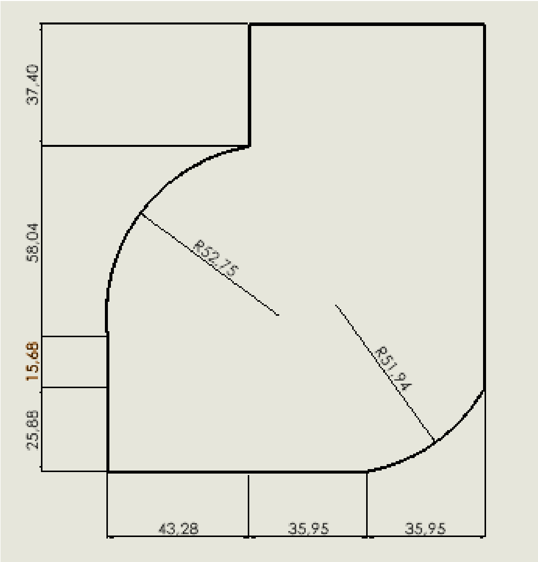


Fig. 4. Creation and dimensioning of the standard 2D views



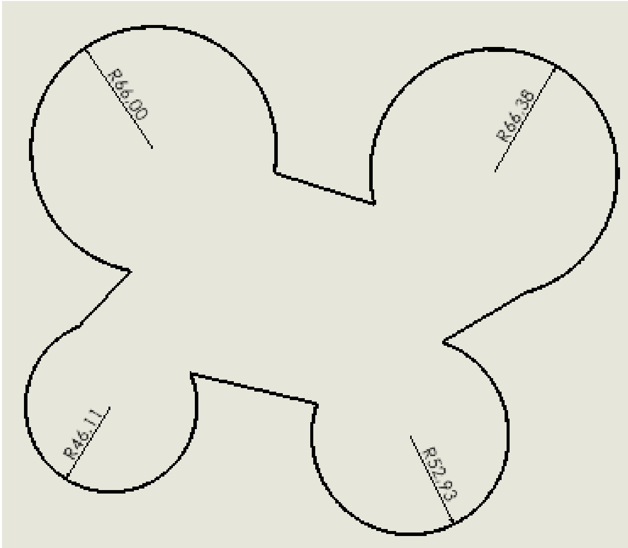


Fig. 5. Dimensioning of arcs

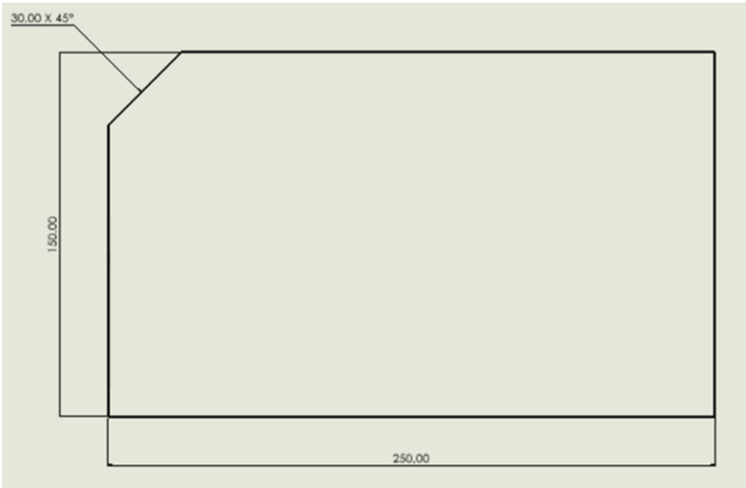
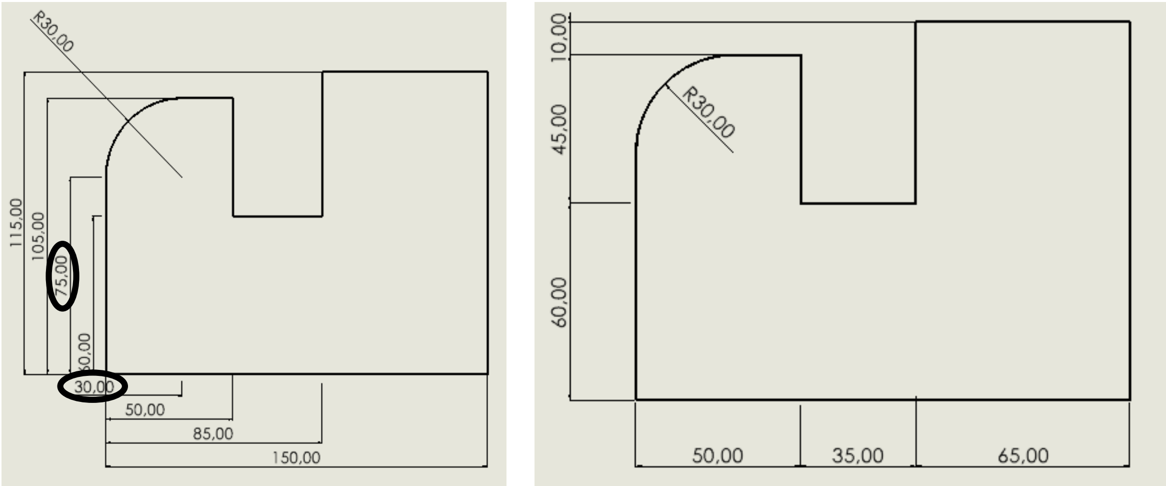


Fig. 6. Dimensioning of chamfers

3. RESULTS AND DISCUSSIONS

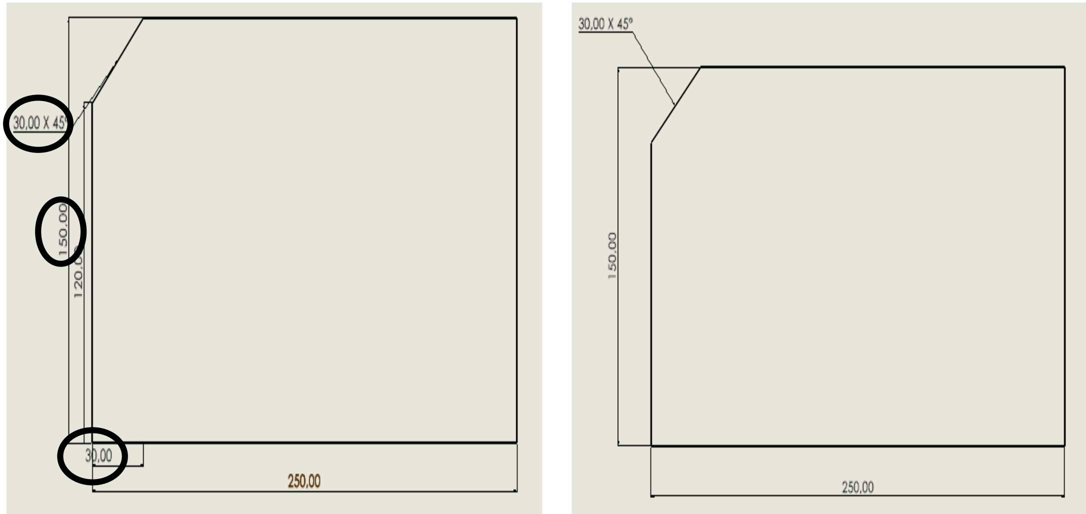
Many CAD systems have the capacity to place dimensions to two-dimensional engineering drawings automatically. For example, SolidWorks™ places dimensions through the *autodimension* command. From our experiments it is clear that our software system creates better placement of dimensions than SolidWorks™. The main advantages that our dimensioning system offers can be summarized in the following:

- Adjusts the size of the dimensions automatically in order to be readable. Often, the *autodimension* command produces dimensions of small size that need to be enlarged manually.
- Does not produce redundant dimensions. However, in Figures 7a and 9a the *autodimension* command dimensions the same arc three times. Also, in Figure 8a the chamfer dimension is repeated three times.
- Does not produce dimensions that overlap each other. However, in Figure 7a after the *autodimension* command the horizontal dimension 30 and the vertical dimension 60 overlap dimension lines, in Figure 8 the horizontal dimension 30 overlaps a dimension line. Also, in Figure 10a dimensions are inside the view and overlap dimension lines.
- Dimensions each element of the drawing. However, in Figure 10a the *autodimension* command does not provide the dimensions of the ellipse.
- Does not repeat the same dimension in different views of the same drawing. However, the *autodimension* command usually dimensions each 2D view independently from any other view in the drawing.



a **b**

Fig. 7. (a) Dimensions are placed by SolidWorks™. (b) Dimensions are placed by our software

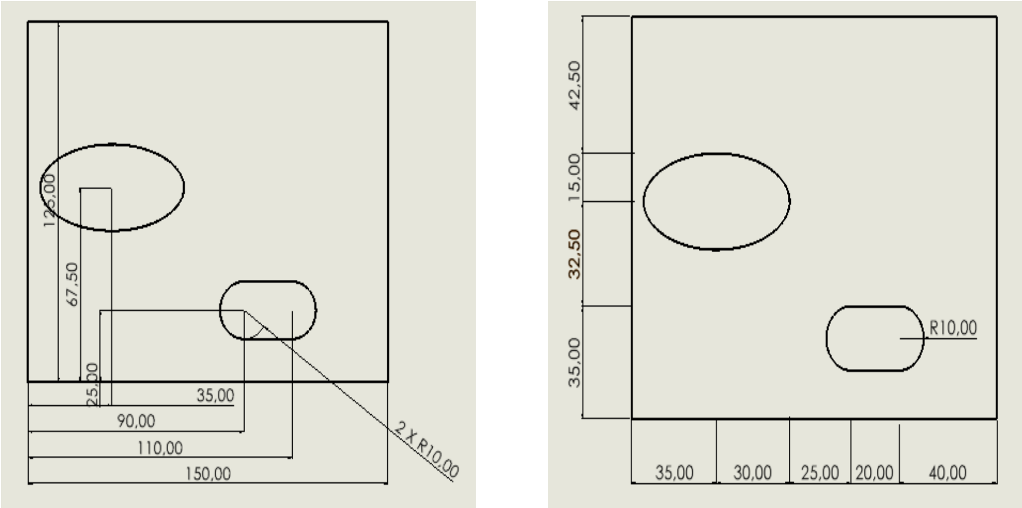


a **b**

Fig. 8. (a) Dimensions are placed by SolidWorks™. (b) Dimensions are placed by our software

a **b**

Fig. 9. (a) Dimensions are placed by SolidWorks™. (b) Dimensions are placed by our software



a b

Fig. 10. (a) Dimensions are placed by SolidWorks™. (b) Dimensions are placed by our software

4. CONCLUSIONS

Mechanical drawings continue to play an important role in the manufacturing process. Automation of the placement of dimensions to mechanical drawings is very desirable, since the manual placement of dimensions is a tedious, time-consuming and error prone process. Even today, state of the art CAD systems produces automatic placement of dimensions, which needs to be improved manually. In this paper we have presented a software system for the automatic placement of dimensions to graphical elements of mechanical drawing, which was developed and incorporated into a commercial CAD system (SolidWorks™). It places dimensions with higher accuracy than existing CAD systems, with guaranteed quality and without the need of manual improvement of the placed dimensions.

REFERENCES

1. ASME y14.5. Dimensioning and Tolerancing: Engineering Drawing and Related Documentation Practices (https://www.asme.org/codes-standards/find-codes-standards/y14-5-dimensioning-tolerancing). 2018.

2. ISO 128-1. Technical drawings: General principles of presentation (https://www.iso.org/obp/ui/#iso:std:iso:128:-1:ed-1:v1:en). 2003.

3. Ali, K., Hartmann, K., Strothotte, T., (2005). *Label Layout for Interactive 3D Illustrations*. Journal of the WSCG, 13(1), 8-20.

4. Been, K., Daiches, E., Yap, C., (2006). *Dynamic map labeling*, IEEE Transactions on Visualization and Computer Graphics, 12(5), 773-780.

5. Bekos, M. A., Kaufmann, M., Papadopoulos, D., Symvonis, A., (2011). *Combining traditional map labeling with boundary labeling*. In I. Cerna, T. Gyimothy, J. Hromkovic, K. Jefferey, R. Kralovic, M. Vukolic, and S. Wolf, editors, 37th International Conference on Current Trends in Theory and Practice of Computer Science (Proc. SOFSEM 2011), pp. 111-122. Springer-Verlag.

6. Bekos, M. A., Kaufmann, M., Symvonis, A., Wolff, A, (2007). *Boundary labeling: Models and efficient algorithms for rectangular maps*, Computational Geometry, 36(3), 215-236.

7. Bond, A., Ahmed, S., (1989). *Knowledge-based automatic dimensioning*, Artificial Intelligence in Engineering, 4(1), 32-40.

8. Bernard Chazelle et al., (1999). *Application challenges to computational geometry: CG impact task force report*. In B. Chazelle, J. E. Goodman, and R. Pollack, editors, Advances in Discrete and Computational Geometry, volume 223 of Contemporary Mathematics, 407-463. American Mathematical Society, Providence.

9. Chen, K., Feng, X., Lu, Q., (2001). *Intelligent dimensioning for mechanical parts based on feature extraction*, Computer-Aided Design, 33(13), 949-965.

10. Christensen, J., Marks, J., Shieber, S., (1995). *An empirical study of algorithms for Point Feature Label Placement*, ACM Transactions on Graphics, 14(3), 203-232.

11. Di Battista, G., Eades, P., Tamassia, R., Tollis, I. G., (1999). *Graph Drawing: Algorithms for the Visualization of Graphs*, Prentice Hall, Upper Saddle River, NJ.

12. Dogrusoz, U., Kakoulis, K. G., Madden, B., Tollis, I. G., (2007). *On Labeling in Graph Visualization*, Special Issue on Graph Theory and Applications, Information Sciences Journal, 177(12), 2459-2472.
13. Edmondson, S., Christensen, J., Marks, J., Shieber, S., (1997). *A General Cartographic Labeling Algorithm*, Cartographica, 33(4), 321-342.
14. Fitzgerald, W., Gracer, F., Wolfe, R., (1981). *Grin: Interactive graphics for modeling solids*, IBM Journal of Research and Development, 25(4), 281-294.
15. Fitzgerald, W. J., (1981). *Using axial dimensions to determine the proportions of line drawings in computer graphics*, Computer-Aided Design, 13(6), 377-382.
16. Freeman, H., (1991). *Computer name placement*. In D. J. Maguire, M. F. Goodchild, and D. W. Rhind, editors, Geographical Information Systems: Principles and Applications, pp. 445-456. Longman, London.
17. Goyal, V. K., Valentina, R., Cruza, M. J., Nieves, N. J., (2014). *Characterization of a 2D Geometry Using C++ Interface*, American Transactions on Engineering & Applied Sciences, 3(12013), 15-30.
18. Hashimoto, J., Fukushima, H., Kowaguchi, W., (1986). *High Performance CAD System with Full Automatic Dimensioning and Multi-Modeling Features Based on Engineering Workstation*, pp. 358-378. Springer Japan, Tokyo.
19. Imhof, E., (1975). Positioning names on maps. *The American Cartographer*, 2(2), 128-144.
20. Kakoulis, K. G., Siozos, A. G., (2017). *On the Automatic dimensioning of Technical Drawings*, Sixth International Conference on Advances in Mechanical & Robotics Engineering (Proc. AMRE 2017), pp. 30-34.
21. Kakoulis, K. G., Tollis, I. G., (2001). *On the Complexity of the Edge Label Placement Problem*, Computational Geometry, 18(1), 1-17.
22. Kakoulis, K. G., Tollis, I. G., (2003). *A Unified Approach to Automatic Label Placement*, International Journal of Computational Geometry and Applications, 13(1), 23-60.
23. Kakoulis, K. G., Tollis, I. G., (2013). *Labeling algorithms*. In R. Tamassia, editor, Handbook of Graph Drawing and Visualization, pp. 489-515, Chapman and Hall/CRC Press.
24. Kakoulis, K. G., Tollis, I. G., (2016). *Modifying orthogonal drawings for label placement*, Algorithms, 9(2), 22.
25. Li, C. L., Lee, Y.H., Yu, K. M., (2006). *Automatic datum dimensioning for plastic injection mould design and manufacturing*, The International Journal of Advanced Manufacturing Technology, 28(3), 370-378.
26. Marks, J., Shieber, S., (1991). *The computational complexity of cartographic label placement*, Technical Report 05-91, Harvard University.
27. Raffaeli, R., Cicconi, P., Germani, M., (2016). *Automation of drafting execution by schemes definitions and feature recognition*, Computer-Aided Design and Applications, 13(4), 459-470.
28. Ren, A., Yu, Q., Xu, Y., (2002). *Drawing recognition for automatic dimensioning of shear-walls*, Tsinghua Science and Technology, 7(6), 555-559.
29. Requicha, A.A.G., (1977). *PADLI: dimensioning and tolerancing*, Report TM 19, Production Automation Project, University of Rochester.
30. Tollis, I. G., Kakoulis, K. G., (2014). *Graph drawing*. In A. B. Tucker and T. Gonzalez, editors, Computing Handbook: Computer Science and Software Engineering, Third Edition, pp. 14:1-21. Chapman and Hall/CRC Press.
31. 27. Wagner, F., and Wolff, A., (1998). *A combinatorial framework for map labeling*. In S. Whitesides, editor, Graph Drawing (Proc. GD 1998), volume 1547 of Lecture Notes in Computer Science, pp. 316-331. Springer-Verlag.
32. Yoeli, P., (1972). *The logic of automated map lettering*. The Cartographic Journal, 9(2), 99-108.
33. Yu, K. M., Tan, S. T., Yuen, M. F., (1994). *A review of automatic dimensioning and tolerancing schemes*, Engineering with Computers, 10(2), 63-80.
34. Yuen, M.M.F., Tan, S.T., Yu, K.M., (1988). *Scheme for automatic dimensioning of csg defined parts*, Computer-Aided Design, 20(3), 151-159.

Received: May 30, 2023 / Accepted: December 15, 2023 / Paper available online: December 20, 2023 © International Journal of Modern Manufacturing Technologies