

Assessment Strategy to Engage Students in Constraining Parametric CAD Drawings*

PEDRO COMPANY, MIQUEL GOMEZ-FABRA, MARÍA JESÚS AGOST and MARGARITA VERGARA

Universitat Jaume I, Castelló de la Plana, Spain, Departament d'Enginyeria Mecànica i Construcció.

E-mail: {pcompany, gomezf, magost, vergara}@emc.uji.es

The appearance of CAD applications with parametric capabilities has not changed some obsolete academic habits based on drawing as if a ruler and compass were still the tools used for this purpose. In this work, it is argued that this is not reasonable in terms of training strategies or in terms of efficiency. To support this statement, a new method of evaluation that measures and compares the efficiency of different geometrical construction approaches is presented. The use of this metric as a teaching strategy is illustrated through an experimental study applied in the classroom.

Keywords: computer-aided design; geometrical constructions; geometric loci; geometric constraints

INTRODUCTION

THE INTRODUCTION OF CAD was over 40 years ago and the advent of parametric CAD more than 15 years ago. However, in many schools of engineering there has not been a change in the loci paradigm that was formerly used to perform geometrical reconstructions. It is true that the introduction of non-parametric CAD tools has simplified constructions, but in many cases the paradigm has been maintained. The simplification has consisted in accepting the new constructions based on the 'virtual' instruments supplied by the CAD application as being exact and in adding new 'primitives' (programmed figures that the computer constructs automatically) to the body of geometrical constructions that are valid for conceiving geometric loci. In fact, the process has been the continuation of the one that was started by accepting the validity of the constructions obtained with instruments other than the unmarked ruler and the compass (set-squares and protractors, ellipse templates, etc.).

Yet, it is a well-known fact that parametric CAD applications make it possible to evolve towards a method that complements and expands the loci methodology by direct management of the geometrical constraints. Thus, certain authors defend the massive-scale incorporation of parametric CAD in the syllabi of Graphics Engineering studies (i.e. [1]). The proposal is not new. In Sutherland's dissertation [2] a regular hexagon was obtained from an irregular hexagon: its vertices had to comply with the condition of belonging to a circumference and then the six

sides of the hexagon were made equal. In this pioneering work CAD was already accompanied by constraints. Sutherland's work proposed a total of 17 different possibilities: parallelism, perpendicularity, incidence, etc.

A new method for comparing the efficiency of different ways of constructing geometrical figures is presented. It is a 'steps' method that enables the users to quantify the cost involved in performing different construction methods. This metric system will help us to prove that efficiency of the parametric methods is at least comparable to that of earlier techniques. Though, a more important objective is addressed: that of acquiring a good profile generation strategy. In our experience, training courses on 3D parametric CAD tend to omit the profile generation task, under the assumption that is is a skill commonly possessed by all students. Hence, many courses focus on generating 3D models and fail to emphasize constraining parametric CAD drawings. During our teaching experience, we have found that students lacking valid strategies to generate 'good' profiles tend to experience greater troubles when it comes to constructing and editing their 3D geometrical models than students who have developed the facility. Furthermore, we find it difficult to get students to understand the differences between good strategies for generating parametric profiles and their current paradigms orientated to either classical instruments or non-parametric CAD environments. We have used a study carried out in the classroom to prove that our 'steps' approach constitutes a practical teaching methodology that encourages students to discover parametric constructions and helps them to learn and elaborate efficient strategies.

* Accepted 28 June 2007.

BACKGROUND

The so-called ‘classical’ methods used to construct geometrical figures can be traced back to such ancient and exceptional milestones in the scientific literature such as Euclid’s *Elements* [3]. It is easy to see that such methods have survived down to the modern day and are widely included in practically all secondary school textbooks in Spain today. This is usually justified by highlighting the instructional value of the classical methods, as well as the fact that it is yet difficult to equip secondary schools with enough computers. Although a more thorough survey is needed, current evidence suggests that the situation may well be similar in other countries. The existence of recent patents for instruments designed specifically for drawing regular polygons is anecdotal but significant [4].

This same situation should not exist in university education.

First, universities are, or should be, equipped with both computers and parametric CAD applications. For instance, Sorby and Hamlin [5] indicate that with the increasing popularity of constraint-based modelling, the question is shifted from ‘Why?’ to ‘Which package?’. Clark and Scales [6] published the results of a study on the opinions of engineering design graphics teachers in the United States and one of their conclusions was that there is a clear tendency towards teaching constraint-based modelling. A more recent study [7] shows the different paradoxes of the current situation in the USA, where a high percentage of teaching using traditional instruments (55 per cent) coexists alongside the very widespread opinion that an important aim for the near future is to research into the ‘best ways to teach constraint-based modelling’.

Second, the classical methods are clearly no longer the most appropriate for constructing plane geometrical figures. But the need for construction methods has not disappeared. Indeed the classical methods are far less efficient and are not more exact than other methods available in CAD applications. It is also true that the simpler or more common figures can be obtained with ‘pre-programmed’ methods and all the user has to do is to call up the primitive from a menu in order to accomplish the desired result. Thus, at first sight it could be concluded that there is no longer any need for the classical or any other method. Nevertheless, it is easy to see that an educational strategy is required to prepare future engineers/designers to analyse complex geometrical constructions and to produce methods to address them. This is because sometimes there is no automatic command or direct procedure for constructing the desired figure from the data the user has available. Furthermore, 2D CAD (‘drafting’) applications are now losing ground to 3D CAD (‘modelling’) applications. As described by Hoffman [8], the sketch interface of modern 3D CAD systems allows the user to trace a rough

sketch, usually composed of lines and circular arcs, and annotate the sketch with dimensions and geometrical constraints. Sketches with constraints can be instantiated automatically, by solving geometrical constraints, as a profile that can be used to generate 3D shapes by operations such as extrude, revolve, protrude and cut. These ‘profiles’ are no more than well-constrained and parameterized geometrical figures.

Consequently, to be able to model it is necessary to construct well-constrained and parameterized 2D geometrical figures. The work by Wiebe, Branoff and Hartman [9] is a recent example of this type of approach. In fact, we consider it to be the one that comes closest to our proposal. The overall objective is similar, since in both cases we are interested in the ‘development of students’ understanding of Euclidean and topological properties of planar and solid geometry using constraint-based 3D modelling tools, not skills training for specific software tools’. Some other works describe similar approaches to this problem, like Martinez and Felez [10], whose approach aimed at learning dimensioning ‘establishes the complete geometry and constraints of a sketch and relates it with the complete dimensioning of the sketch’.

Within this context, the main contributions of our work are to establish the foundations of a method for quantitatively comparing the efficiency of geometrical constructions, and to see an example of how it can be used in teaching. The theoretical root of this approach is linked to one of the basic steps in the ‘design-based’ approach to engineering education, namely ‘detailing the metrics against which the achievement of the objectives can be measured and assessed’ [11]. Besides, as said in [12], there is a growing understanding that educational games may create a new and improved learning culture. Hence, we will see that our method can turn into a game, as today’s students (‘native speakers’ of the digital culture in general, and digital games in particular) can contend to construct geometrical figures with the fewest possible number of steps (minimizing the ‘score’). The game encourages learning and helps pupils to discover the constraint-based approach earlier (by generating new ‘evolutions’ of the figures).

CONSTRUCTION METHODS

Taking constructions of regular polygons as an example, we are going to show how changing over from the classical instruments to CAD applications clearly increases efficiency. And we will also see that moving from non-parametric to parametric CAD entails a small decrease in efficiency but offers the advantage of making the introduction of any subsequent changes simpler and more flexible.

Obviously, for the example of regular polygons to be representative of the more general problems

involved in geometrical constructions we have ignored the primitives that allow such figures to be drawn directly because they would distort the comparisons of the efficiency of the generic tools. We are therefore going to compare three scenarios: construction methods based on accepting the unmarked ruler (straightedge) and compass as the only valid instruments (classical methods); constructions that make use of the tools typically included in a non-parametric 3D CAD application (except the specific primitives for drawing regular polygons); and, lastly, constructions suitable for a parametric CAD environment (again, without considering the regular polygon primitives).

As is well known, (see, for instance, the 'regular polytope' entry at the Internet site wikipedia.org), the classical approach, using unmarked ruler and compass, to constructing an exact regular polygon is very simple for some polygons. Examples of this would be the triangle, square, hexagon or octagon inscribed within a circumference. Note the dependence on the instruments, because to obtain inscribed octagons all we have to do is to divide the circumscribed circumference into eight parts by separating diameters by 45°; but we need a set-square and a protractor to do so. With just a ruler and a compass more painstaking methods are required. A pentagon can also be constructed inside a circumference by an exact, but particular and more complex, method. Lastly, for polygons that cannot be constructed with a ruler and compass without any error (such as polygons with n sides where n equals 7, 9, 11, etc.), we have to use a generic, painstaking and approximate method for constructing a regular n -sided polygon within a given circumference.

If a user had to draw a regular polygon with a non-parametric CAD application without using the specific primitive of the n -sided regular polygon, he or she could use methods that take advantage of the fundamental metric properties whereby all the sides of the polygon are of equal length and the angles between adjacent sides are equal to a value that is inversely proportional to the number of sides in the polygon. Therefore, the method for drawing regular polygons without using the corresponding primitive can be as simple as setting the divisor of the 'entity references' (also known as 'snaps', relationship controls, etc.) for the number of sides the desired polygon has, drawing the circumscribed circumference and then drawing the segments using its divisions as vertices. Thus, the loci method does not disappear, although it is greatly simplified because obtaining the points that divide a figure (segment, circumference, etc.) becomes an operation that is performed automatically.

With a parametric CAD application we cannot be sure whether we are going to be able to reduce the workload on the user. But what is achieved is that the construction produced can be easily modified in order to adapt it to new values of the basic parameters. For instance, after constructing a

regular polygon within a circumference with a radius r , we can change the radius and obtain a new polygon without having to repeat the construction. It is just as easy to make the length of the sides of the polygon the independent parameter. As a result, defining the polygon inscribed within a circumference and adding the condition of equal sides is efficient. Note that the construction will be more laborious if we do without the circumference because we will have to make both the angles and the sides equal. Thus, familiarity with geometric properties (such as the fact that a regular polygon is always inscribed within a circumference) helps us to choose more efficient methods. And it should also be noted above all, that the construction method is no longer based on loci but instead on restraints of a topological (vertices belong to a circumference with a variable radius) and geometrical nature (sides must be equal to each other).

OUR STUDY

In the previous section we performed a qualitative comparison of the three methods of constructing geometric figures. But in order to be able to compare implementation costs quantitatively, we propose a method based on counting the 'steps' needed to construct the same figure. For the processes to be comparable, first we suggest simulating the drawing by means of classical instruments in a CAD application, using only tools that simulate a ruler, compass, set-square, protractor and T-square. Each click on the mouse to choose commands on the menus, select drawing elements or mark new points is then defined as a step. Using the keyboard to introduce a command, coordinate or whatever is also considered to be a step. In sum, a step is taken as being the introduction of any command or parameter needed to perform a construction.

It is true that the settings of CAD applications can have an important effect on the overall cost. The most significant aspect is the settings of the automatic, or semi-automatic, aids for establishing relations between geometrical entities, such as connecting the start end of a new segment to the finish end of the previous segment, drawing a concentric circumference around another previously drawn one, and so forth. These aids have a number of different denominations, such as entity references, snaps, relationship controls, etc. Moreover, not all of them are equally powerful and efficient. However, this does not affect our study because we are only going to use the most common options, such as the automatic detection of all the 'points of interest' (end points, intersections, etc.), which often avoid the need for auxiliary constructions to 'mark' points that are going to be used in later operations because these points can be found automatically by the reference

detector built into the application. In short, we will count the steps assuming that the applications are always set up in the most efficient way. Furthermore, we will also assume that the user utilises strategies to reduce the number of steps. For example, drawing a segment requires three steps: selecting the 'draw segment' command (one step), marking the initial vertex (one step) and marking the end vertex (one step). Drawing n segments one after the other and with connected vertices, however, requires $3+(n-1)+1$ steps. This is because the first segment requires 3 steps, but those that follow only require one step (to mark the finish end). A final step is added that is needed to interrupt the 'chain' of segments (which corresponds to pressing the 'esc' key or performing some similar action). Nevertheless, this is an aspect that exemplifies the differences between applications because some programs, such as SolidEdge, automatically interrupt the chain when a polygon is generated. Hence, $3+n$ steps would be needed for an open chain with n sides and $2+n$ for an n -sided polygon. An extra step would have to be added for each side if their lengths are specified.

The cost, in terms of steps, of the classical method of constructing a hexagon inscribed within a circumference is 15 steps and can be seen in detail in the table in the Teaching experience section.

Supposing that setting the divisor of the 'entity references' for the number of sides in the desired polygon requires steps (since it is an atypical setting), we counted three steps to change the settings and three steps to return to the defect setting (once the polygon has been drawn). We must also add three more steps to draw the circumference and the $(3+n)$ steps required to draw the n sides using its divisions as vertices. The cost for a polygon with n sides would therefore be: $3+3+3+(3+n) = 12+n$. Hence, the cost of drawing a hexagon inscribed within a circumference with a non-parametric CAD application would be 18 steps.

Modifying the polygons so that they have known sides, by means of the dilatation operation, has an additional cost of 12 steps to determine the similarity relation, three steps to select the original figure and three steps to apply the similarity relation. In other words, a total of $(12+n) + 18 = 36$ steps are required for the hexagon.

Lastly, if we construct the hexagon with the parametric method proposed above, it is seen that 26 steps are needed (Fig. 1):

1. Drawing the circumscribed circumference (Three steps: selecting the circumference, marking the centre and marking a random point on the circumference or indicating its radius).
2. Drawing the sides by joining randomly chosen points on the circumference. Drawing each side requires three steps: selecting the 'segment' command, marking the start point and marking

the end point. If the six sides are drawn in a single operation, they only require $3+n = 9$ steps.

3. Establishing the equality relation between a side chosen at random and the other $n-1$ sides. The equalling operation is selected, and the first and each of the other sides are marked. Altogether, $(1+2*(n-1)) = 11$ steps.
4. Dimension the diameter or the side, according to the preferred control parameter (three steps).

The operation is therefore less efficient than classical construction and construction with non-parametric CAD (26 versus 15 and 18 steps, respectively). But, as is the case with non-parametric CAD, this operation is generic (i.e. valid for any regular polygon). The generalization of the steps described in Fig. 1 for an n -sided polygon would therefore be: $3+(3+n-1)+(1+2*(n-1))+3 = 7+3*n$. Hence, it is easy to see that this construction is more efficient than those of regular polygons that have no simple specific method.

Moreover, we can now start to appreciate the great advantage of parametric methods as we realise that we can modify both the diameter of the circumference and the lengths of the sides (which require only two steps: selecting the dimension and changing its value). No costly dilatations or special methods are needed to obtain a polygon with a certain number of sides. In the case of a hexagon we can also control the 'engineering' parameter of the distance between faces, for example.

When all is said and done, the step method allows us to quantify the cost of the three methods for constructing geometrical figures and to see that constructions generated by means of parametric CAD applications have a slightly higher cost than the classical methods and constructions carried out using non-parametric CAD tools. Nevertheless, they offer advantages when the figure has to be edited in order to modify any of the parameters that define it.

APPLICATION TO TEACHING

The most convincing argument in favour of using parametric CAD tools and forgetting the old paradigms is a hands-on study carried out by the students themselves.

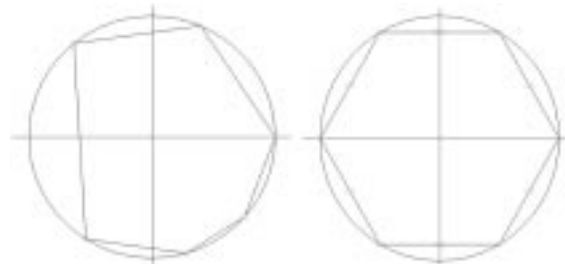


Fig. 1. Steps involved in constructing a hexagon in a parametric CAD application

First, the study asked a group of students to draw a simple geometrical shape in a parametric CAD application. The group was made up of 22 first-year engineering students. These students had been taught to draw in a 2D CAD environment (MicroStation v.8) under the geometric loci paradigm during the first term and the test was passed at the beginning of the second term. They had also learnt about basic 2D metric geometry in their secondary studies. However, before the experiment none of them had been previously exposed to 3D CAD or 2D parametric profiles.

The experiment lasted for one two-hour session in the CAD laboratory. During the first 45 minutes, the students were taught about the basics of the SolidEdge v.17 interface. Specifically, they were taught to run the software, to identify the different menus and to access the 2D profiles sketching environment. They also received instruction on the different tools for drawing primitives (segments, arcs, rectangles, etc.). Additionally, they learnt about the tools for creating constraints while drawing ('intellisketch') and the tools for creating, editing or deleting constraints after the geometrical figure has been drawn. Only the tools were introduced; strategies concerning the most efficient use of those tools were not explained. During this time, a video projector connected to the instructor's computer allowed them to see the instructor drawing some shapes while introducing the different menus and tools, and they were allowed to run the application freely, as they liked, on their own computers. No task was assigned to the students other than listening to the instructor.

After this generic instruction period, they were asked to draw a geometrical shape on their own. The task was explicitly introduced by means of the following text:

1. You are going to draw a geometrical figure with SolidEdge, and you must find the shortest and most efficient method to do it.

2. Geometrical constructions can be broken down into operations but we cannot just add up the number of operations in order to know which method is the most efficient because not all the operations are the same length.
3. In order to estimate and compare the 'cost' of the drawings you must count the 'steps' needed to draw the figure you have been asked to produce. Each click on the mouse to choose commands on the menus, select drawing elements or mark new points is counted as a step. Using the keyboard to introduce a command, coordinate, dimension or whatever is also considered to be a step. In short, a step is taken as being the introduction of any command or parameter needed to perform a construction.
4. An example of the steps carried out to construct a regular hexagon is described in Table 1.
5. You are to perform the most efficient construction (i.e. with the fewest steps) to draw the geometrical figure given on the other sheet of paper. The characteristics of the figure you are to construct are explained there and you should follow them closely in order to obtain exactly the figure you have been asked to draw. The sheet also explains a method to obtain the figure. It is not necessarily the most efficient method. If you find a better one, change it!
6. You can make as many attempts as you want before you finally settle for the method you consider to be the most efficient. Once you have chosen a method you will have to describe all the operations and steps in a table like Table 1.

The students were asked to draw one of four different shapes that were randomly distributed around the class. These were named exercises 1 to 4, and were explained in detail in the texts included in the annex.

The students were given a maximum of one hour to finish the task. During this time, they could ask the instructor questions to try and clear up any doubts they had, but they were not allowed to ask

Table 1. Steps to construct a regular hexagon

Operation	Steps
Draw the circumscribed circumference with radius r .	3 steps: select circumference, write the radius or diameter and mark the centre.
Mark a start point.	Zero steps if a 'key' point is chosen on the circumference (for example, one of the vertices of its four quadrants): it should not be marked because it can be detected using 'entity references' or 'snaps'.
Mark a new point by drawing a circumference with its centre at the last point that was marked and with a radius r .	Since the 'circumference' command is already selected from the previous step, and the radius or diameter have also been indicated, only the centre remains to be marked (1 step).
Repeat the previous operation three more times.	The cost of repeating the previous operation 3 times is 3 steps. An alternative method is to copy the previous circumference three times (5 steps).
Draw the sides by joining the points that have been marked.	Drawing each side requires 3 steps: selecting the 'segment' command, marking the start point and marking the end point. But If the six sides are drawn in a single operation, the 'segment' command is only selected once. And each end point is used as the start point for the next side. In all, 8 steps are required.

for suggestions about the best strategies to solve the exercise. When asked about what was meant by 'good' shapes, the instructor described a good profile as one that can be changed without 'crashing'. This term was immediately understood by students because, at that very same time, they were suffering repeated crashes as they tried to edit the profiles that underwent unexpected changes.

The same students were asked to draw a new figure (Fig. 5 in the Annex below) after another 6 to 12 hours' exposure to SolidEdge (drawing profiles, parts, assemblies and plans). Another group of 19 students were also asked to solve the second test profile. The second group was made up of students who began to create 3D models after a very short explanation about constraint-based geometry and after a first term where they had been taught to draw in a 2D CAD environment (MicroStation v.8) under the geometric loci paradigm, which was still the basis of their behaviour. In other words, they had been exposed to the same teaching experiences, the only exception being the pilot study described above. The post-test was introduced by the following text:

1. You are going to draw a geometrical figure with SolidEdge, and you must look for the shortest and most efficient method to do it.
2. You are to perform the most efficient construction (i.e. with the fewest steps) to draw the geometric figure given below. The characteristics of the figure you are to construct are explained and you should follow them closely in order to obtain exactly the figure you have been asked to draw. You should be able to modify the figure by changing any of its data.
3. You are also given a method to obtain the figure. It is not necessarily the most efficient method. If you find a better one, change it!

4. You can make as many attempts as you want before you finally decide on the method you consider to be the most efficient.

ANALYSIS

By the end of the first study, the instructor felt that the strategy had successfully fulfilled the purpose of drawing students' attention to profiles and of preventing them from starting to create models too soon on top of bad profiles. The 'game' of finding the best strategy, in terms of quantifiable steps, forced them to try out different strategies and sometimes led them to find the constraint-based one. Hence, guided self-learning was successful with some students. In the case of others, the experiment made them receptive to the instructor's explanations given during the first fifteen minutes of the next session. As a result, all the students appeared to be able to begin creating 3D models from true parametric profiles after another 45 minutes of explanations by the instructor about creating 3D shapes from 2D constrained profiles. During the second hour of the second session (just three hours from zero!) all of them began to construct a 3D model with a medium degree of complexity.

All the students submitted a file containing the profile they had drawn and a sheet of paper with the description of the steps they had followed. The results are tabulated in Table 2. Students are ordered by the exercise they were randomly assigned (column 1). The instructor's assessment of the profiles appears in column 2 of Table 2. The scale is: 0) task not done at all; 1) drawing partially matches the intended figure) 2) drawing fully matches the intended figure, but does not support

Table 2. Results of the first test

Student	Profile Assessment	Steps by student	Steps by instructor	Constraint-based approach
E1-1	1	28	30	Partial
E1-2	4	24	20	Full
E1-3	3	18	18	Partial
E1-4	3	16	18	Partial
E1-5	2	18	18	None
E2-1	1	(0)	35	Partial
E2-2	2	(18)	35	Partial
E2-3	2	(27)	35	Partial
E2-4	2	25	35	None
E2-5	2	33	35	None
E2-6	3	(13)	35	None
E3-1	2	15	15	None
E3-2	4	22	25	Full
E3-3	2	16	15	None
E3-4	2	16	16	None
E3-5	3	22	22	Partial
E4-1	3	(19)	27	Partial
E4-2	3	24	23	Partial
E4-3	3	(17)	30	Partial
E4-4	3	32	32	Partial
E4-5	2	16	18	None
E4-6	3	29	27	Partial

edition changes; 3) drawing fully matches the intended figure, and partially supports edition changes and 4) drawing fully matches the intended figure and fully supports edition changes. The number of steps, as counted by students, is tabulated in the third column, while the number of steps, as counted by the instructor, is given in the fourth column. This includes the real number of steps required by the approach followed by each student. The difference with the previous column is therefore the number of steps not counted or wrongly counted by each student. Students who discovered the constraint-based construction approach on their own are indicated in the last column as “full”, while “partial” means that the student used some constraints, but not as many as he or she should have. The last option, “none”, means that the student used only the constraints provided automatically by the application, through its ‘intellisketch’ tool.

Although some students did not count the steps or missed some operations (their estimated number of steps appear in brackets), it can be seen that there is a certain amount of dispersion in the number of steps counted by students who followed similar strategies, as well as a mismatch between their calculations and the instructor’s calculations. It is therefore clear that the short explanation included in the test was not enough for the students to adopt the same counting criteria. A better explanation must be found in order to obtain valid ‘scorings’ for the ‘game’ in the future. Perhaps providing students with a table of the steps required for the most common tasks (like drawing a segment or an arc, or copying or scaling a figure) would be the most convenient way to homogenise the calculations.

Autonomous discovery is a well-reputed strategy to acquire and fix long-term working strategies. Its only drawback is that it is too time consuming in most cases. This point has been confirmed in this study, since only two students autonomously acquired the desired knowledge in the very short time that the experiment lasted. And eight students did not discover them at all. However, the experiment proved to be useful, since the post-test showed that most of them discovered and assumed the constraint-based strategy during the ensuing laboratory sessions, unlike what happened to other students who were not exposed to a similar experience. Results from the post-test (tabulated in Table 3) showed that 45 per cent of the former group built their profiles under a fully constrained paradigm, while only 10.5 per cent of the latter group adopted the fully constraint-based paradigm. Moreover, 21 per cent of the latter group had not yet adopted any constraint-based strategy.

In fact, the difference between both groups is even higher. Apparently, just 10 out of 22 members of the first group had gained a full understanding of the constraint-based approach, but actually the evaluation “4, partial” describes some students who did manage to create fully constrained drawings, although they still added some geometric loci. They thus gained full parametric control over the figure, but indirectly through some loci that were added unnecessarily. In fact, only 6 out of 22 students in the first group obtained less than the maximum profile assessment. Worth noting is the surprising lowest assessment for one of the only two students who obtained the best assessment in the first test (E3-2).

Table 3. Results of the second test

Student	Profile Assessment	Constraint-based approach	Other students	Profile Assessment	Constraint-based approach
E1-1	4	Full	O-01	3	Partial
E1-2	4	Full	O-02	1	Partial
E1-3	3	Partial	O-03	2	Partial
E1-4	3	Partial	O-04	2	Partial
E1-5	3	Partial	O-05	1	Partial
E2-1	4	Full	O-06	2	None
E2-2	4	Full	O-07	1	Partial
E2-3	4	Full	O-08	4	Full
E2-4	4	Full	O-09	1	Partial
E2-5	4	Partial	O-10	3	Partial
E2-6	4	Full	O-11	2	Partial
E3-1	4	Partial	O-12	4	Full
E3-2	1	Partial	O-13	2	Partial
E3-3	4	Full	O-14	1	Partial
E3-4	4	Full	O-15	2	Partial
E3-5	4	Partial	O-16	2	None
E4-1	4	Partial	O-17	1	None
E4-2	4	Partial	O-18	3	Partial
E4-3	4	Full	O-19	1	None
E4-4	3	Partial			
E4-5	3	Partial			
E4-6	4	Partial			

CONCLUSIONS

The step counting method is an incipient technique that needs to be refined and validated experimentally. We need to check empirically whether there is a desirable linear relation between the number of steps and performance costs in terms of time; if this were the case, it would enable us to estimate performance costs from the number of steps. We must also learn to count all the steps, as we have found that some tasks are more complex than others and it can therefore be deduced that they need to be broken down into more steps. In other words, the complexity of the steps has to be homogenised to prevent tasks requiring different amounts of time from being catalogued as equivalent. In any case, it is promis-

ing method for analysing the efficiency of different geometrical constructions.

Throughout the trial phase we observed that the method can also become a game for students in which they can compete to look for methods for constructing geometrical figures with the fewest possible number of steps. The game encourages learning and helps pupils to discover the constraint-based approach earlier. We believe that if students are asked to edit already-constructed shapes, that is, to generate new 'evolutions' of the figures, the process could become faster. In conclusion, the method is valid for getting future designers to think not only in terms of geometrical shapes and loci, but also helps them to discover and appreciate the advantages of the parametric nature of shapes more quickly.

REFERENCES

1. N. W. Hartman, Towards the Definition and Development of Expertise in the Use of Constraint-based CAD Tools: Examining Practicing Professionals. Ph.D. Thesis. Raleigh, North Carolina. (2003).
2. I. Sutherland. Sketchpad, a man-machine graphical communication system. Ph.D. Thesis. Massachusetts Institute of Technology. (1963).
3. Euclid. The Thirteen books of Euclid's Elements translated from the text of Heiberg, 2nd ed. New York Dover. (1956).
4. K. Guite. A geometrical instrument for constructing regular polygons. Patent Number IN9900929-I2. (2006).
5. S. A. Sorby and B. H. Hamlin. A Comparison of Two Commercial Constraint-Based Modelers for Use in an Engineering Graphics Course. *Eng. Design Graphics J.* **65**(1), 2001, pp. 30–38.
6. A. C. Clark and A. Scales. A Study of Current Trends and Issues Related to Technical / Engineering Design Graphics. *Eng. Design Graphics J.* **64**(1), 2000, pp. 24–34.
7. A. Scales, J. H. Petlick and A. C. Clark. A Survey of Current Trends and Issues for Engineering/ Technical Graphics Education: Results from a Five-Year Follow-up. Proceedings of 2005 ASEE Southeast Section Conference. (2005). <http://cee.citadel.edu/asee/%2Dse/proceedings/ASEE2005/ASEE2005SE.htm>
8. C. Hoffmann. Constraint-Based CAD. *J. Computing and Information Science in Eng.* **5**(3), 2005, pp. 182–187. <http://www.cs.purdue.edu/homes/cmh/distribution/PapersChron/ConstrBasedCAD.pdf>.
9. E. N. Wiebe, T. J. Branoff and N. W. Hartman. Teaching geometry through dynamic modeling in introductory engineering graphics. *Eng. Design Graphics J.* **67**(2), 2–3, pp. 12–20.
10. M. L. Martinez and J. Felez, J. An oriented constraint solving-based methodology approach to learn dimensioning. *Int. J. Eng. Educ.* **22**(2), 2006, pp. 359–372.
11. C. L. Dym. Design, Systems and Engineering Education. *Int. J. Eng. Educ.* **20**(3), 2004, pp. 305–312.
12. B. Foss and T. Eikaas. Game play in Engineering Education—Concept and Experimental Results. *Int. J. Eng. Educ.* **22**(5), 2006, pp. 1043–1052.

Pedro Company earned his Ph.D. in Mechanical Engineering in 1989 and has been Full Professor of Engineering Graphics at Universitat Jaume I since 1996. His research fields are Technical Drawing, CAD, Computer Graphics and Sketch-Based Modelling, with more than 20 international papers and communications on those areas published to date. He is currently participating in the development of a new sketch-based modelling interface and its applicability for teaching. He is also involved in Emotional Design and Collaborative Product Engineering projects.

Miquel Gómez-Fabra holds a Ph.D. in Industrial Engineering since 2004. He is an Assistant Professor of Engineering Graphics at the Universitat Jaume I. His research focuses on Engineering Geometry, especially Flexible Geometry, CAD and Learning.

María Jesús Agost is an Assistant Professor of Computer Aided Design Tools and Ergonomics at the Universitat Jaume I. She has participated in research projects about

product development in collaborative environments within the ceramics sector. Her current fields of research interest are Customer Integration in Product Design, Emotional Design and Computer Aided Design.

Margarita Vergara has held a Ph.D. in Industrial Engineering since 1998 and is an Associate Professor of Engineering Graphics and Ergonomics at Universitat Jaume I. Her research fields are Ergonomics, Emotional Design and Technical Drawing. She is the author of about 30 international papers and communications.

ANNEX

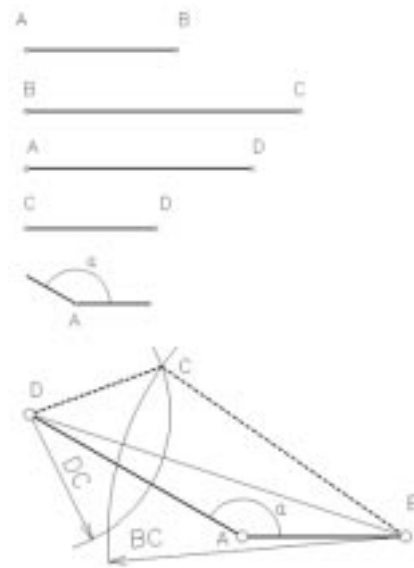
Exercise 1

A quadrilateral is a closed figure made up of four segments that are connected at their vertices. It is a particular type of irregular polygon. The segments that join alternate vertices are known as diagonals.

You are to construct a quadrilateral, the four sides and one angle being known (AB, BC, AD, CD and a in the figure).

You can start the construction by drawing the known angle (AB, AD and a). We then get a diagonal (BD) that enables us to construct the rest of the triangle from its three sides (BD, BC, CD).

To construct a triangle when its sides are known (AB, AC and BC), we draw any one of the three sides, for example AB; we then draw a circumference with a radius AC with A as its centre, and another circumference with a radius BC taking B as its centre. The two points where the two circumferences intersect are the possible solutions. We choose the point that fits the required triangle (the one that allows us to move round the vertices from A to B to C to D in an anticlockwise direction).



Data: $AB = 60$ mm; $BC = 95$ mm; $AD = 82$ mm;
 $CD = 46$ mm and $a = 150^\circ$.

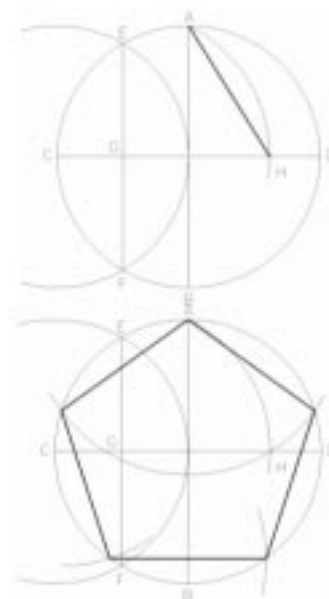
Exercise 2

A regular pentagon is a polygon with five vertices and five sides that fulfils the condition that the five sides are equal and the five angles are also equal.

A pentagon inscribed within a circumference has all its vertices contained inside it.

Construct a pentagon inscribed in a circumference with a radius of 125 mm.

One construction procedure consists in drawing a circumscribed circumference. Draw two perpendicular diameters (AB and CD). Make a copy of the circumscribed circumference, with its centre at C. Trace the chord that is common to both circumferences (EF). Trace the arc with its centre at G (midpoint of EF) and which passes through A until it bisects CD at H. Use the distance AH to mark the five vertices. Draw the circumference AH, select the circumference, and copy the circumference three times. Draw the five sides.



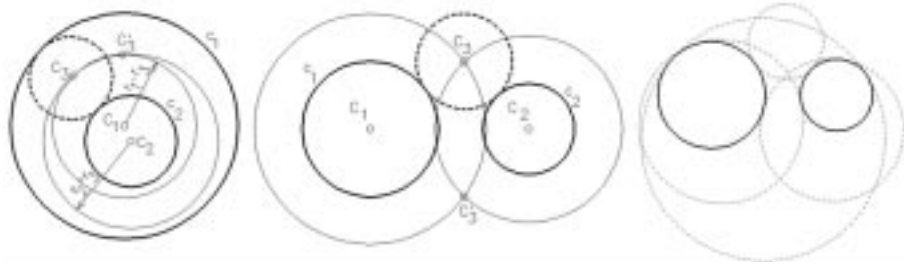
Exercise 3

Two circumferences are tangents if they share a single common point. This common point is called the point of tangency and is on the line defined by the centres of the two circumferences.

You are to determine a circumference that is tangent to two others. You must draw the circumference and the radii that go through the points of tangency.

The problem of finding a circumference with a given radius ($C_3 r_3$) and tangent to two others ($C_1 r_1, C_2 r_2$) can be solved by tracing a circumference with its centre at C_1 and radius r_1-r_3 and another circumference with its centre at C_2 and radius r_2+r_3 , if the circumferences are interior. All we have to do is to draw the circumference with its centre at C_1 and with a radius r_1+r_3 and another circumference with its centre at C_2 and with a radius r_2+r_3 , if the circumferences are exterior. The points where the two circumferences intersect are the two possible centres of the circumference c_3 we are looking for. Depending on the value of r_3 , however, the solution can be three exterior circumferences or two interior and one exterior.

Data: distance between the centres C_1 and $C_2 = 85$ mm; $r_1 = 55$ mm, $r_2 = 38$ mm, and $r_3 = 108$ mm. If more than one solution exists, the one with the highest centre position must be chosen.

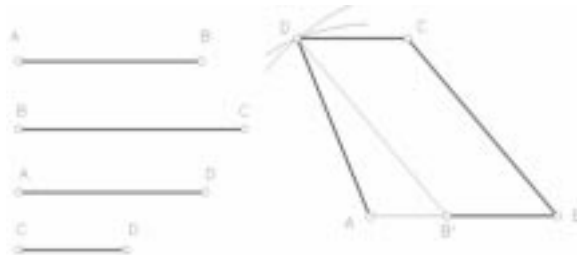


Exercise 4

A trapezium is a quadrilateral that has two parallel sides (which are called bases or basic sides) and the distance between them is called the height.

To determine a trapezium, four data are needed (for example, the lengths of its four sides), since the condition that the bases are parallel is the fifth datum of this type of quadrilateral.

To construct a trapezium, first we draw the auxiliary triangle defined by the two non-basic sides and one side which is equal to the subtraction of the two bases. Thus, if AB and CD are the basic (parallel) sides, we draw a triangle with sides $AB' = AB - CD$, $B'D = BC$ and AD . We then draw AB over AB' and CD parallel to AB and going through D . Lastly, BC is drawn.



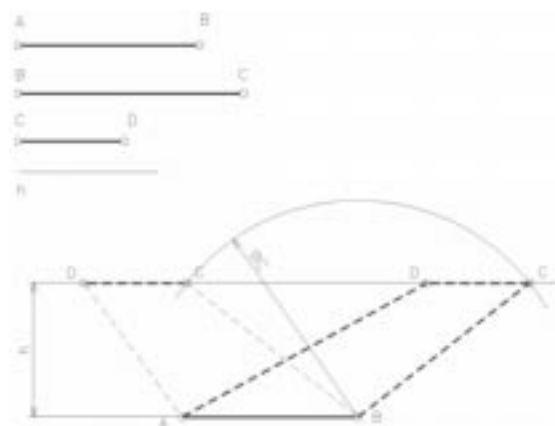
Data: $AB = 60$ mm; $BC = 75$ mm; $AD = 66$ mm and $CD = 38$ mm.

Exercise 5

A quadrilateral is a closed figure made up of four segments that are connected at their vertices. It is a particular type of irregular polygon. The segments that join alternate vertices are known as diagonals. A trapezium has a pair of parallel sides (which are called bases or basic sides) and the distance between them is called the height.

You are to construct a trapezium whose bases, a third side and the height are known (AB, CD, BC and h in the figure).

The classical method of drawing this figure consists in drawing two parallel lines at a distance equal to the height. On one of them we draw one of the two basic sides (AB , for example) and from the corresponding end we draw an arc with a radius equal to the non-basic side (from A if it is AD and from B if it is BC). The third vertex of the trapezium (D or C) is at the point where the arc intersects the second parallel line. We measure the basic side from this point and thus obtain the fourth vertex. The problem, in general, obviously has two solutions in each semiplane.



Data: $AB = 64$ mm; $CD = 42$ mm; $BC = 95$ mm, and $h = 50$ mm.