

## **RECURSIVE MODELLING OF STEEL SKELETONS**

Markku Heinisuo

Rakenteiden Mekaniikka, Vol. 27

No. 3, 1994, pp. 3 - 21

### **ABSTRACT**

The paper deals with the recursive definitions of critical points when designing steel skeletons. The points are dependent on the previous points and other design data. The design data used are the initial data in the database and when considering connections the profile data. Twelve definitions are given in the paper. These are proved to be sufficient for the applications, which are: outlines of the steel skeletons, connections and analysing models of the steel skeletons. The system developed is especially suitable for the definition of connections. The code to generate the geometrical and analyzing models of connections remains valid although the amount of connections is increasing.

### **INTRODUCTION**

When modelling steel skeletons there are at least three main principles that must be followed:

- The geometrical model must be arbitrary
  
- The new model must be easy to define
  
- The old model must be easy to manipulate.

The first postulate is of the main importance. The layout of the building is almost always project dependent. Also, the connections of the skeleton are project dependent in spite of the standardization done by the codes (Eurocode 3, 1992) and other instances

(TSP, 1989). For instance, the program that handles the primitive entities of the steel skeleton, such as bars, must include the possibility that the cutting of the bar is done arbitrarily, i.e there may be different amount of cuttings of the bar (Fig. 1. a.). The amount of the cuttings is case dependent. Usually there are only two cuttings per end of the bar (Fig. 1. b.).

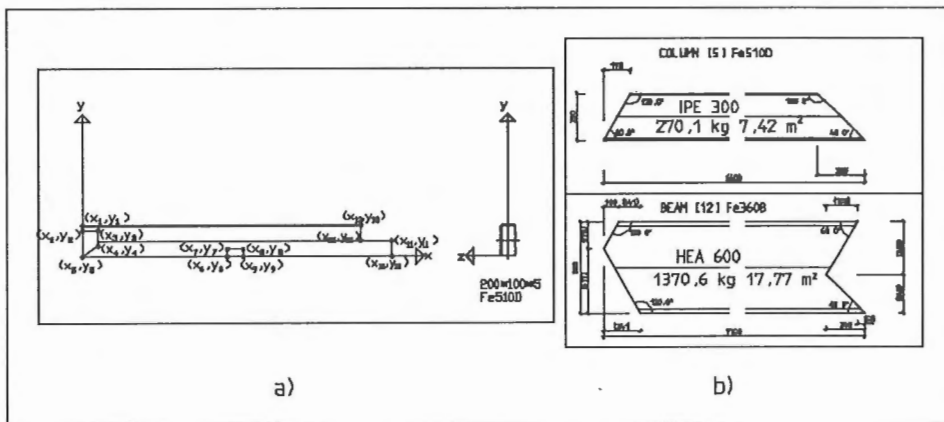


Fig. 1. Cuttings of a bar.

How to define the cuttings of the bars in the program? It seems easy to solve this problem: The number of the cuts is defined as an input to the program. Sometimes this (the number of cuts of the end) is dependent on the geometry of the other ends of the bar. Moreover, the geometry of the other end is dependent on the geometry of some other bar etc. This gives the idea of the recursive modelling of the bars and connections and moreover the whole skeleton.

The second postulate gives the idea of the user friendly programming. The graphical real time modelling is of the main importance, so that the designer can see all the time what he is doing. Also, the definitions and methods used must include the minimum principle: there must be those and only those definitions that are needed in practice.

If the designer has modelled e.g. a steel frame with span  $L = 10$  m and now he has to model a similar frame with span  $L = 9,5$  m (other initial data are the same) then it must be possible to do this by changing only one data in the model. The third postulate above means this. The example of the frame seems very simple but if the frame includes a truss then it becomes more complicated to handle the situation (stretching of the frame) by using trivial CAD-program's alternatives. It must be noted that gaps, fittings and other joint equipments were planned to remain unchanged when stretching the frame above.

The steel skeleton is a structural unit, and it includes structural members (beams, columns, bracings, trusses) connected by structural connections. The parts of the members are bars, fittings, bolts (including washers and nuts) and welds (Heinisuo, 1994). All these can be modelled by using standard CAD-programs. If the CAD-program utilizes the object oriented features then it is possible to define the relations between objects (see Bento et al., 1994). All the ways to model the skeletons which are nowadays in the use in the commercial programs are not known, because these are not available in the public literature (see e.g. Torp-Pedersen et al., 1991).

The standard CAD-programs (e.g. AutoCAD, trademark of Autodesk Inc.) include statements, that can be used to define relations between drawing objects when making the geometrical model of the steel skeleton. These are **C**enter, **P**erpendicular, **T**angent etc. The use of these statements can be defined by programming tools (e.g. AutoLISP when using AutoCAD). By this way it is possible to write a program that makes the recursive relationships between objects. In the following, it is proposed a macro language for the recursive programming. The implementation to the computer can be done by using any language available. The recursive system developed is applied to the geometric modelling of the steel skeleton (outline) and its connections and to the automatic generation of the analysing model from the geometrical model. The applications of the system have been presented before (Heinisuo, et al., 1991). The definitions and an idea of the iteration between geometrical and analyzing models are given in this paper. The system proposed is applied to two dimensional problems.

## DEFINITIONS OF POINTS

The most important entity when defining the geometrical model of the steel skeleton is a point. The structural units are defined from point to point. The structural connections are defined to the points. The fittings are defined from point to point etc. When defining the points recursively and, after that, by defining all the other entities by using the names of the points, then it is possible to write a recursive program with ease.

Def. 1:  $x=$        $y=$       Layer=

This is usually the first definition of the points. The steel skeleton can be put to the layout model of the whole building by using this definition. The coordinates  $x$  and  $y$  (in mm) can be given recursively from e.g. modulus lines of the building but in the following it is supposed that after this statement the point has real values for  $x$  and  $y$ . The points may form groups that are good to separate from each others. The variable called Layer controls this.

Def. 2: Point= P1= P2= Dx= P3= P4= Dy= Layer=

This defines the local shift from Point. The local shifts  $Dx$  and  $Dy$  can be defined as proportional lengths from the distances between points as seen from the definition.

Def. 3: Point= P1= P2= Dir= P3= P4= L = Layer=

This defines the point by using the direction (in degrees) and length from the other point. The direction and length can be defined by using points as seen from the definition. In this case the variables  $Dir$  and  $L$  are relative values with respect to the angles and the lengths.

The following convention is also used: if the first point (e.g. P3 above) is a **positive** integer and the second (P4 in this case) is zero then the corresponding data ( $Dx$ ,  $Dy$ ,

Dir or L) is taken from the database. The database must be ordered so that the data is in the right place. It is natural to begin with data number 1 and continue as far it is needed. Here it must be mentioned that the initial data needed e.g. for the steel skeleton must be kept as minimum so that the model developed is as general as possible. Note also, that a **negative** integer (for P3) is needed in the following.

This data in the database must be ordered so that it is dependent on the entity that is considered. When defining the steel skeleton the data concerns that. When defining the connection the data concerns the connection which we are defining. The address of this database is needed when transferring the data **back** from the analysing model to the geometrical model. The geometrical model is gressed iteratively from the analysing model as follows. The starting point **must always be** the geometrical model (Heinisuo, 1994). When defining the geometrical model for the first time, all the data needed for the proper definition of the geometry is not known. These are e.g. sizes of welds. These are known only after the analyses of the skeleton is performed. The same situation holds also e.g. for flexible joints. The stiffnesses of these joints are known only when the loadings of these joints are known following EC3 (Eurocode 3, 1993). So, there must be some initial quesses of these kinds of data in the database of the initial values. These data must be updated after the analysing process has been done. Then the proper data from the geometric model is tranferred to the analysing model and the iteration must be done.

These three (or in fact Defs. 1 and 3) definitions are needed and they are often sufficient for the definition of steel skeleton. When using the same system for the definitions of connections of the skeleton more profound definitions are needed. Before considering that, some applications of the definitions 1-3 are given.

#### Example 1.

A simple symmetrical steel frame can be defined by using four initial data values: span, free height, height of the eave and slope of the roof. Also, the global coordinates of

some point (usually the column base) are needed. This data is transferred from the layout model of the building. This data is sufficient for the definition of the geometrical model of the frame. There may be some boundary conditions given by the architect (e.g. the thickness of the column or the layout of the truss) or by the mechanical engineer (e.g. space demands to the ventilation). These boundary conditions must also be transferred from the layout model to the designer of the frame.

The definitions of the most important points of the frame are shown in Fig. 2 in details. The definitions of the points are shown (Def. 8 is also needed in the example, as seen later) in the table form (as they are defined) in Fig. 3. The program that calculates the final coordinates of the points is easy to write and moreover the program is **independent** of the case. I.e., we can use the same program to calculate the coordinates of the points of the skeleton and later the geometry of the connections and so on. When we have the coordinates of a point we can, e.g., draw it. The error control is always present because here it is used the real time graphical presentation of what we are doing. Moreover, if the recursive loop does not converge then the program stops instantaneously and the designer can correct the definitions.

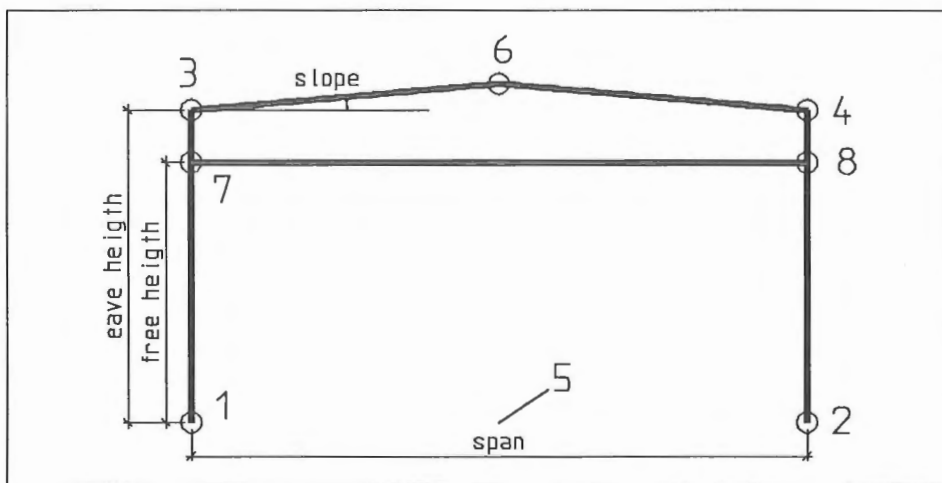


Fig. 2. Definition of a simple frame

Point 1 Def.= 1:

x= 0.000 y= 0.000

Layer= 1

Point 3 Def.= 3:

Point= 1 P1= 0 P2= 0 Dir= 90.000

P1= 2 P2= 0 L= 0.000

Layer= 1

Point 5 Def.= 3:

Point= 1 P1= 1 P2= 2 Dir= 0.000

P1= 1 P2= 2 L= 0.500

Layer= 2 (not shown in the figure)

Point 7 Def.= 3:

Point= 1 P1= 1 P2= 3 Dir= 0.000

P1= 4 P2= 0 L= 0.000

Layer= 1

DATA1: SPAN = 12000.000

DATA2: EAVE\_HEIGHT = 6000.000

DATA3: SLOPE = 5.000

DATA4: FREE\_HEIGHT = 5000.000

Point 2 Def.= 3:

Point= 1 P1= 0 P2= 0 Dir= 0.000

P1= 1 P2= 0 L= 0.000

Layer= 1

Point 4 Def.= 8:

Point= 3 P1= 1 P2= 2 Dir= 0.000

Point= 2 P1= 1 P2= 3 Dir= 0.000

Layer= 1

Point 6 Def.= 8:

Point= 3 P1= 3 P2= 0 Dir= 0.000

Point= 5 P1= 1 P2= 3 Dir= 0.000

Layer= 1

Point 8 Def.= 8:

Point= 7 P1= 1 P2= 3 Dir= 0.000

Point= 2 P1= 1 P2= 3 Dir= 0.000

Layer= 1

Fig. 3. Definitions of the points of the frame

It can be seen that there can be more (fixed) variables than the four initial values mentioned above. In the example it was assumed, that the columns are located vertically, the baseline is horizontal and the middle point of roof is in the middle of the span. It can also be seen from the example, that the definitions are not unique but, instead, they can be done in different ways. By changing the initial data or fixed values in the definitions we can get the different layouts for the same frame. Fig. 4 presents some variations of the frame.

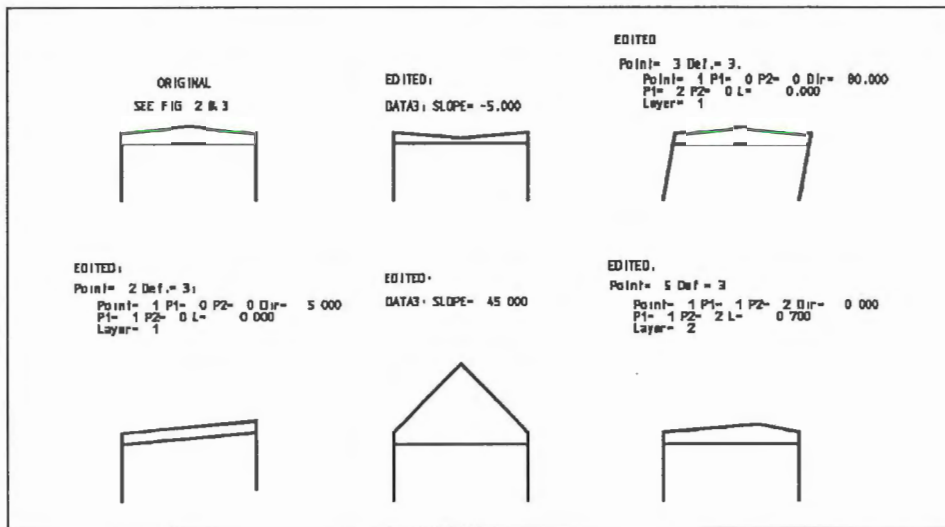


Fig. 4. Variations of the simple frame

Fig. 5. presents some simple examples of the steel skeletons and their initial data. Some bars are also defined to the models in order to give clearer graphical presentation.

As stated above, more information is needed when defining the connections of the steel skeletons. Firstly, the definition of the first points (e.g. P1 above) can be a negative integer and the integer part refers to the number of the bar joining in the connection. Therefore, the order of the bars must be defined in the connection. Moreover, the second integer (P2 in this case) refers to the entity of the bar in the database. So, the bars must be defined in the library by using some system. Here it is used the library where it is defined the measures of the bars by using the following system:

Hot rolled I profiles: height, width, thickness of the flanges, thickness of web, rounding radius of the corners

Hot rolled or cold formed rectangular tubes: height, width, thickness of wall, rounding radius of the corner





As an example the definition  $P1 = -2$   $P2 = 3$  means that the thickness of the flange (or the thickness of wall) of the second bar of the connection is used as the numerical value for  $Dx$  or  $Dir$  if the definition is 2 or 3.

The library can be enlarged correspondingly. The library must also include some numbering of the profiles and the corresponding standards. The library must have all the necessary data needed for the geometrical and analysing models and only that. All the other properties needed in the CAD-program (inertia moments, sectorial coordinates, etc) must be possible to calculate by using the data in the library. On the other hand these values can be calculated if needed. They need not to be stored to the database. The most important thing is that the user of the library knows where the data is so that he can refer to the right position. Secondly, some more definitions are needed to make the models easier to handle than without them. In the following, the definitions used are firstly presented and after that there are given some motivations to these definitions. (Note that Defs. 4-5 are not in use.)

Def. 6: Point=  $P1=$   $P2=$   $Dir=$   $P3=$   $P4=$   $Dx=$   $Layer=$

Def. 7: Point=  $P1=$   $P2=$   $Dir=$   $P3=$   $P4=$   $Dy=$   $Layer=$

Def. 8: Point=  $P1=$   $P2=$   $Dir=$   $Point=$   $P3=$   $P4=$   $Dir=$   $Layer=$

Def. 9:  $P1=$   $P2=$   $P3=$   $P4=$   $P5=$   $Layer=$  (min proj)

Def.10:  $P1=$   $P2=$   $P3=$   $P4=$   $P5=$   $Layer=$  (max proj)

Def.11:  $P1=$   $P2=$   $P3=$   $P4=$   $P5=$   $Layer=$  (min sin)

Def.12:  $P1=$   $P2=$   $P3=$   $P4=$   $P5=$   $Layer=$  (max sin)

Def.13:  $Start=$   $End=$   $P1=$   $P2=$   $Dif=$   $Estart=$   $DirP=$   $Am=$   $Layer=$

Def.14: Point=  $P1=$   $P2=$   $Dir=$   $x=$   $Layer=$

The definitions 6-8 are explained in Fig. 6.

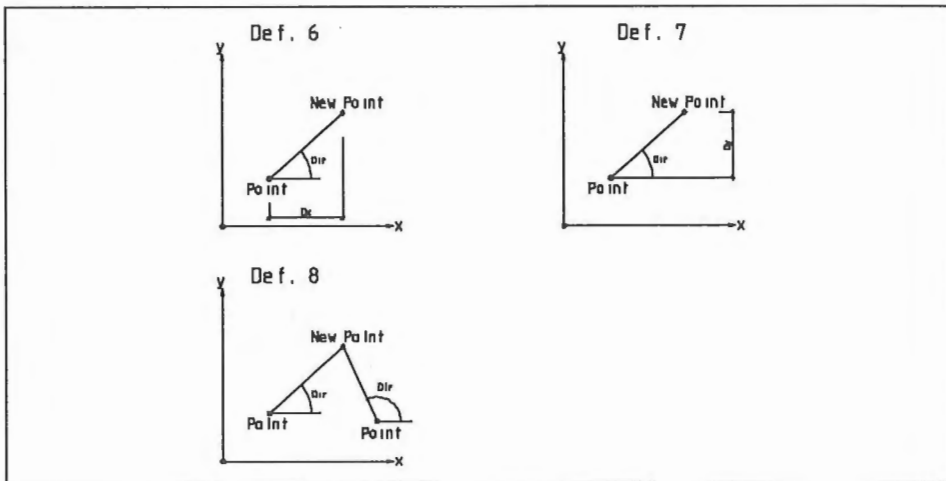


Fig. 6. Definitions 6-8

The definition 9 means that the point is chosen from the group  $\{P3, P4, P5\}$  so that the projection of the point from the line  $P1-P2$  is minimum. The definition 10 means that the corresponding maximum is chosen (see Fig. 7). Defs. 11 and 12 define the point so that the minimum (or maximum) sinus of the length from the line  $P1-P2$  is the criteria (see Fig. 7).

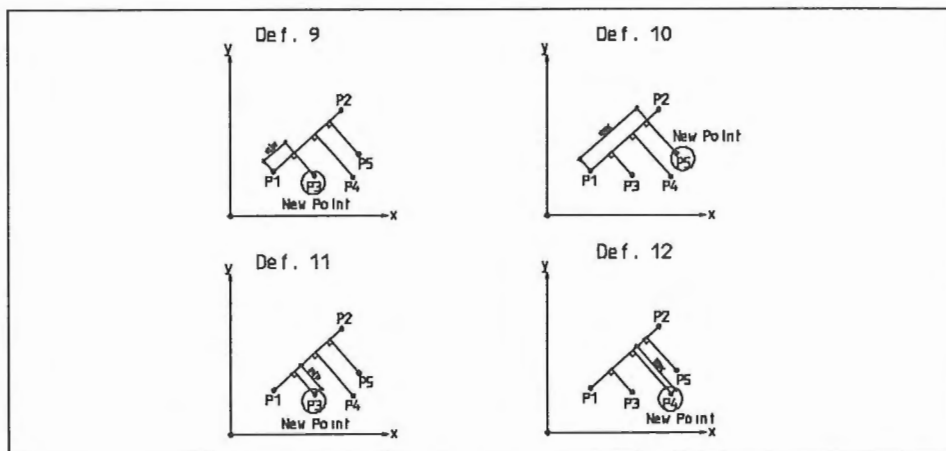


Fig. 7. Definitions 9-12

These definitions are needed e.g. when defining the connection of the skeleton shown in Fig. 8. It is good if the connections shown in Fig. 8 can be defined as a same connection so that the amount of connections can be kept as minimum.

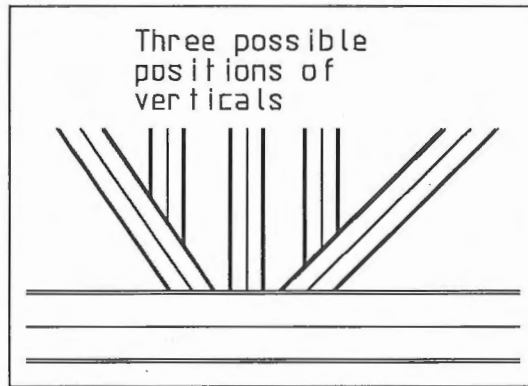


Fig. 8. Connection, which can be defined by using Def. 9

The definition 13 is needed when it is checked whether e.g. bolts can be put to the line. The points are located at the line between Start and End tributary Dif starting from Start. If there is not enough space for these points then the rest are put to the line Estart-DirP starting from Estart. A variable  $A_m$  is the amount of points (see Fig 9). The definition 14 means that the point is located to the given direction from the reference point with the given x coordinate. This is needed when defining e.g. verticals of the truss to the specific positions.

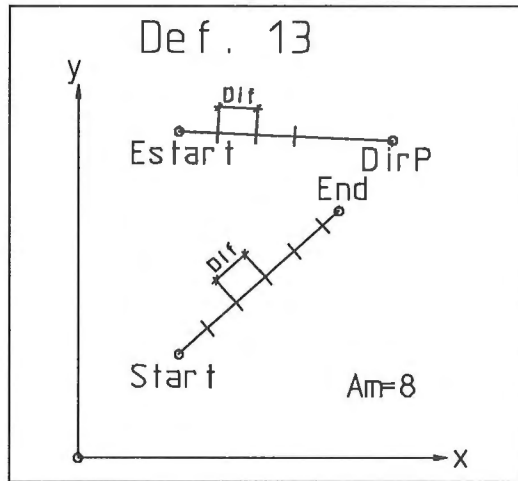
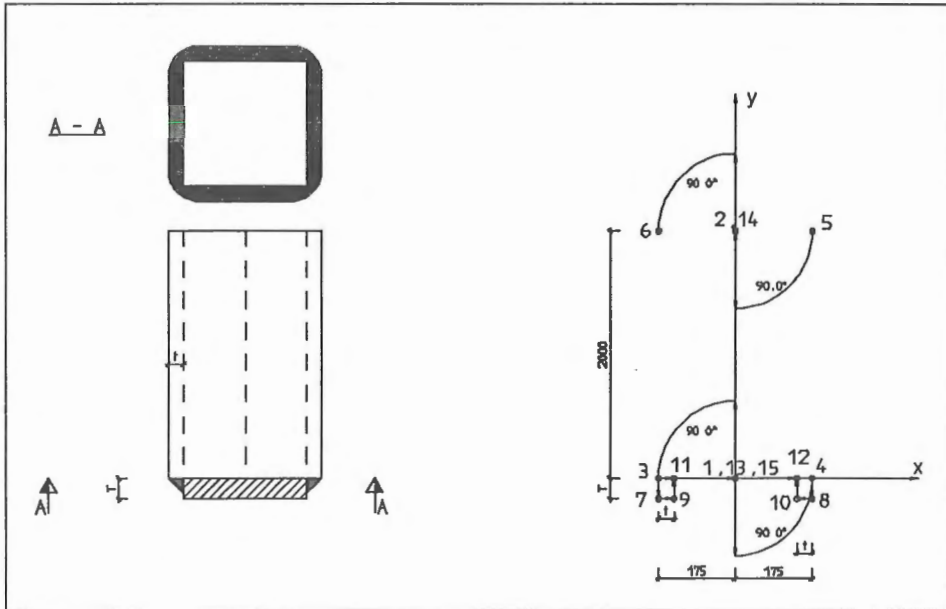


Fig. 9. Definition 13

Example 2.

Fig. 10 presents a connection of the steel skeleton. The initial data of the connection is the thickness of the end plate (fitting). The end plate is rectangular and its size is defined recursively in the connection. The weld is assumed to be a full filled weld ( $a = \text{minimum} \{T, t\} / \text{SQRT}(2)$ ). Note, that due to the calculation of the welding volume there must be different connections e.g. to I-bar and RHS-bar. Fig. 10 presents the definitions of the connection with comments. The analysing model of the joint is very simple and it is the line from the point 13 to the point 14 (when using one dimensional finite elements). It must be noted, that the absolute coordinates given in Fig. 10 are local and they are arbitrary. The joint is connected to the final model recursively so that the same connection can be used in many places in the same project and the definition can be stored for the further use in the projects in the future. The absolute values of the coordinates in Fig. 10 are replaced in fact recursively by the true values. The definitions of the points must include all the data needed when defining later the cuttings of the bars, the geometry of the fittings, the welds, the bolts, the analysing model of the connection, the measures and other data needed in different drawings etc (see Fig. 11 later).



Point 1 Def.= 1:

x= 0.000 y= 0.000

Layer= 1

Point 3 Def.= 3:

Point= 1 P1= 1 P2= 2 Dir= 90.000

P1= 0 P2= 0 L= 175.000

Layer= 1

Point 5 Def.= 3:

Point= 2 P1= 2 P2= 1 Dir= 90.000

P1= 0 P2= 0 L= 175.000

Layer= 1

Point 7 Def.= 3:

Point= 3 P1= 6 P2= 3 Dir= 0.000

P1= 1 P2= 0 L= 0.000

Layer= 1

Point 2 Def.= 1:

x= 0.000 y= 2000.000

Layer= 1

Point 4 Def.= 3:

Point= 1 P1= 2 P2= 1 Dir=90.000

P1= 0 P2= 0 L= 175.000

Layer= 1

Point 6 Def.= 3:

Point= 2 P1= 1 P2= 2 Dir=90.000

P1= 0 P2= 0 L= 175.000

Layer= 1

Point 8 Def.= 3:

Point= 4 P1= 6 P2= 3 Dir= 0.000

P1= 1 P2= 0 L= 0.000

Layer= 1

Point 9 Def.= 3:

Point= 7 P1= 7 P2= 8 Dir= 0.000

P1= -1 P2= 3 L= 0.000

Layer= 1

Point 11 Def.= 8:

Point= 9 P1= 1 P2= 2 Dir= 0.000

Point= 3 P1= 3 P2= 4 Dir= 0.000

Layer= 1

Point 13 Def.= 3:

Point= 3 P1= 3 P2= 4 Dir= 0.000

P1= 3 P2= 4 L= 0.500

Layer= 1

Point 15 Def.= 3:

Point= 13 P1= 4 P2= 3 Dir= 0.000

P1= 0 P2= 0 L= 0.000

Layer= 1

DATA1: THICKNESS = 40.000

Point 10 Def.= 3:

Point= 8 P1= 8 P2= 7 Dir= 0.000

P1= -1 P2= 3 L= 0.000

Layer= 1

Point 12 Def.= 8:

Point= 10 P1= 1 P2= 2 Dir=0.000

Point= 3 P1= 3 P2= 4 Dir= 0.000

Layer= 1

Point 14 Def.= 3:

Point= 13 P1= 1 P2= 2 Dir=0.000

P1= 1 P2= 2 L= 1.000

Layer= 1

Fig. 10. Example of the connection

## DISCUSSION

The definitions above can be used to define the layout of the steel skeleton. In the modelling process there are needed also the definitions of the bars between the points and definitions of fittings, bolts, welds, different drawing controls etc. These are beyond the scope of this paper. The propositions for product models of entities called bars and fittings are given in Ref. (Heinisuo, 1994). Some examples for the other aspects are given in Ref. (Heinisuo et al., 1991). The model is easy to create when the definitions are used in the format in which they are defined above. Also the changing of the definition is easy by editing the definition. Once the model is made the changing of the initial data is easy by editing the database.

When using the derived system for the modelling of the primitive entities (bars, fittings, welds) of the steel skeleton they can be defined by the system. The fourth entity, bolts, can be modelled by using only the initial data. So, it is possible to give the diameter of the bolts. Here, it is assumed that the thickness of the plates joined by the bolt and the direction of the bolt is given as an input from the model developed to the library of bolts. The library gives the proper size (length, washers, nuts) of the bolt by using this data (diameter, direction, thickness of plates to be joined and of course the strength and standards of the bolt). The same holds also for the anchor bolts. There is a danger that the computer chooses different bolt to every connection. It would be possible to define the library of bolts as the library of profiles above was defined and used, but it was not done in this study. The final sizes of the primitive entities are defined after the iteration between the geometrical and analyzing models has been done.

The system described in this study can be used to define the analysing models of steel skeletons in a compatible manner. The finite element models of the connections are defined locally and recursively in the connections and they are combined to the finite elements of the bars. The connections may include 1, 2 or 3 dimensional finite elements and their geometrical properties are defined by using the system developed. Other data needed for the analysis of the connections and the whole skeleton must be transferred by using the database of the skeleton. These other data are loadings and support conditions. So far, only one dimensional finite elements are used when modelling the connections. In the future more complicated models are planned to be used. Fig. 11 presents some examples of the geometrical and analysing models of the steel skeleton and its details. E.g. the location of the bolt text (2M24\*110) and other data needed in the different drawings are defined by using the system described above.



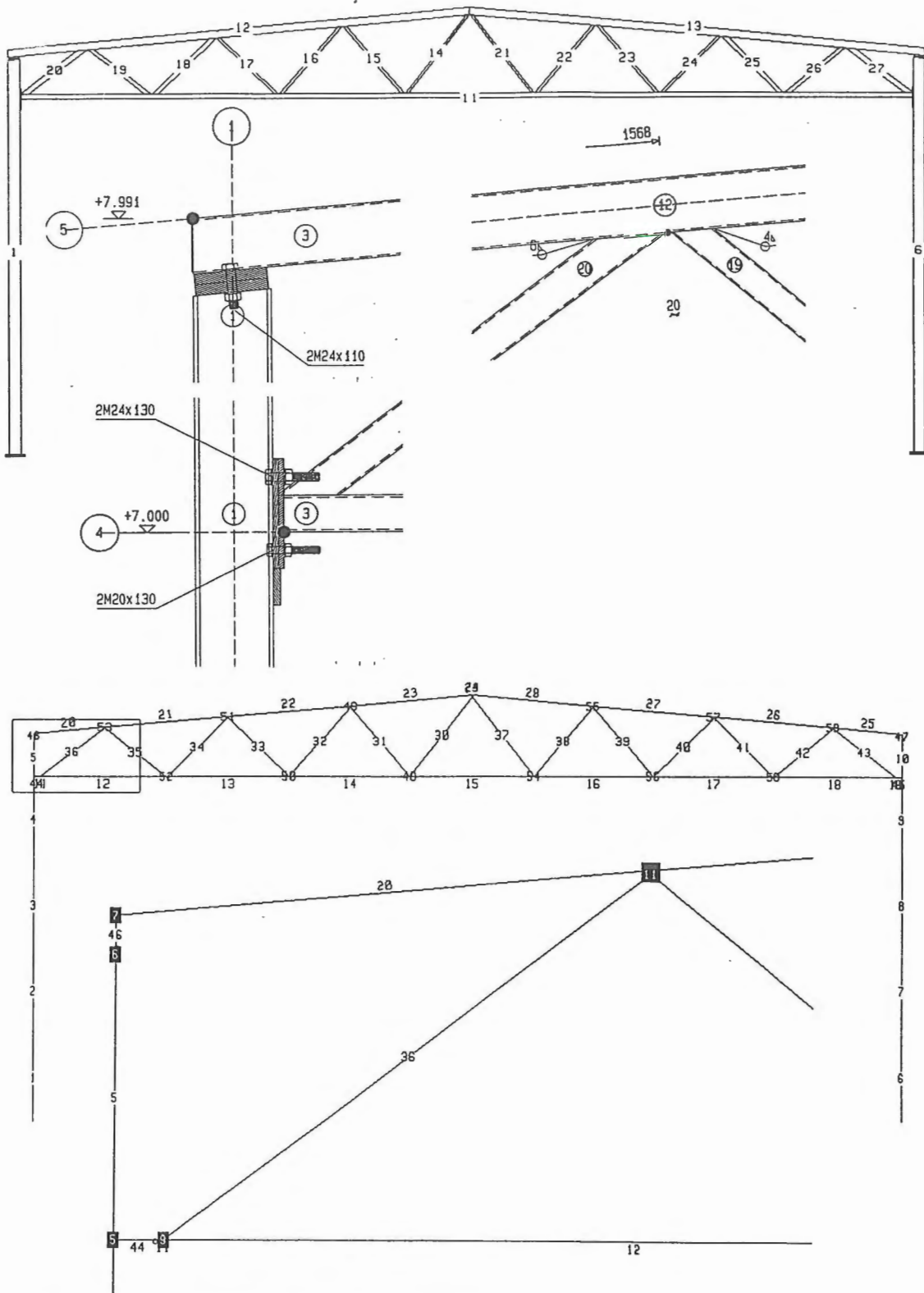


Fig. 11. Geometrical and analysing models of steel skeleton

The system developed is especially suitable for the recursive definitions of the connections of steel skeletons. It is believed, that we are as near as possible to the situation where there is only one "universal connection" in the steel skeleton, which is the ideal case. The definitions of the different connections are case dependent but the code to generate the geometrical and analyzing models remains valid for all the connections. The amount of the code does not increase when the amount of the connections is increasing.

The system is used for the modelling of steel skeletons. It is believed, that the system can be applied to other structures (concrete, wood, composites), too. The similar system can be developed to the modelling of three dimensional structures.

#### **ACKNOWLEDGEMENTS**

The engineers (especially Mr. Annila) of the KPM-Engineering Ltd are acknowledged of the co-operation when implementing the system presented. Finnish Academy is gratefully acknowledged from the financial support to the research project.

#### **REFERENCES**

AutoCAD Versio 11 Suomen kielinen käsikirja, Julkaisu AC11RM, Göteborg, 17. elokuuta 1990, Copyright 1982,83,84,85,86,87,88,89,90 Autodesk, Inc.; Autodesk AB (in Finnish)

Bento J., Feijo B., Lloyd Smith D., Engineering design knowledge representation based on logic and objects, Applications of Artificial Intelligence in Structural Engineering, Ed. Ian F. C. Smith, Proceedings of the 1st Workshop of the European Group for Structural Engineering Applications of Artificial Intelligence (EG-SEA-AI) held at EPFL, Lausanne Switzerland, March 21-11, 1994, pp. 167-181

Eurocode 3, Design of Steel Structures, DD ENV 1993-1-1:1992

Heinisuo M., Expert systems for design of steel structures, Part 1: Introductory report, Tampere University of Technology, Department of Civil Engineering, Structural Mechanics, Report 14, Tampere, 1994

Heinisuo M., Möttönen A., Paloniemi T., Nevalainen P., Automatic design of steel frames in a CAD-system, Proceedings of the 4th Finnish Mechanics Days, Ed. Niemi E., Research Papers 17, Lappeenranta University of Technology, Lappeenranta, 1991, pp. 197-204

Torp-Pedersen U., Barlach C., Simonsen E., Development of application systems under the Eureka 130 CIMSTEEL project, Proceedings of the Nordic Steel Colloquium, Denmark, 1991, pp. 501-511

TSP, Teräsrakenteiden suunnittelun uudet pelisäännöt, Teräsrakenneyhdistys r.y., 1989 (in Finnish)

Markku Heinisuo, Dr.Tech., Senior Research Fellow, The Academy of Finland, c/o Tampere University of Technology, Department of Civil Engineering, Lab. of Structural Mechanics