Mathematical qualifications for Using a CAD Program

By Burkhard Alpers

Aalen University of Applied Sciences, Germany

Abstract

It is well accepted that mathematics plays an important role in modern engineering but most of it is mediated by computer programs like CAD, FEM or machine element computation programs. This contribution investigates for CAD programs, which mathematical qualifications are still important for reasonable and efficient use. Geometric and algebraic qualifications are identified where the geometric ones differ from those usually taught in analytic geometry courses. Some conclusions are drawn on how this may inform the mathematics education of mechanical engineering students.

1. Introduction

In the daily life of a mechanical engineer, contact with mathematics is usually mediated by computer programs like machine element computation programs, CAD programs or finite element software. It has often been claimed that although much of the mathematics is "buried" in the software, an engineer still has to have a good mathematical understanding in order to make reasonable use of the software. It is the goal of the research reported in this contribution to shed more light on this claim by investigating the necessary mathematical qualifications for using a CAD program.

CAD programs are very large software systems with many features for different user groups ranging from "normal" design work to using an application programming interface in order to develop additional features. Since we are particularly interested in the daily work of an engineer, we focus on those parts which normally appear in constructive design work. The majority of our graduates find a position in this area since our institution, being a university of applied sciences, is oriented towards practical work. For identifying these parts, a colleague (lecturer for machine elements and design) who worked for several years in the car industry set up a typical design task (design of a support for an ABS box with some restrictions concerning stress and eigenfrequencies) which was solved by two students in their last semester who had already spent two practical semesters in industry. The colleague served as a mentor and provided information on how well the work of the students reflected the practical work of engineers. We let the students document their work process, interviewed them and let them show how they set up their design using typical industrial CAD programs (Pro/Engineer® and SolidEdge®). Moreover, the author investigated one of the CAD programmes more intensely in order to get information on other mathematically interesting features that did not show up during the design task.

The structure of the paper is as follows: Section 2 contains a description of the task and shows solutions developed by the students. In section 3 we describe the geometric concepts and procedures that must be known for making reasonable use of a CAD program. We provide some examples mostly from the task described in the previous section. In section 4 we give an overview of algebraic qualifications. Since these were not needed in the work on the task, this section is based on a closer investigation of a CAD program. A colleague was interviewed in order to get a better understanding of the importance of the respective features for the daily

work of an engineer. In section 5, we draw some conclusions and outline consequences for the mathematical education of mechanical engineers.

The research reported in this paper is part of a larger project investigating the mathematical expertise of mechanical engineers (Alpers, 2006). Workplace studies have been in the focus of research on mathematical education for some years (Bessot/Ridgway, 2000) but there are just a few studies for the university level (Kent/Noss 2002, 2003). In those, the mediation of mathematics by computer programs has been identified as an essential feature. For CAD programs, this is investigated in more depth in this contribution.

2. The task: Construction of a support for an ABS box

For an ABS box (ABS: Automated Braking System) a support is to be constructed within a defined installation space. The outside dimensions of the aggregate, the dimensions of the installation space and the coordinates of the attachment points can be obtained from CAD files (see figure 1). The carrier should meet the following requirements:

- The attachment points for the aggregate must be accessible.
- Strength: Stress and strain of the material (simple steel) should stay in the linear-elastic range.
- Stiffness: The carrier must be sufficiently stiff such that its first eigenfrequency is above 250 Hz.
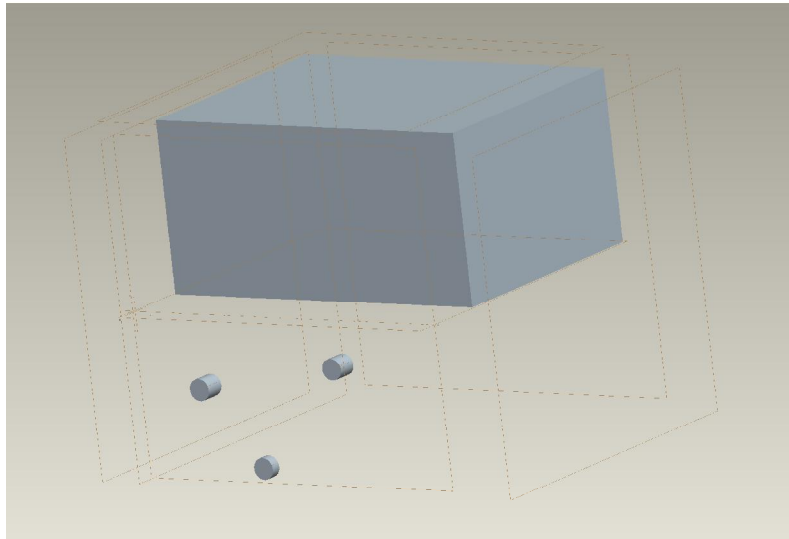- The carrier should be as lightweight as possible.

FIGURE 1. Position of box, installation space and attachment points.

The students which were hired for solving the task used the CAD programs Pro/Engineer® and SolidEdge® to make their designs and other programmes to check the requirements concerning strength and stiffness which are not considered here (cf. Alpers, 2006). Figure 2 shows the solution geometries the students came up with. The mathematical qualifications necessary for constructing these geometries provide the main input to the considerations in the subsequent section.
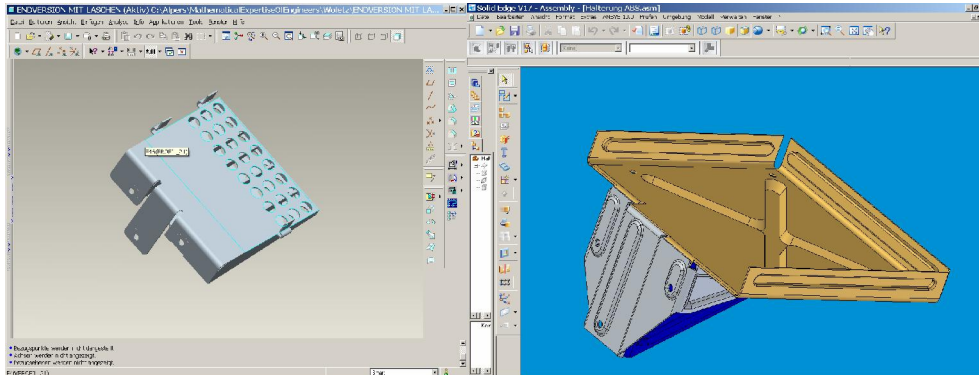
FIGURE 2. Constructions provided by students.

3. Geometric qualifications

Velichova (2002) has already identified geometric concepts that are important for engineering work in general. But there is no distinction as to whether knowledge and skills concerning these concepts are necessary for *producing* tools like CAD systems or for merely *using* them to work on design tasks. It is our intention here to identify as precisely as possible the knowledge and skills that are required from the *user* since our graduates will work in general on the user side. The following table gives an overview of the qualifications:

(i.) Changing the view by translation, rotation and zooming

(ii.) Finding algorithms for the construction of parts using basic objects and set operations (union, intersection, minus)

(iii.) Knowledge of basic geometric objects like point, line, plane and concepts like coordinate system and orientation

(iv.) Generation of 2D cross sections using geometric relations (e.g. parallelism, orthogonality, equality of properties like length)

(v.) Generation of 3D bodies from 2D sketches using geometric operations (extrusion, rotation, ...)

(vi.) Orientation of 2D views (fixing a 2D view for specification and extension of 2D figures into three dimensional space)

(vii.) Production structure of a construction (parent-child-relationships between objects)

(viii.) Avoiding problems with numerical inaccuracy by using references instead of absolute values

(ix.) Understanding characteristic parameters in predefined geometric application objects (e.g. drill hole, spline, chamfer)

(x.) Integration of production aspects

(xi.) Develop a 3D object into a 2D object for production reasons

(xii.) Flexible knowledge about alternative ways of doing things

TABLE 1. Geometric qualifications

In the sequel, we describe these qualifications in more detail and give some examples.

(i.)   Changing the view by translation, rotation and zooming

In order to display a 3D body on the screen, it is orthogonally projected onto a plane, the so-called image plane. The projection is also called "trimetric" since the images of the coordinate unit vectors of an underlying coordinate system have different length.
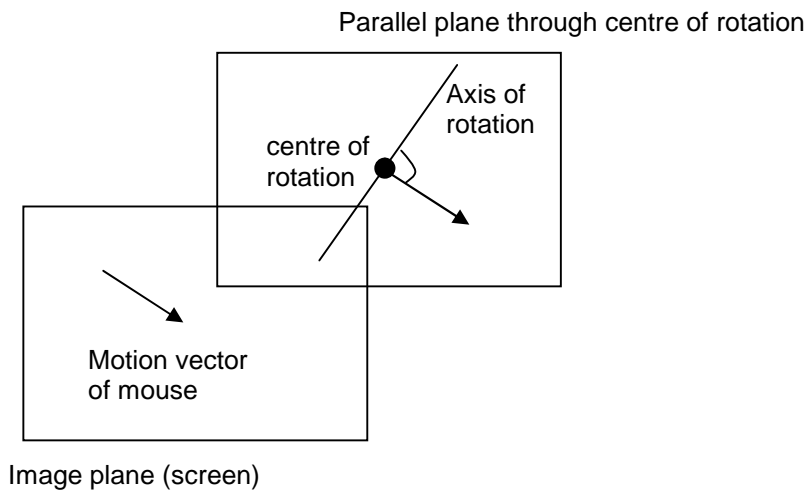

FIGURE 3. User action and rotation.

When the user wants to change the view, there are three operations at hand: translation, rotation and zooming. Translation operations move the body parallel to the image plane and are simply characterised by the translation vector. Rotations can not be described and performed that easily. In general, a rotation is fixed by specifying the axis and the angle of rotation. For doing this, there is a so-called "center of rotation" and a corresponding coordinate system. The default for the center is the so-called center of the model (which is "somewhere in the middle" of all depicted objects including reference planes but it is not described in more detail). If the mouse pointer is moved into a certain direction on the screen the model rotates where the axis of rotation is the line orthogonal to the motion vector of the mouse which contains the center of rotation and lies in the plane through this center which is parallel to the image plane (Figure 3). The rotation is independent of the starting point of the mouse but the angle of rotation depends on the length of the motion vector (so this is also a nice example for a "free" vector). This is quite complicated (and there are even more possibilities not mentioned here), and questioning an engineer who is very fluent with the system revealed that he could not describe the rotation in mathematical terms. Nonetheless, he was able to use the rotation in order to provide the view he wanted to obtain. This phenomenon can be interpreted as "understanding through use" as it is called by Kent and Noss (2002, 2003). The immediate optical feedback the user gets from the screen enables him to find the necessary dragging motions quite easily. If the author tries to generate a certain view having the knowledge on the rotation axis and angle, he is no more efficient, to say the least. As a mathematician one is usually accustomed to try to understand phenomena exactly whereas the engineer as user can achieve what he wants simply by trying and getting feedback having an adequate tool. We definitely do not claim that this extends to the whole area of work in engineering but we consider this as an example where mathematical thinking and engineering use diverge.

(ii.)   <u>Finding algorithms for the construction of objects using basic parts and set operations (union, intersection, minus)</u>

Designing a machine or a mechanism essentially consists of designing its parts using basic building blocks (the construction of such blocks using plane sketches is described further below). The blocks are added or one block is (in the sense of point sets) subtracted from the other one. From the viewpoint of real production the latter can be considered as removing material but a special production method is not implied here. Mathematically, the operations can be interpreted as union $A \cup B$ and minus (A\B: A minus B) operations with point sets where, however, boundary points are always preserved (for an engineer the question of whether the body without the boundary points is still the same is rather meaningless). In the CAD systems we investigated it is also possible to remove the complement of a body (i.e. for a body B the set $\mathbb{R}^3 \setminus B$). Therefore, one can construct intersections of bodies A and B by building $A \setminus (\mathbb{R}^3 \setminus B)$. One could even generate an intersection by merely using union and minus operations: $A \cap B = (A \cup B) \setminus (A \setminus B \cup B \setminus A)$. But this certainly is not very efficient. Figure 4 provides an example for the set operations.
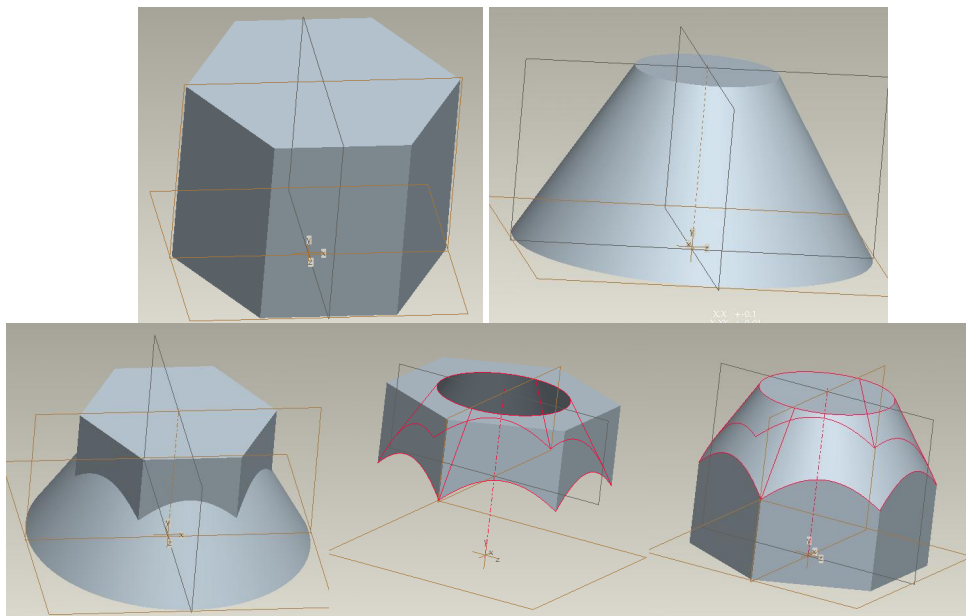

FIGURE 4 (Pro/Engineer ®). Example for union, intersection and minus (after Stürmer, 2004, p.49)

The main qualification on the part of the user consists of turning a rough image of the part that exists in his imagination into an algorithm for constructing it in the CAD system using the operations available. Additionally, this algorithm should be as simple as possible such that it can be easily understood and modified (maybe by different users) later on. One possible way to proceed is to construct an "enveloping" body first and then remove material instead of starting from a complicated sketch of a cross section and extruding it afterwards. Figure 5 shows how one of the students used this strategy while designing the support of the ABS box. Figure 6 depicts the opposite strategy for creating a rough reconstruction of a brake calliper where simple geometric objects were added.

Finally, from all the single parts the overall machine is built (which is called assembly) but here, the parts are just added and the main task consists of placing the parts by defining how they are ordered in space. For the latter, sometimes separate common reference objects (e.g. planes) are defined in the single parts such that one can change one part without getting into the danger of destroying the fit (cf. the next item).
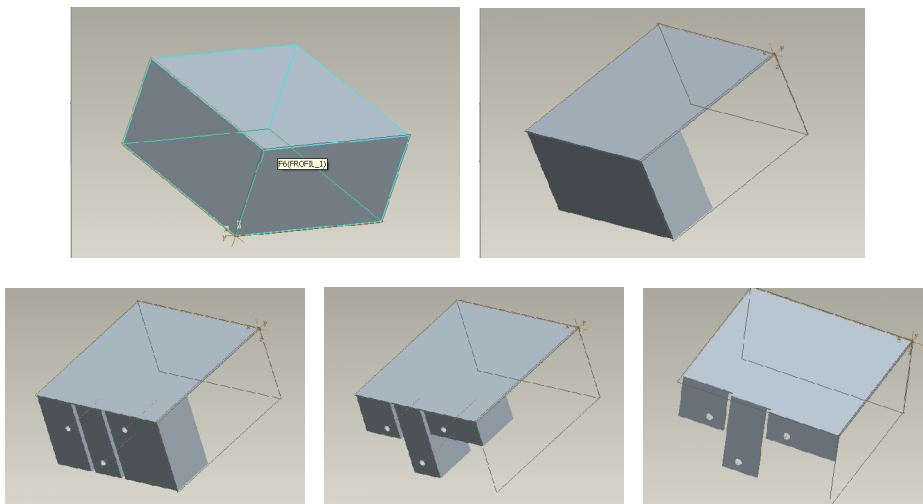

FIGURE 5 (Pro/Engineer®). Constructing the support of an ABS box.

When generating a new part by adding or removing material one can often make exact numeric specifications (e.g. make a sketch and generate a body by the method of extrusion using an extrusion depth of 30mm). Alternatively, one can specify that material is removed in a certain direction in arbitrary depth. When parts are added later, the removal is also applied to these parts (e.g. when a drill hole is to be extended). Moreover, one can specify a certain surface up to which the removal should be performed. Such a specification allows easy modification afterwards whereas fixed numerical declarations must be changed one by one. In order to make an easily modifiable design, the user should think in relations ("extends up to") rather than in absolute values ("has a length of").
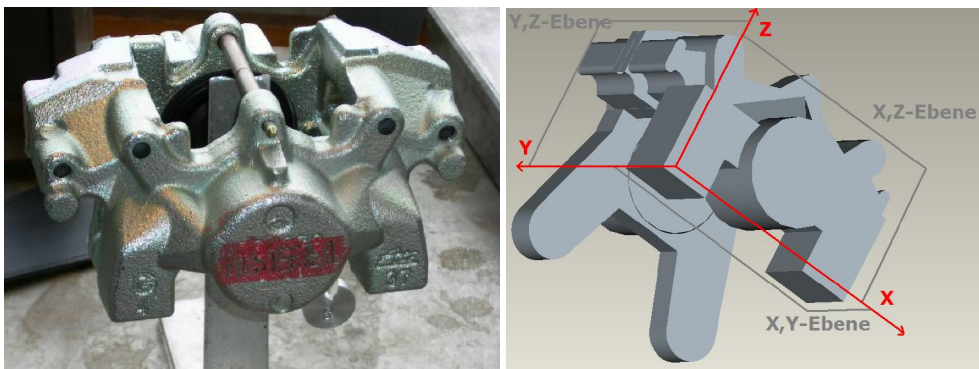

FIGURE 6 (Pro/Engineer®). Reconstruction of brake calliper.

(iii.)   <u>Knowledge of basic geometric objects like point, line, plane and concepts like coordinate system and orientation</u>

The elementary affine objects like point, line, plane are essential when working with a CAD program (there is a distinction between finite objects such as line segments and infinite objects like axes but we neglect this in what follows). Most of the objects are defined by referencing such basic objects. If one wants to construct such an affine object one has to know by which data and other objects they are fixed since such data and references have to be specified. For example, a line (axis) can be specified by referencing two (different) planes which contain the line, or by two points lying on the line, or by a point lying on the line and a plane orthogonal to the line. The basic planes which are created right at the start are the coordinate planes in a coordinate system (right-handed system). These planes – like any other plane – have an orientation, i.e. a "positive" and a "negative" side (more precisely: half-space). With the coordinate planes, the positive side is the one the third coordinate unit vector points to. A position in three-dimensional space can be fixed by translating and/or rotating reference objects (axes, planes) and by using the new reference objects later on. So, the referencing method can also be used for creating new reference objects from existing ones. Here, the user must think in an operative or algorithmic way but not (as is usually done in mathematics education) in an algebraic way.

As an example consider the body in figure 7. After the construction of the enveloping cuboid the material in the left upper corner is to be removed. For doing this, one must construct a plane through the points making up the corners of the triangle. First, these points are created as reference points. Such a point can be fixed by specifying that it lies on two faces of the cuboid (the faces can also be referenced) and has a certain (signed) distance to a third face. Finally, a body is constructed on this plane by extrusion and this body is then (set-theoretically) subtracted from the cuboid. There are several alternative methods, so the above description serves just for illustration purposes (in Pro/Engineer® one can even construct chamfers for corners not just for edges, so this is a "shortcut" avoiding the construction of the inclined plane).
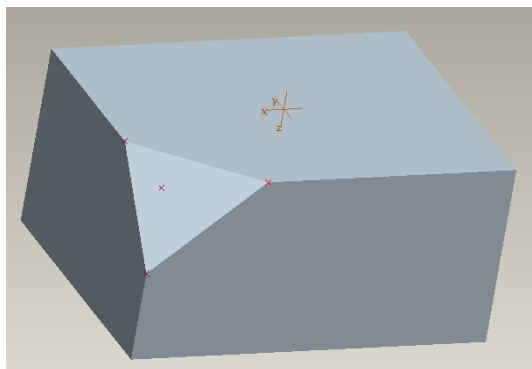

FIGURE 7 (Pro/Engineer®). Cuboid with cut-off.

When the parts of a design have been constructed, each having a local system of coordinates, the parts must be put together (called "assembly") in order to complete the construction. This "putting together" means that the position of the parts relative to each other must be fixed (and hence the relative position of the corresponding local coordinate systems). This is not performed in an algebraic way, e.g. by setting up transformation matrices. Rather, the relative position is fixed by identifying common objects in two systems, e.g. a plane in one system is

identified with another plane in the second system. Again, the question is by which references the relative position is uniquely determined. Since planes have an orientation, one can include orientation when identifying planes and thus make the position specification complete. If the user chooses the wrong orientation, he can simply switch orientation, so a deeper thinking in advance is not required. One can also identify two lines in two systems with each other but then one has no way to specify how the lines are placed on each other since lines do not have an orientation (for whatsoever reason). Figure 8 shows three reference planes that were used to define the assembly of the three sheet metal parts one of the students constructed. An alternative way consists of setting up a "skeleton" of reference objects first which are then used in the construction of each part in order to place it correctly.
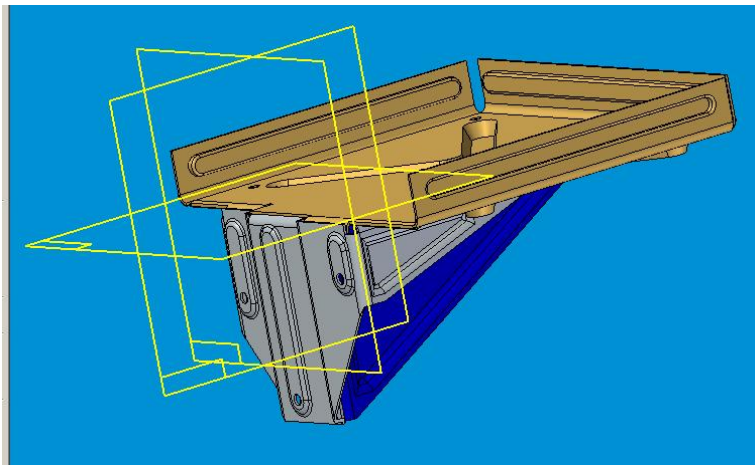

FIGURE 8 (SolidEdge®). Reference planes used for assembly.

(iv.)   Generation of 2D cross sections using geometric relations (e.g. parallelism, orthogonality, equality of properties like length)

3D-CAD programs include a 2D so-called sketcher for producing cross sections containing lines, circles, rectangles and also curves with variable curvature like splines. The sketcher tries to support the user in quickly setting up rough sketches by providing a catch mode for end points and an automatic way of determining possibly intended relations (parallelism, orthogonality, equality of length, ...). Based upon this, the user can then provide the exact measures and – if not yet included correctly – the relations between geometric elements of the sketch.

In order to use this tool efficiently (and not get messages on over determination all the time) the user has to recognise by which relations and data the intended geometric figure is uniquely determined. As is also the case with dynamic geometry programs, the user should identify essential properties of the figures which should be preserved when the figure is modified. Only if this is the case can single measurements be adapted easily without having to change all the measurements in the sketch. We call this "design for modification".

As an example, consider part of the sketch for a crimping produced by one of the students (figure 9). The student's preferred method of fixing the shape was by providing measurements. Parallelism was achieved by picking two points of a line section and determining the distance to the opposite line section as 10. The left upper corner of the sheet is placed on the middle line between the line sections by specifying that its distance to one of the line sections is 5. The distances equal to 25 (between points) guarantee that the line sections

have equal length and can be closed using half circles. Note that the specification could have been set up using relations like parallelism and symmetry making later modification easier.
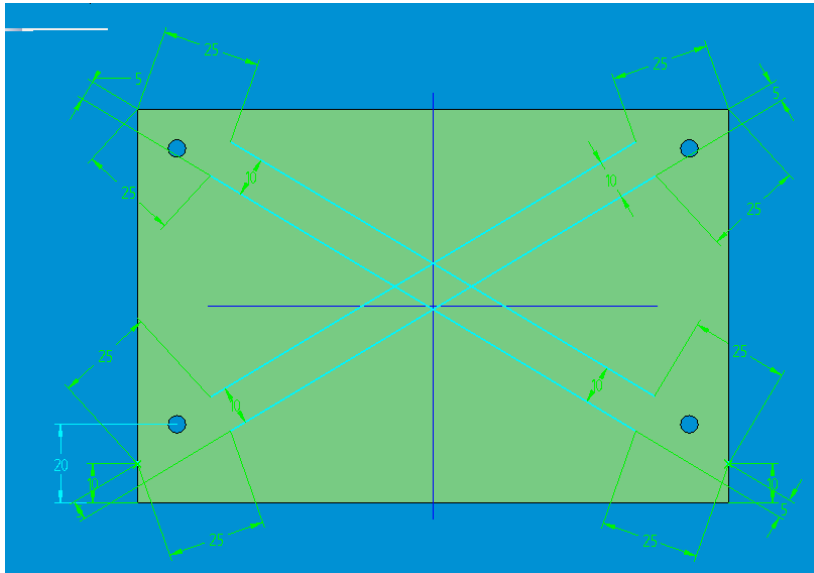

FIGURE 9 (SolidEdge®). Part of sketch for cross-section of crimping.

Allowing the sketcher to make implicit assumptions on the intentions of the user and to set up relations respectively can also lead to problems since later on (when trying to change measurements) over-determination can occur and recognising all the relations involved in the problem can be quite tedious. Changing of measurements has also to be performed with care since such a change can turn down or distort the figure such that recognising the parts of the figure can be hard. Distortion can be avoided when all measurements are adapted using the same factor (i.e. scaled). Then, the shape of the figure is preserved.
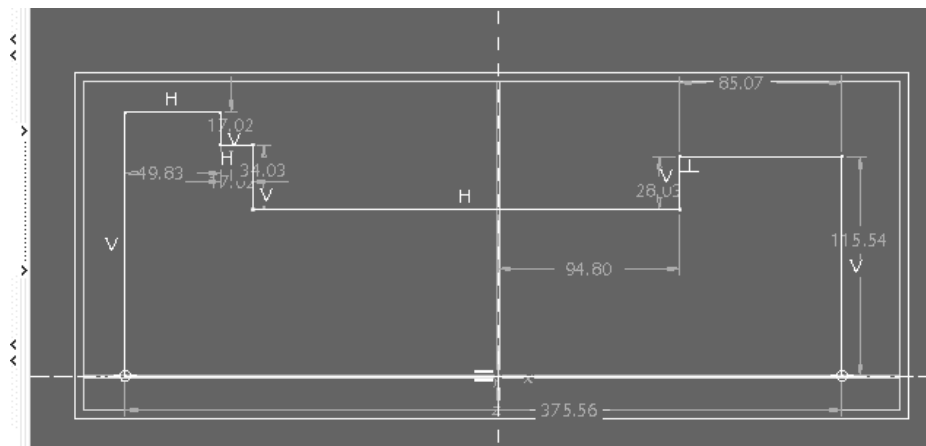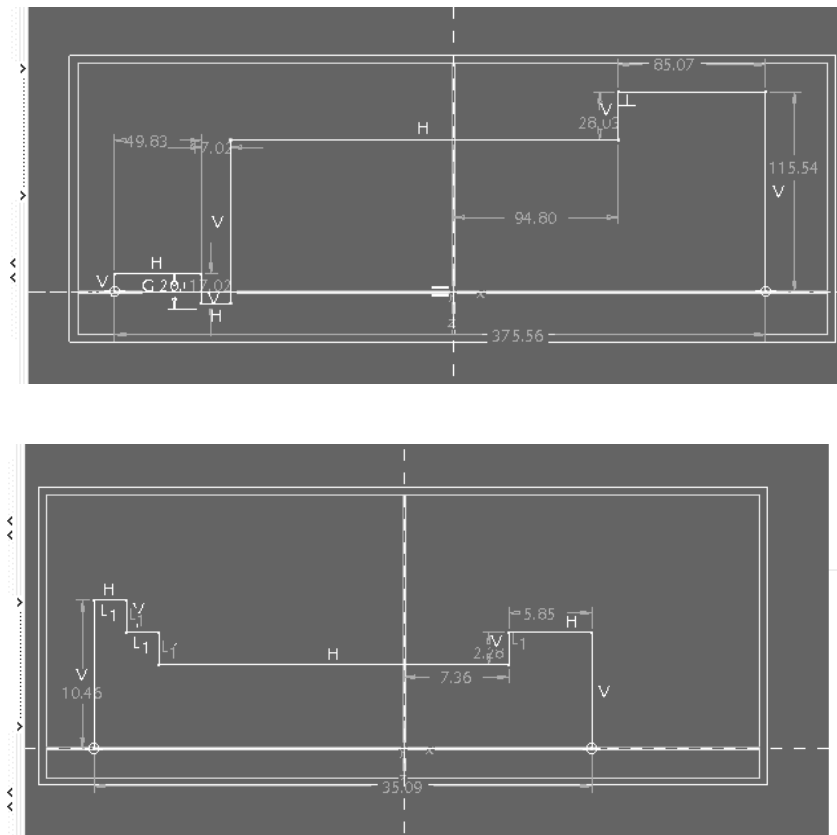
FIGURE 10 (Pro/Engineer®). Half a cross-section of a shaft with different diameters, a distorting change, and automatic specification of relations.

Figure 10 shows half of a cross-section of a shaft with different diameters. The first picture shows a rough sketch of the intended shape where measurements are automatically created by the program. In the second picture, the first height was specified. Since the numerical value was much smaller than the height set up automatically, there is an unwanted turn down. The third picture gives an example of how relations that are set up automatically by the sketcher can be disturbing. The equality of distances termed L1 is not wanted and must be removed.

(v.) <u>Generation of 3D bodies from 2D sketches using geometric operations (extrusion, rotation, ...)</u>

The single parts making up the overall construction are mostly designed from two-dimensional sketches using extrusion or rotation operations. There are also other operations for creating surfaces from curves (e.g. Coons patches) or bodies from surfaces (e.g. connecting two surfaces) but in most constructions they are not used (and if they are applied the user does not have to know the mathematical representations and algorithms they are based upon). The user has to decide which method works for constructing the desired part (sometimes, both rotation and extrusion work when the part is cylindric and symmetric with respect to rotation). First, the plane on which to place the sketch must be created (cf. figure 11 which contains all the reference planes used in the design of the ABS box). Then, a generating 2D-sketch must be designed and created. Finally, depth and direction of extrusion or angle and orientation of

rotation have to be fixed. When the principle shape of a part has been built in this way, details are added like drill holes, fillets or chamfers (compare figures 2 and 5). The latter are pre-defined objects in a CAD system and are discussed in more detail below.
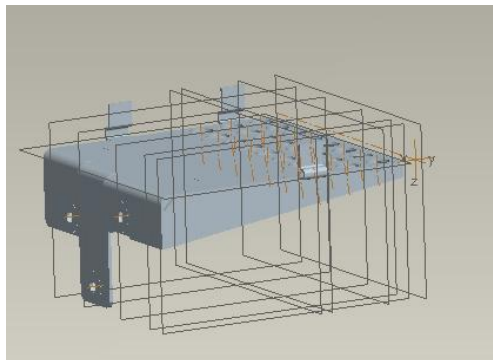

FIGURE 11 (Pro/Engineer®). Reference planes for support design.

(vi.)   Orientation of and in 2D views (fixing a 2D view for specification and extension of 2D figures into three dimensional space)

When a plane is chosen (or has been constructed) for placing a sketch on it, one has to determine first which side of the plane to look at in the sketcher by choosing an orientation. Secondly, one has to specify an intersecting oriented object in order to fix what is left and what is right (or upper and lower part). Moreover, for fixing the position of objects in the plane, one needs references with respect to which the objects can be placed. Such references are usually intersections of the plane with other objects. Normally, two planes intersecting the sketch plane are specified as references such that a two-dimensional coordinate system is set up. Then, the position of objects can be determined by providing metric or relational information with respect to these lines of reference. The description shows that orientation aspects play an important role in trying to get a certain view and the reference system. The task is made considerably easier by offering the user to switch orientations with a mouse click. As a consequence, the user can simply try one orientation and switch if it turns out to be the wrong one. So, thorough thinking in advance is not necessarily required.

Once the sketch has been made, one has to provide information on the direction of extrusion or the orientation of rotation but again the easy switching feature makes thorough thinking obsolete here.

(vii.)   Production structure of a construction (parent-child-relationships between objects)

Constructions even of a single part of an assembly can be quite complex. A tree structure showing the interdependencies and the sequence of production serves to retain an overview and make changes. Figure 12 depicts a part of such a tree for the ABS box design shown in figure 2 (left solution). "S2D0001", for example, is the name of a sketch used for constructing the 3D-body "Profil1" by extrusion. For constructing a new object, often existing objects are used (e.g. for removing material from them or for using them as references). The corresponding relationships between objects are called parent-child relationships. When an object is deleted, all the child objects are also deleted for avoiding inconsistencies. Therefore, the user must have an overview of the relational structure of the objects constructed so far. It

is recommended in text books on CAD to use just a few basic reference objects which all other objects are related to in order to keep the relational structure simple.

The sequential structure of the tree can also be used to retrace the construction process: Running down the tree effects that one object after the other is added to the figure depicted on screen. The tree is also used for inserting objects later on or for choosing objects that are to be modified.
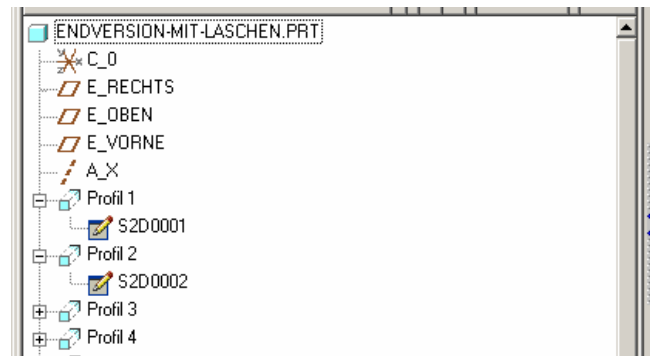

FIGURE 12 (Pro/Engineer®). Part of the object tree of the support.

(viii.)   Avoiding problems with numerical inaccuracy by using references instead of absolute values

Frequently, there are several ways to determine a geometry or a position. One can reference other objects (specifying, e.g., an extrusion depth by providing a surface object that serves as a boundary surface) or one can insert absolute metric values. The latter can cause problems because of numerical inaccuracies. For instance, there can be problems for the CAD system to recognise that bodies are attached to each other. The user should know about the possibility of such inaccuracies, i.e. he should not have a "naive" concept image concerning the exactness of numerical specifications. He should apply this knowledge by choosing construction methods (in general by using references) that avoid such problems.

(ix.)   Understanding characteristic parameters in predefined geometric application objects (e.g. drill hole, fin, chamfer)

Besides constructing bodies from scratch it is also possible in Pro/Engineer® and SolidEdge® to use pre-constructed geometries with application meaning like drill hole, fin, or chamfer. The shape of such geometries is specified by parameters as is shown for drill holes in figure 13 (left part). For doing this in a reasonable way, it is essential to know the meaning of these parameters. Moreover, the position must be fixed by pointing to referenced objects and inserting numerical values for distances or angles (figure 13, right part). Normally, there is no use of coordinates but one could create reference coordinate systems and then use coordinates as reference data but this is hardly ever done.
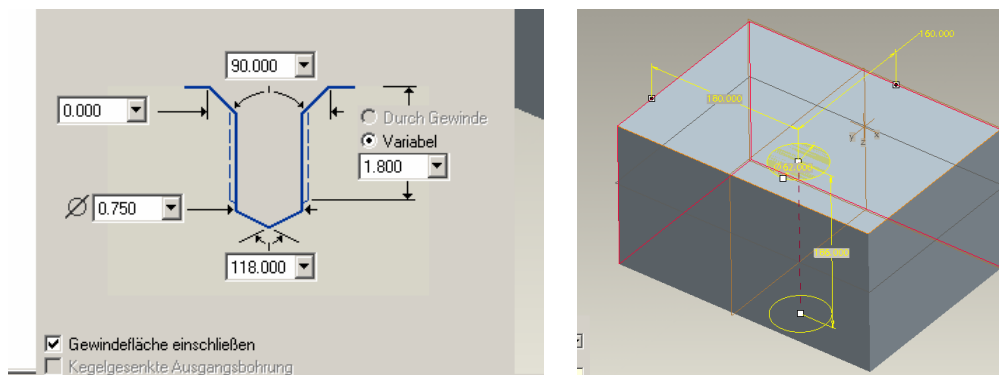
FIGURE 13 (Pro/Engineer®). Placement and properties of a drill hole.

(x.)   Integration of production aspects

Geometric design and production are separate activities. If in geometric design material (or rather: volume) is removed in order to create a certain shape, then this does not imply a specific production process (milling, turning, grinding, …). How fillets are realised in production is also in no way determined by the geometric specification in a CAD program. Nevertheless, simultaneous thinking about production methods (including assembly) is quite important for cost-effective production. Moreover, there is even direct support for turning a construction into something that can be used for production. There is, for example, a sheet metal mode which supports the construction of parts built from sheet metal (e.g. insertion of crimpings) and which also provides data for the production process. Bent sheet metal can be developed into plane sheets (if possible geometrically, see for more detail the next item) and this way the contour can be determined which has to be given to a laser cutting device. Moreover, the radii for the bending process are computed. Even in "normal" geometric construction, data important for the production process is generated, e.g. when a boundary surface is computed providing essential information for a milling process.

If there is no support within the CAD system, it is up to the user himself to create parts which can be produced cost-effectively. E.g., in parts to be produced by bending, one has to include cut-outs as is shown in figure 14 in order to reduce stress although from a purely geometric point of view they are of no use. When creating fillets as in figure 2 it is questionable whether in real production the radii will be the same as in the construction. In a bending process, e.g., the pair of radii (interior and exterior) cannot be fixed completely. The interesting question here is whether or not a small difference between design and produced part really matters.
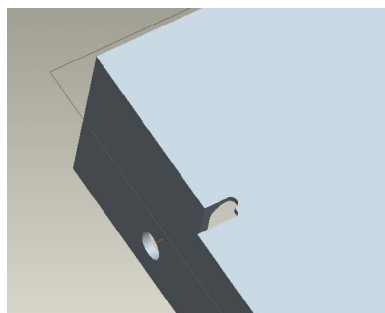

FIGURE 14 (Pro/Engineer®). Additional cut-out to reduce stress.

Parts should be constructed in a way such that an assembly is possible even if the single parts have production tolerances. In figure 15 below, there is a cut-out between the fin and the bent metal such that an exact production of both parts is not necessary. Therefore, from a production point of view it is not wise to go exactly to boundary points or lines but to stop "a bit" before. This is somewhat contrary to the precise way of doing geometric constructions preferred by mathematicians.
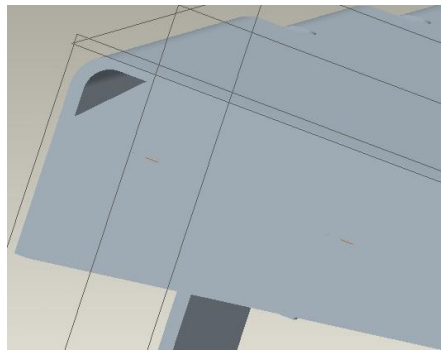

FIGURE 15 (Pro/Engineer®). Cut-out between fin and bent metal.

(xi.) <u>Develop a bent object (in 3D) into a 2D object for production reasons</u>

As stated above, the sheet metal mode supports the generation of information for the production process. This includes developing a bent sheet into the plane. Clearly, this cannot work when the sheet has been modified in a way that distorts the developing process. E.g. when crimpings have been inserted in order to enlarge stiffness, this is no longer possible. If the crimpings are placed in plane regions of the bent metal this could be ignored. But if the crimpings are placed on bent parts this is no longer possible; this is also evident from the very purpose of crimpings: to prevent the bent metal from changing the bending. So, in order to use the sheet metal mode properly, the user should have a geometric understanding of developing a surface into a plane and of what geometric modifications destroy this process.

It has to be added that idiosyncrasies of the CAD system used are also important, so it is not just geometrical reasoning. In SolidEdge® for example, developing was impossible after inserting any crimpings. In Pro/Engineer®, however, it was possible as long as the crimpings are placed in plane regions.

(xii.) <u>Flexible knowledge about alternative ways of doing things</u>

As is the case in analytic and synthetic geometry, there are often several ways of achieving a certain construction goal. In order to specify the relative position of two parts in an assembly, one can identify common planes or common lines. If one possibility does not lead to a satisfactory solution (maybe because of idiosyncrasies of the tool) then it is important to have alternatives at hand. Therefore, the engineer should know a variety of procedures, such that he can react flexibly in such situations. Even if other ways are more complicated under "normal circumstances", they might be quite helpful as workaround when the easiest way does not work.

4. Algebraic qualifications

When working on the specific task described in section 2, the students did not have to use any algebraic specifications of geometric objects like linear equations or vector representations for lines and planes. At least in a conventional mathematics curriculum for mechanical engineers, such representations are used for description and computation. In the CAD systems under consideration, shape and position of objects were defined by using operations, references and data. This does not mean, however, that the CAD programs are free of any algebra. In the sequel, we present some algebraic specifications available in parametric CAD systems which are important in more advanced usage and more complex modelling. Table 2 gives an overview of these features (the list is not meant to be exhaustive).

(i.) Specification of algebraic dependencies in parametric constructions
(ii.) Functional thinking for goal-oriented variation in a large design space
(iii.) Algebraic specification of curves in parameter representation
(iv.) Algebraic representations of geometric objects in exchange data formats

TABLE 2. Algebraic qualifications

(i.)   Specification of algebraic dependencies in parametric constructions

With parametric CAD system one can create flexible constructions where some measurements can be specified depending on other measurements. For example, one can specify that a certain distance a should be twice as long as another distance b. For this, symbolic names for the distances under consideration must be introduced, and then the dependency must be expressed algebraically: $a=2b$ (for a more sophisticated example on designing series of gears see Raytchev, 2001). This way, it is possible to easily set up a type series for a certain kind of machine parts, e.g. for gear wheels. There are (at best only a few) basic independent quantities and dependent quantities, and by changing the independent quantities a new object can be created. Here, the user must think in symbols which have a concrete application meaning, and this meaning is (for better understanding in later considerations) frequently reflected in the names of the variables. Such a usage was already recognized by (Wake/Williams, 2003) when Excel®-Worksheets were used for flexible computing and cell names indicated the meaning of cell content.

Even if the user is not capable of setting up algebraic dependencies himself, he should at least be able to understand and use parametric constructions set up by others. Although parametric construction provides a very easy way of modification, it is scarcely used in "normal" (non series) constructions since the other features of a CAD programs often allow already for sufficient ease in modification while preserving geometric relationships.

(ii.)   Functional thinking for goal-oriented variation in a large design space

When an initial construction has been set up or an existing construction is given, one can vary this. There are lots of quantities which can be varied like measurements, number of fins, position of fins or drill holes, properties of the material used and so on. So, there are not only many variables but they also have different types (real, integer, enumeration of properties, ...). The user must be capable of navigating in this huge space of variables in a reasonable and goal-oriented way. But in general – as stated above – the relationships between variables are not given in an algebraic symbolic way. Nevertheless, in Pro/Engineer® there are so called

analysis tools for investigating functional relationships. One can vary one quantity and the program creates a graph (and corresponding numerical data) showing the development of the dependent quantity. Therefore, functional thinking is required here even if no symbolic function expression is available. This can also be used for investigating mechanisms like cam devices constructed in the CAD program. Then, the motion of the output can be traced depending on the motion of the driving rod.

(iii.)   Algebraic specification of curves in parameter representation

When – in a non standard application – the user wants to specify a curve in parameter representation (be it as a boundary curve in a two-dimensional sketch or as a template curve to create more complex bent pipes), the user has to input the coordinate functions *x(t), y(t), z(t)* algebraically. Moreover, the system Pro/Engineer® requires the user to scale the parameter interval since *t* must run from 0 to 1. So, it might be necessary to re-parameterise the curve. Here, the respective algebraic knowledge must be present.

(iv.)   Algebraic representations of geometric objects in exchange data formats

A look at exchange data specifications (like IGES or STEP, for the latter see ISO 10303-42, 2000) reveals that the numerical values describe coefficients in algebraic representations of geometrical objects. So, the latter are obviously necessary for understanding the meaning of the data. In standard applications of a CAD, such exchange data is never investigated by the user. He simply saves his construction in a standard format and opens the file with another program. Yet, one has to deal in more detail with such data formats when one wants to set up such an exchange file "by hand" (or rather: by programming). This was for example the case in a diploma thesis written under the supervision of the author where the output of a FEM program was converted into a (standard) STEP file such that it was available for further processing in a CAD environment (surface reconstruction, cf. Alpers/Gantner, 2001). We mention this here to avoid the impression that there is just the standard usage of a CAD program.

5. Conclusions

The investigation provided some insight in the kind of mathematical qualifications needed for using CAD programs efficiently. Although the students themselves could not identify much mathematical thinking within their work, a closer look revealed that this originates from an identification of mathematics and work with symbols (algebra or differential calculus) which the students experience in their mathematical education. Although indeed, symbolic algebra and functions only show up in some advanced, more sophisticated areas of the CAD programs, the major part is also strongly influenced by mathematical concepts which can be roughly described as operational synthetic geometry for setting up constructions from basic blocks and relational geometry for fixing the geometry and the position of a part by defining relations (distances, parallelism, orthogonality and so on). In the contribution, we give some examples for these kinds of mathematics and we identify the necessary mathematical qualifications. This can serve to enhance the mathematical education of mechanical engineers in order to better prepare them for their daily work.
As a first consequence, we will integrate examples and tasks from the CAD area in the "normal" mathematics education. We think that set operations (union etc.) can be made much more meaningful for students when they have bodies in a CAD system as representations for

sets (besides number sets or Venn diagrams). One could for example provide a range of basic bodies and a more complex body and ask students to describe the complex one using the basic objects and set operations. They could then even check their work using a CAD system (which is used from semester 1 on). In geometry education, one can also deal with tasks where a sketch is given like the one in figure 9 (but without relations or measurements) and students have to think about a complete set of relations to determine it (maybe even requiring some properties to be preserved when one quantity is modified). Such tasks can be quite challenging and initiate deeper thinking processes than are necessary for carrying out computational procedures. As another positive side effect of this investigation, the author is better capable of understanding the geometrical thinking and arguing of the students that comes from their preoccupation with CAD programs.

REFERENCES

ALPERS, B. 2006. The mathematical expertise of mechanical engineers. Proc. SEFI MWG Seminar, Kongsberg (to appear).

ALPERS, B., GANTNER, P. 2001. Prozesskettenunterstützung für die Umformsimulation – Rückführung von Umformgeometrien in CAD-Systeme, Proc. 19th Int. CAD-FEM Users' Meeting, Potsdam.

BESSOT, A., RIDGWAY, J. (Eds.) 2000. Education for Mathematics in the Workplace. Dordrecht: Kluwer.

ISO 10303-42 2000. International standard: Industrial automation systems and integration-Product data representation and exchange-Part 42: Integrated generic resource: Geometric and topological representation, Geneva: ISO.

KENT, Ph., NOSS, R. 2002. The Mathematical Components of Engineering Expertise: The Relationship between Doing and Understanding Mathematics. IEE 2nd Annual Symposium on Engineering Education, London.

KENT, Ph., NOSS, R. 2003. Mathematics in the University Education of Engineers. A Report to the Ove Arup Foundation.

RAYTCHEV, R. 2001. Modellierung von Zahnrädern mit Pro/Engineer. I-NEWS, PTC-Magazin: PTC. (electronically in http://ww3.cad.de/foren/ubb/uploads/armino/Zahnradmodelierung1.pdf)

STÜRMER, U. 2004. Flächen- und Volumenmodellierung von Bauteilen mit Pro/Engineer Wildfire. München/Wien: Hanser Verlag.

VELICHOVA, D. 2000. Geometry in engineering education. *European Journal of Engineering Education* **27**/3, 289-296.

WAKE, G., WILLIAMS, J. 2003. Using Workplace Practice to Inform Curriculum Change. In: LAMON, S.J., PARKER, W.A., HOUSTON, S.K. (Eds.). Mathematical Modelling: A Way of Life. Proc. ICTMA 11. Chichester: Horwood, 189-199.

WYNDORPS, P. 2004. 3D-Konstruktion mit Pro/ENGINEER-Wildfire, 2nd Ed., Haan-Gruiten: Verlag Europa Lehrmittel.