

# On the Automatic Dimensioning of Technical Drawings

K. G. Kakoulis and A. G. Siozos

**Abstract**— Traditional (2-D) technical drawings are still used by people involved in the production chain to manufacture parts to exact specifications. Of the many processes involved in the production of a good technical drawing, dimensioning is one of the most tedious, time-consuming and error prone. Most state of the art CAD systems generate 2-D technical drawings from 3-D models automatically. They also place dimensions automatically, however, in a rather primitive way. In practice, the automatically generated dimensions are the starting point for a manual improvement of the dimension positions. In this paper, we present fast and efficient techniques for the automatic placement of dimensions to 2-D technical drawing, which we have implemented and incorporated into a commercial CAD system. The dimensions produced follow the basic drawing standards and do not produce redundant dimensions.

**Keywords**—CAD, Automatic Dimensioning, Automatic label placement, technical drawings

## I. Introduction

The advances in the field of *Computer Aided Design* (CAD) systems have revolutionized the industrial design process at all levels. However, the traditional technical drawings are still used by people involved in the production chain to manufacture parts to exact specifications. A precise technical drawing having a graphic representation of the part with complete dimensioning assists in the prevention of errors and financial loss [7]. Of the many processes involved in the production of a good technical drawing, dimensioning is one of the most tedious, time-consuming and error prone. It is worth noting that even in the most advanced CAD systems the dimensioning of technical drawings is done basically manually. To minimize errors dimensioning standards are used throughout the industry. Still, the proper dimensioning of technical drawings depends largely on the designer's experience and method of work.

Most state of the art CAD systems generate 2-D technical drawings from 3-D models automatically. They also place dimensions automatically, however, in a rather primitive way. They place dimensions to each graphical feature of the drawing without taking always into account the basic drawing standards. For example, it is common to have overlaps of

dimension lines, redundancy or misplacement of dimension. In practice, the automatically generated dimensions are the starting point for an intensive and even more error prone manual improvement of the dimension positions.

If a CAD system can generate 2-D technical drawings with correct dimensions automatically, the time of product design can be reduced significantly. Given the fact that almost all advanced CAD systems can generate 2-D drawings out of 3-D models, we have focused our effort in dimensioning 2-D technical drawings automatically.

In this paper, we present techniques for the automatic placement of dimensions to graphical features of 2-D technical drawing. We will refer to this as the *Automatic Dimensioning Placement* (ADP) problem. Our techniques can be used with any type of 2-D drawing as input (e.g., architectural, mechanical or topographical drawings). Because they do not derive the actual dimensions from a 3-D part, they can be used to dimension drawings produced from converting scanned paper drawings to vector data. Furthermore, our approach is fast, efficient, does not produce redundant dimensions and warranties that all graphical features will be dimensioned. Our paper is organized as follows: In Section 2 we reviews the related work in automatic placement of dimensions. In Section 3 we define the ADP problem. In Section 4 we present the main ideas of our algorithms and a software prototype, incorporated into a commercial CAD system, with an implementation of our algorithms. We conclude in Section 5 with directions for future work.

## II. Related Work

A key task in information visualization is the annotation of the information through label placement. The problem of automatic label placement is important [8], and has applications in many areas including Information Visualization [21], Cartography [10], GIS [16], Graph Drawing [11, 26], and Engineering [9].

Most of the research addressing the labeling problem has been focused on labeling graphical features of geographical maps [4, 13, 20, 27] and graph drawings [12, 22]. The problem of placing labels on the boundary rectangle of the drawing where each graphical feature is connected to its associated label through a curve, called leader, has been studied in [3, 6, 23].

For the automatic placement of dimensions to technical drawings most of the research has been focused on generating dimensions for 2-D drawing of mechanical parts automatically from their 3-D models. Requicha [25] presented a technique for dimensioning drawings corresponding to 3-D models created with the Constructive Solid Geometry (CSG) solid

---

K. G. Kakoulis  
Western Macedonia University of Applied Sciences  
Greece

A. G. Siozos  
Western Macedonia University of Applied Sciences  
Greece

modeling technique. Its application is limited because in CSG the geometric objects must be defined procedurally and built from set-theoretic combinations of primitive 3-D models. More work on dimensioning objects created with CSG can be found in [14, 15, 30]. Hashimoto [17] presented an algorithm for automatic dimensioning of 2-D drawings. Only linear and radial dimensions were covered and no adequate checking of dimensioning was provided. Bond [7] introduced a rule-based system based on domain knowledge. Early attempts in automatic dimensioning have been summarized in [29]. Chen et al. [9] presented a technique that uses a rule-based expert system approach. With the placement of one dimension, a forbidden region is constructed so that all subsequent dimensions will not be placed in this region. The limitation of this method is due to its sequential nature.

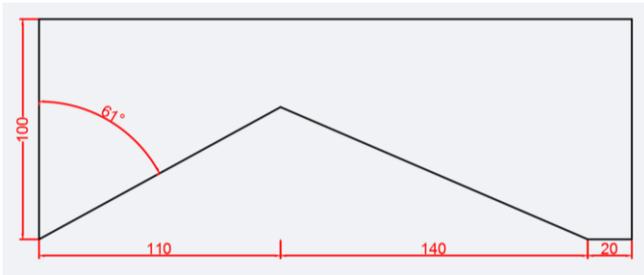


Figure 1. A drawing with dimensions produced by the ADP Algorithm.

### III. The ADP Problem

For the ADP problem the goal is to communicate the information for each graphical feature via dimensions in the best possible way. A critical task in the labeling process is to decide the best position for a dimension with respect to its corresponding graphical feature. It is difficult to quantify all the characteristics of a good label placement, because they reflect human visual perception, intuition, tradition and experience. Generally speaking, in a successful assignment, labels must be positioned such that they are legible and follow basic aesthetic quality criteria [18, 28]:

- A label must not overlap other labels or graphical features.
- A label must have legible size.
- A label must be easily identified with exactly one graphical feature.
- 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 view point. 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 (see Figure 2).

The purpose of dimensioning drawings is to allow the user to understand the spatial arrangement of the objects being drawn [7]. 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 in general be placed as close to their element as possible.
- Redundant dimensions must be avoided.
- Dimensions should not overlap.
- A dimension line should never be drawn through a dimension figure.
- Intersection between dimension lines and/or projection lines must be avoided.
- Similarly situated dimensions (e.g., parallel) should be grouped together and are usually spaced at a convenient distance apart with smaller ones on the inside.
- For interior elements (i.e., not on the periphery of the projection) it is often advisable to place the dimension on the periphery, and to use extension lines.
- Distribute the dimensions uniformly as far as possible.

The ADP problem can be viewed as 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:

- Minimize the total number of dimensions on each side.
- Minimize the total length of the main dimension lines.
- Minimize the total length of the extension dimension lines.

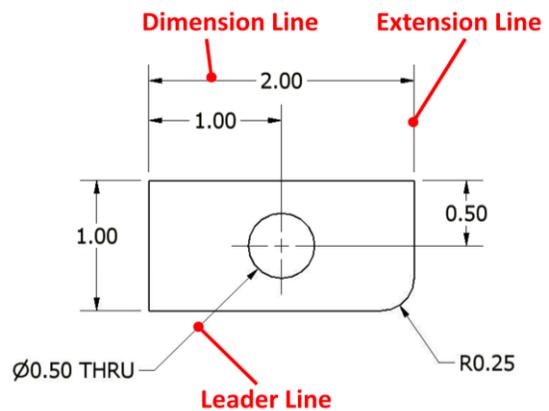


Figure 2. The structure of a dimension.

In order to minimize any of the above criteria, we have to decide on which sides of the bounding rectangle the dimensions must be placed. However, this task corresponds to the Partition problem, which is a well known NP-complete problem. Furthermore, the label placement problems are typically NP-hard [4,5,19,22,24]. Because automatic labeling

is a very difficult problem we rely on heuristics to provide practical solutions for real world problems.

In the next section we present a practical algorithm for placing dimensions automatically to 2-D technical drawings.

#### IV. A Polynomial Time Algorithm for the ADP Problem

In the following we present a polynomial time algorithm for the ADP problem. First, we make the following assumptions to simplify our description:

- The geometric entities to be dimensioned are line segments, circles and arcs.
- The drawing is bounded by a single closed polygon.

The basic idea of our 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 order to decide on which side of the bounding rectangle  $R$  of  $G$  the dimension of a line segment  $s$  will be placed, we employ the following technique: we compare 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  $G$ . Next, we define the center of gravity of  $G$ :

**Definition 1.** Let  $L = (e_1, e_2, \dots, e_n)$  be the set of all the line segments of the drawing  $G$ . Also, let  $k_i$  be the midpoint of the line segment  $e_i$ . We define as the center of gravity of  $G$  the point  $P$  with coordinates:

$$x_p = \frac{1}{n} \sum_{i=1}^n x_{k_i}, \quad y_p = \frac{1}{n} \sum_{i=1}^n y_{k_i}$$

By using the center of gravity of  $G$  to decide on which side of  $R$  we will project and thus place the dimensions of a line segment, we achieve two things: (i) A well balanced placement of dimensions around  $R$  and (ii) a placement of the dimensions close to their annotated features.

In a preprocessing step the bounding rectangle  $R$  and the center of gravity  $P$  of  $G$  are calculated. 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, we firstly process line segments that both of their end vertices touch edges of  $R$ . If a line segment  $s$  is vertical (horizontal) then it coincides with the corresponding edge of  $R$  and therefore a single dimension can be added. If  $s$  is oblique then it touches diagonally opposite sides of  $R$ . If the center of gravity  $P$  is above (right) of the midpoint of  $s$  then a horizontal projection is added on the bottom (left) side of  $R$ . Next, we process line segments that one of their end vertices touches an edge of  $R$ . The projections of the defining vertices of  $s$  are treated the same way as previously. Furthermore, if the line segment is not horizontal or vertical, it is preferably

dimensioned with polar dimensions (angle and relative distance from its neighboring line segment). Then, line segments that are completely inside  $R$  are processed in a similar fashion. Finally, centers of circles and arcs are added using ordinary dimensions on the opposite sides of  $R$ . The radii are added locally.

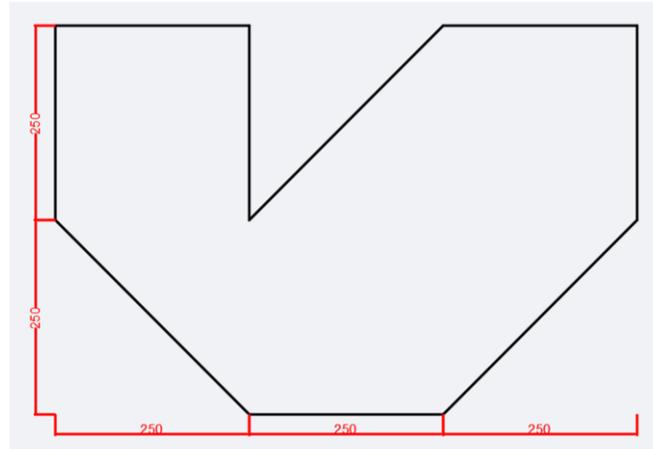


Figure 3. A drawing with dimensions produced by the ADP Algorithm.

Next, we present an algorithm for solving the ADP problem:

##### Algorithm ADP

**Input:** A technical drawing  $G$ .

**Output:** An assignment of dimensions for each graphical feature of  $G$ .

1. Calculate the bounding rectangle  $R$  of  $G$ .
2. Calculate the center of gravity  $P$  of  $G$ .
3. For each line segment  $s$  of  $G$ .
  - If  $s$  lies on an edge  $e$  of  $R$ .
    - project the end vertices of  $s$  on  $e$ .
  - else
    - Calculate the position of the midpoint of  $s$  with respect to  $P$ .
    - Project  $s$  to the appropriate sides of  $R$ .
4. Sort the projected line segments on each side of  $R$ .
5. Remove duplicate dimensions.
6. Add dimensions to circles and arcs.

From the description of Algorithm ADP it is clear that all line segments are projected onto the sides of the bounding rectangle of the input technical drawing. Thus, no line segment will be without dimensions assigned to it. Hence, we have the following lemma:

**Lemma 1.** Algorithm ADP assigns dimensions to each graphical feature of the input technical drawing.

In step 5 of Algorithm ADP we remove any duplicate dimensions. First, the end vertices of each line segment are projected onto the sides of the bounding rectangle  $R$  of the input technical drawing  $G$  only once. Thus, no graphical feature will be dimensioned twice. However, there are cases (see Figure 4) where implicit duplication of dimensions can be

occurred on opposite sides of  $R$ . In order to eliminate implicit duplication of dimensions we employ the following technique: We sweep concurrently the projection points stored in the lists corresponding to opposite vertical (horizontal) sides of  $R$ . If the same interval is found in both lists, then, this interval is removed from the list corresponding to the right (top) side of  $R$ , since the most preferable position to place a dimension, according to the drawing standards, is to the left and down with respect to the bounding rectangle of the technical drawing. Hence, the following is true:

**Lemma 2.** *Algorithm ADP does not produce duplicate dimensions.*

Finally, the most time consuming step of Algorithm ADP is step 4, where we sort the projected line segments on each side of  $R$ . The running time of step 4 is  $O(n \log n)$  running time, where  $n$  is the number of graphical features of the input technical drawing. Hence, we have the following:

**Theorem 1.** *Algorithm ADP assigns dimensions to each graphical feature, without duplications, of the input technical drawing  $G$  in  $O(n \log n)$  running time.*

### A. A System for the ADP problem

The techniques for solving the dimensioning problem presented in this paper have been implemented in VB and incorporated into a commercial CAD system (AutoCAD) through its Application Program Interface (API). A user can draw or insert a 2-D drawing into AutoCAD and position dimensions automatically to all line segments, circles and arcs by executing the implemented plugin. The dimensions produced follow the basic drawing standards. In addition, there are no redundant dimensions. The dimensions of the technical drawings in Figures 1, 3 and 5 have been placed by the implemented AutoCAD plugin.

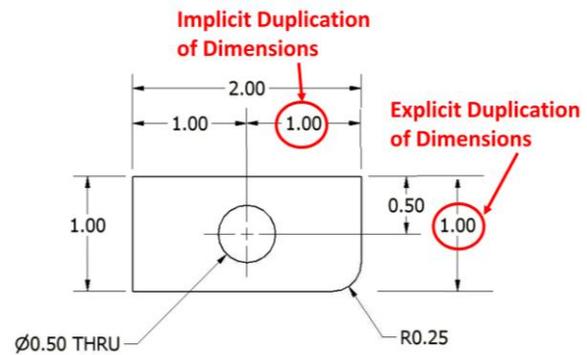


Figure 4. An example of explicit and implicit duplication of dimensions.

## v. Conclusions and Future Work

Two dimensional technical drawings still play an important role in the manufacturing process. Placing dimensions to those drawings is tedious, time-consuming and error prone, thus, its automation is very desirable. Current CAD systems produce an initial placement of dimensions, which has to be improved manually. In this paper we have presented algorithms for the automatic placement of dimensions to graphical features of 2-D technical drawing. A corresponding software prototype was developed and incorporated into a commercial CAD system (AutoCAD).

Future work includes extending the functionality of the current software prototype by positioning, for example, dimensions to sections, details and chamfers. Furthermore, detection of symmetries of the input drawing and adjustment of the dimension positions to reflect those symmetries could be an interesting problem.

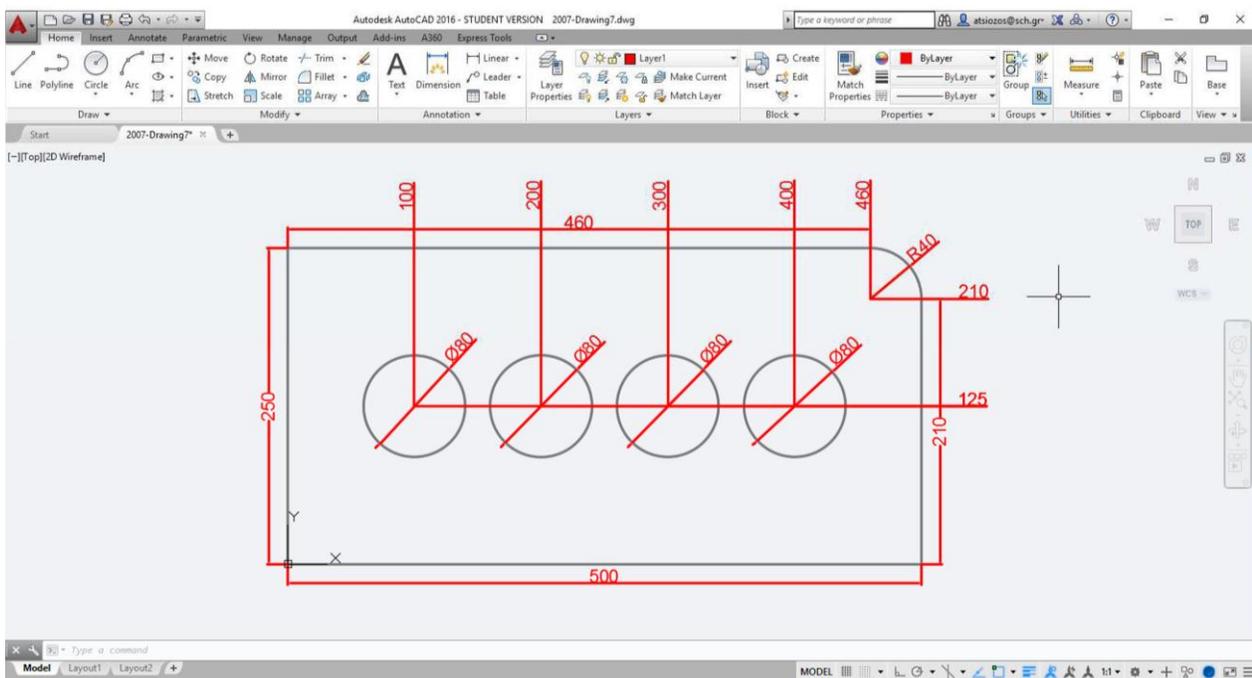


Figure 5. A drawing with dimensions produced by the ADP Algorithm. Ordinary dimensions have been used for the centers of circles and arcs.

## References

- [1] ASME y14.5. Dimensioning and Tolerancing: Engineering Drawing and Related Documentation Practices. 2009.
- [2] ISO 128-1. Technical drawings: General principles of presentation. 2003.
- [3] K. Ali, K. Hartmann, and T. Strothotte. Label Layout for Interactive 3D Illustrations. *Journal of the WSCG*, 13:1{8, 2005.
- [4] K. Been, E. Daiches, and C. Yap. Dynamic map labeling. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):773{780, 2006.
- [5] M. A. Bekos, M. Kaufmann, D. Papadopoulos, and A. Symvonis. Combining traditional map labeling with boundary labeling. In I. Cerna, T. Gyimothy, J. Hromkovic, K. Jefferey, R. Kralovic, M. Vukolic, and S. Wolf, editors, 37<sup>th</sup> International Conference on Current Trends in Theory and Practice of Computer Science (Proc. SOFSEM 2011), pages 111-122. Springer-Verlag, 2011.
- [6] M. A. Bekos, M. Kaufmann, A. Symvonis, and A. Wolff. Boundary labeling: Models and efficient algorithms for rectangular maps. *Computational Geometry*, 36(3):215-236, 2007.
- [7] A. Bond and S. Ahmed. Knowledge-based automatic dimensioning. *Artificial Intelligence in Engineering*, 4(1):32-40, 1989.
- [8] Bernard Chazelle et al. 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*, pages 407-463. American Mathematical Society, Providence, 1999.
- [9] K. Chen, X. Feng, and Q. Lu. Intelligent dimensioning for mechanical parts based on feature extraction. *Computer-Aided Design*, 33(13):949-965, 2001.
- [10] J. Christensen, J. Marks, and S. Shieber. An empirical study of algorithms for Point Feature Label Placement. *ACM Transactions on Graphics*, 14(3):203{-232, July 1995.
- [11] G. Di Battista, P. Eades, R. Tamassia, and I. G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs*. Prentice Hall, Upper Saddle River, NJ, 1999.
- [12] U. Dogrusoz, K. G. Kakoulis, B. Madden, and I. G. Tollis. On Labeling in Graph Visualization. *Special Issue on Graph Theory and Applications, Information Sciences Journal*, 177(12):2459-2472, 2007.
- [13] S. Edmondson, J. Christensen, J. Marks, and S. Shieber. A General Cartographic Labeling Algorithm. *Cartographica*, 33(4):321-342, 1997.
- [14] W. Fitzgerald, F. Gracer, and R. Wolfe. Grin: Interactive graphics for modeling solids. *IBM Journal of Research and Development*, 25(4):281-294, July 1981.
- [15] W. J. Fitzgerald. Using axial dimensions to determine the proportions of line drawings in computer graphics. *Computer-Aided Design*, 13(6):377-382, 1981.
- [16] H. Freeman. Computer name placement. In D. J. Maguire, M. F. Goodchild, and D. W. Rhind, editors, *Geographical Information Systems: Principles and Applications*, pages 445-456. Longman, London, 1991.
- [17] J. Hashimoto, H. Fukushima, and W. Kowaguchi. High Performance CAD System with Full Automatic Dimensioning and Multi-Modeling Features Based on Engineering Workstation, pages 358-378. Springer Japan, Tokyo, 1986.
- [18] E. Imhof. Positioning names on maps. *The American Cartographer*, 2(2):128-144, 1975.
- [19] K. G. Kakoulis and I. G. Tollis. On the Complexity of the Edge Label Placement Problem. *Computational Geometry*, 18(1):1-17, 2001.
- [20] K. G. Kakoulis and I. G. Tollis. A Unified Approach to Automatic Label Placement. *International Journal of Computational Geometry and Applications*, 13(1):23-60, 2003.
- [21] K. G. Kakoulis and I. G. Tollis. Labeling algorithms. In R. Tamassia, editor, *Handbook of Graph Drawing and Visualization*, pages 489-515. Chapman and Hall/CRC Press, 2013.
- [22] K. G. Kakoulis and I. G. Tollis. Modifying orthogonal drawings for label placement *Algorithms*, 9(2):22, 2016.
- [23] C.L. Li, Y.H. Lee, and K.M. Yu. Automatic datum dimensioning for plastic injection mould design and manufacturing. *The International Journal of Advanced Manufacturing Technology*, 28(3):370-378, 2006.
- [24] J. Marks and S. Shieber. The computational complexity of cartographic label placement. Technical Report 05-91, Harvard University, 1991.
- [25] A.A.G. Requicha. PADLI: dimensioning and tolerancing. Report TM 19, Production Automation Project, University of Rochester, 1977.
- [26] I. G. Tollis and K. G. Kakoulis. Graph drawing. In A. B. Tucker and T. Gonzalez, editors, *Computing Handbook: Computer Science and Software Engineering*, Third Edition, pages 14:1-21. Chapman and Hall/CRC Press, 2014.
- [27] F. Wagner and A. Wolff. A combinatorial framework for map labeling. In S. Whitesides, editor, *Graph Drawing (Proc. GD 1998)*, volume 1547 of *Lecture Notes in Computer Science*, pages 316-331. Springer-Verlag, 1998.
- [28] P. Yoeli. The logic of automated map lettering. *The Cartographic Journal*, 9(2):99-108, 1972.
- [29] K. M. Yu, S. T. Tan, and M. F. Yuen. A review of automatic dimensioning and tolerancing schemes. *Engineering with Computers*, 10(2):63-80, 1994.
- [30] M.M.F. Yuen, S.T. Tan, and K.M. Yu. Scheme for automatic dimensioning of csg defined parts. *Computer-Aided Design*, 20(3):151{159, 1988.

About Author (s):



Associate Prof. K. G. Kakoulis works for the Dept. of Mechanical and Industrial Design Engineering, Western Macedonia University of Applied Sciences, Greece. His main research interests include design and analysis of algorithms, information visualization and geometric modeling.



A. G. Siozos holds bachelor degrees in Computer Engineering from the Univ. of Patras, Greece, and in Industrial Design Engineering from the Western Macedonia Univ. of Applied Sciences, Greece. His main research interests include algorithm engineering and geometric modeling.