Report

An innovative tool for teaching structural analysis and design

Author(s):
Pedron, Claudia

Publication Date:
2006

Permanent Link:
https://doi.org/10.3929/ethz-a-005299673

Rights / License:
In Copyright - Non-Commercial Use Permitted
An Innovative Tool for Teaching Structural Analysis and Design

CLAUDIA PEDRON

dott. ing., Civil Engineering, Trento, Italy

INSTITUTE OF STRUCTURAL ENGINEERING
SUISSE FEDERAL INSTITUTE OF TECHNOLOGY

Zurich, October 2006
Foreword

By the beginning of the 19\textsuperscript{th} century many fundamental relations of structural engineering (equilibrium, elasticity, Euler’s stability, etc.) had already been discovered. In the following one and a half centuries structural engineers were then left with the problem of numerically solving well known differential equations which, however, for most real-life problems proved to be unsolvable. Nevertheless, as more and more complex structures had to be built, great efforts were made in the development of either simplifying assumptions (e.g. by Karl Culmann) or cleverly organized hand-computations (e.g. by Hardy Cross) or (apparently) sophisticated analytical or even graphical solution methods.

Looking at these efforts from today’s perspective, it appears that they might well have helped building technology’s spectacular progress in those days (the most important was the invention of reinforced concrete). However, when compared to the developments of the last few decades, they appear today to be of very little relevance indeed, i.e. mainly of historical interest. As for so many professions, also in the field of structural analysis, the computer revolution has been a profound one, both in theory and practice.

In fact, the advent of programmable computers led to fundamental changes in the way structural engineers work. Today, like it or not, structural analysis, whenever needed, is done by computer programs. This means that structural engineers need to cooperate with distant program developers who are experts in computational mechanics but have no knowledge of the specific problems the tools they develop help to solve. Historically, this form of cooperation is new. In order to be possible and fruitful structural engineers must be prepared for it.

Here the question of how to teach structural analysis comes into play. As in the past, students must first understand the fundamentals of structural mechanics laid down centuries ago. But then, and with highest priority, the above mentioned cooperation problem has to be tackled. This means that students must get some basic ideas on how structural analysis programs work internally (which is not too difficult). They must then learn how to model their structures in a computer-friendly way and understand the meaning of the results the computer provides. This is not only what structural engineers today need to know, but also quite helpful for a general understanding of how loaded structures mechanically behave, which is essential for sound structural design.

As many background problems (like the question of how to solve large equation systems) only concern program developers and if, at first, only simple linear structures are considered, these teaching goals can be reached in a relatively short time, e.g. in a two semester introductory course. It is our firm conviction that this teaching material, which today must be considered as basic and fundamental for both practice and research, definitely belongs to the civil engineering curriculum at undergraduate level.

Based on these considerations and on recent developments of e-learning technology, miss Pedron was asked to develop suitable tools for helping to teach structural analysis and design at undergraduate level to both civil engineering and architecture students. This dissertation describes her efforts to reach this goal.

October 2006, Edoardo Anderheggen
# Contents

1 Introduction  
   1.1 Scope ................................................................. 1  
   1.2 Contents ............................................................ 2  

2 Structural Analysis: A Brief History  
   2.1 Preparation Time (1575-1825) ..................................... 7  
   2.2 Discipline Creation Period (1825-1900) ......................... 9  
   2.3 Consolidation Period (1900-1950) ............................... 10  
   2.4 Integration Period (1950-1975) .................................. 12  
   2.5 Diffusion Period (1975 until now) ............................... 12  
   2.6 Information and Communication Technology (ICT) in the Civil Engineering and Architecture Today ................................. 13  
   2.7 Creativity in Engineering Education ............................. 14  

3 Teaching the Fundamentals of Structural Analysis and Design  
   3.1 Traditional Teaching ............................................... 15  
   3.2 Alternative Approaches .......................................... 17  
   3.2.1 “Hands on” Experiments ...................................... 17  
   3.2.2 Graphic Methods .............................................. 19  
   3.2.3 Computer Simulations ........................................ 20  
   3.3 Didactic Theories .................................................. 21  
   3.3.1 Teacher-Centered Instruction and Behaviourism .............. 22  
   3.3.2 Student-Centered Instruction and Constructivism ............ 22  
   3.4 Use of Technology in Science Education .......................... 26
3.4.1 Use of Technology to Deliver Information .......................... 27
3.4.2 Use of Technology to Build Competencies ....................... 28
3.5 Learning with EasyStatics ........................................... 30
   3.5.1 User Friendliness and Customization .......................... 31
   3.5.2 Exploration, Observation, Reflection and Negotiation ....... 32
   3.5.3 Situated and Problem Based Learning .......................... 40
   3.5.4 Demonstrations, Exercises and Collaborative Learning ....... 40
3.6 Programming Challenges ............................................. 41

4 EasyStatics in the context of E-Learning ............................. 45
   4.1 The Value of E-Learning ........................................... 46
   4.2 E-Learning Platforms ............................................. 46
      4.2.1 Characteristic Attributes of E-Learning Platforms ......... 47
      4.2.2 Easy to Use and Flexible Platforms ....................... 49
      4.2.3 Choosing the Most Adequate E-Learning Platform ......... 50
      4.2.4 The Lifespan of E-Learning Platforms ..................... 51
      4.2.5 Basic Components of an E-Learning Platform .............. 51
   4.3 EasyStatics E-Learning Platform .................................. 53
      4.3.1 Files Managed by the EasyStatics E-Learning Platform .... 55
      4.3.2 Users and Users’ Rights ..................................... 55
      4.3.3 User Interface ............................................... 56
      4.3.4 Welcome Page for Registered Users ........................ 57
      4.3.5 Teacher Environment ....................................... 58
      4.3.6 Student Environment ....................................... 59
      4.3.7 Database Structure ......................................... 59
      4.3.8 Course Preparation ......................................... 61
      4.3.9 Experience Made ........................................... 62

5 EasyStatics as a Tool for Teaching ................................... 63
   5.1 Basics of The Finite Element Method ............................. 64
   5.2 Static Case - 1st Order Theory .................................... 65
5.2.1 Global and Local System ........................................ 65
5.2.2 Global System: Assembling and Solution .................. 66
5.2.3 Element Results ................................................. 67
5.2.4 Exercises ....................................................... 67
5.3 2nd Order Theory - Stability .................................. 79
5.3.1 Normal Force Influence on Bending Behaviour .......... 79
5.3.2 Element Geometric Stiffness Matrices ...................... 79
5.3.3 Structural Analysis by 2nd Order Theory .................. 80
5.3.4 Linear Stability Analysis .................................... 80
5.3.5 Further Considerations ...................................... 81
5.3.6 Exercises ....................................................... 81
5.4 Dynamic Eigenmodes ............................................ 87
5.4.1 Inertia Forces and Mass Matrices ........................... 87
5.4.2 Eigenfrequencies and Eigenmodes .......................... 88
5.4.3 Exercises ....................................................... 88
5.5 Rigid-plastic Analysis .......................................... 94
5.5.1 Plane Truss and Frame Models ......................... 94
5.5.2 Limit Analysis by the Static Theorem ................... 95
5.5.3 Limit Analysis by the Kinematic Theorem .............. 96
5.5.4 Advantages and Drawbacks of Plasticity Theory ......... 97
5.5.5 Exercises ....................................................... 97
5.6 Exercises on Section Dimensioning ......................... 105

6 Academic Experience with EasyStatics ....................... 111
6.1 Computer Statics Course ...................................... 111
6.2 Introductory Course ............................................ 112
6.3 EasyStatics Assessment ....................................... 115

7 Conclusions and Future Developments ....................... 117
7.1 Conclusions ..................................................... 117
7.2 Future Developments ........................................ 118
A Appendix: *EasyStatics* Functionality

A.1 Model Building ................................................. 121
A.2 Moving Objects ................................................. 123
A.3 Analysis Types .................................................. 124
A.4 Results ......................................................... 124
A.5 Stress Fields and Truss Model ................................. 125
A.6 Global and Local Instability ................................. 126
A.7 Plane Frame or Plane Truss ................................. 127
A.8 Section Dimensioning ........................................... 127
A.9 Check Line ................................................... 131
A.10 Customizing the Graphical User Interface .................. 131
A.11 Show and Freeze Numerical Values .......................... 132
A.12 Sketching the Model on a Background Image ............... 132
A.13 Help Functions ............................................... 133
A.14 Copies of a Model ............................................ 133
A.15 Compare Function ............................................. 133
A.16 Internal Editor ............................................... 134
A.17 Computational States ......................................... 136
A.18 Exporting Pictures ........................................... 136
A.19 Internal HTML Editor ......................................... 136
A.20 Connection with the *EasyStatics* Database Server ....... 137
Chapter 1

Introduction

1.1 Scope

We address here the question of how today’s Information and Communication Technology (ICT) can help teaching and learning structural design at the undergraduate level. For students in architecture and civil engineering this is a core subject with a centuries-old tradition.

While innovative ICT-tools are already applied in a number of traditional subjects, structural design should be considered as a special case. In fact, already in the 1960s the computer completely revolutionized structural analysis both in its theoretical basis (Computer Shapes Theory) and its practical applications.

In our opinion, this is not always adequately taken into account when teaching this old discipline to students in architecture and civil engineering. As before, they need to understand how the structures they design are expected to carry loads, but now they also have to know how to use the software tools they will encounter in practice. In fact, undergraduate students should first of all learn how to properly model real-life structures for the computer to provide meaningful results. They then have to learn how to extract from the computed results the information which is relevant to structural design. Both are difficult tasks. However, the theoretical intricacies of structural analysis, i.e. today mainly of the Finite Element Method (FEM), can be addressed at the graduate level or, at least for architects, not at all. The ability to manually perform structural analysis calculations is never needed.

Computer programs, if specifically and properly designed for this purpose, can become a useful tool for developing a “feel” for structural design. Two questions arise: What should students learn when first confronted with structural design? How can they be helped by tailor-made software? To answer these questions a simple but not trivial FE-program called EasyStatics has been developed that is intended to become a teaching and learning tool for undergraduate courses. It has been designed as a “virtual structural laboratory”, as easy to use as paper and pencil, as intuitive as a hand calculator and as captivating as a video game, so as to provide its users with the motivation needed for efficient learning.
Chapter 1. Introduction

The EasyStatics program is embedded in an Internet-based platform, from and to which students and teachers can down- and upload executable EasyStatics files as well as all information which is relevant for a specific course.

1.2 Contents

In Chapter 2 some highlights of the history of structural analysis from the Renaissance period until today are briefly reviewed. It is shown how the discovery of the Finite Element Method, the advent of the Personal Computer and, later, of today’s Information and Communication Technology have completely revolutionized the way civil engineers work. In fact, nowadays, they no longer analyze structures manually, but need to become familiar with new technology, i.e. they have to learn how to use FE-programs properly, correctly interpret computer results, use networking, databases, etc. It follows, that in order to train them adequately for their future profession, teachers should review their programs in terms of both the content and the way of providing knowledge. This can be done by properly exploiting the possibilities offered by the new media.

In Chapter 3, firstly the traditional way of teaching structural analysis and design to architecture and civil engineering students is briefly sketched. Although the teaching has to be focused on different goals, both need to develop a good understanding of structural behaviour. But the abstract mathematical models adopted in traditionally oriented structural analysis and design courses sometimes appear to be inadequate to transfer a sufficient understanding of basic structural principles. Facing this problem, some teachers use alternative methods, which include, for example, “hands on” experiments, graphical methods and computer simulation programs.

Two general didactic methods are then considered, which are based on the so-called “behaviourist” and “constructivist” approaches. The first one corresponds to the traditional way of teaching, according to which knowledge is transmitted from the teacher, the sole authority in the class, to the learner. The limit of this approach is that often theoretical knowledge is passed on with the students having only a passive role. Alternatively, according to constructivism, knowledge is created or rediscovered by the students, which is an active process.

A teacher-centered (behaviourist) or a student-centered (constructivist) instruction determines the way ICT is to be used in education. In the first case, technology is seen as a medium to deliver information which is considered to be stored in the computer. In the second case, it is considered as a vehicle helping in the process of constructing knowledge. Students are supposed to learn with and not from technology.

The last section of chapter 3 is dedicated to the computer program EasyStatics, a “virtual laboratory” allowing students to create simple plane frame and truss structures like plane frames and trusses with no predefined geometry, under arbitrary load and support conditions and with members of different sizes and materials, whereby after
any model change the results are recomputed and immediately shown. Students can therefore improve their understanding by observing how parameters affect structural behaviour. While, in a quasi game-like way, they interactively manipulate the model, they compare different structural situations and understand how structures behave and why one design alternative appears to be better than another. All this is possible through several functionalities implemented in the program, which was designed exclusively for teaching purposes. The constructivist use of its functionalities is what is expected to make EasyStatics a valuable tool for teaching and learning structural analysis and design at the undergraduate level.

In Chapter 4 it is discussed how the newly developed e-learning platform works internally and how it can be used by teachers to prepare a structural design course and by students to solve and document exercises.

Some theoretical background for the different analysis types provided by EasyStatics is found in Chapter 5 with possible demonstrations and exercises being suggested.

In Chapter 6 the academic experience we were able to gather so far in different schools and universities is reported together with teachers’ and students’ comments on it.

In the final chapter a general assessment of the current state of the EasyStatics project is given as well as some hints on possible future developments.

The design and the development of the complex EasyStatics program is to be considered an essential component of this doctoral work. In the appendix the way EasyStatics actually interacts with the user is therefore presented, with special emphasis on all the features specially designed for achieving the above mentioned goal of the constructivist approach.
Chapter 2

Structural Analysis: A Brief History

Structural analysis provides a set of scientific rules to more or less accurately model a structure and then to proportion each of its parts so that loads can be safely carried. This includes the theory of structures, whose subjects are models and numerical methods, and the strength of materials relating to the dimensioning of structural members [1].

The use of scientific rules to design structures is relatively recent. The Colosseum and the Pantheon in Rome, the Greek temples, the Aya Sofia in Istanbul, the gothic cathedrals and all ancient buildings which survive to this day, were built not only without any computation but also without any theory as we know today.

Rules of proportion developed through experience and practical training were used to design and build such structures. The medieval 'master' knew how to handle the material as well as how to give the building an 'architectural' design. It is only after the Renaissance that the aesthetic and structural aspects began to diverge and two distinct professional figures emerged: the architect and the engineer. The first concentrated on the rules of proportions and on esthetics while the second explored the scientific rules embedded in the practice of building.

Starting from the Renaissance, K.E. Kurrer [2] divides the history of structural analysis into five periods: the Preparation Time from 1575 until 1825, the Discipline Creation Period from 1825 until 1900, the Consolidation Period from 1900 until 1950, the Integration Period from 1950 until 1975 and the Diffusion Period from 1975 until now.
Figure 2.1: Colosseum in Rome.

Figure 2.2: AyaSophia in Instambul.
2.1 Preparation Time (1575-1825)

In his Discorsi (1638) Galileo introduced the concept of strength of materials.

Through experiments on wooden specimens he determined the absolute strength in tension and subsequently investigated the strength of a cantilever beam embedded in a masonry wall under self weight, observing its behaviour with an increasing load applied at its free end. Although the calculated bending strength was incorrect (due to the assumption of a linear stress distribution in the section at the fixed end of the beam) Galileo reached the correct conclusion that the strength of a rectangular beam is proportional to the width and to the square of the height of its section. Through his work Galileo made a significant contribution to the bending problem.

Many researchers investigated the question of the strength of the material, starting from the hypothesis of Galileo.

While Galileo was mainly concerned with the strength of the material, Robert Hook (1629-1695) in England developed the theory of elasticity.

In the same period Simon Stevin (1548-1620) dealt with the problem of the decomposition of the forces.

In these years the predominance of geometry and the separate development of statics, strength of the material and elasticity theories did not allow for a proper analysis of structural elements. In the 18th century infinitesimal calculus found application in astronomy, theoretical mechanics, geodesy and civil engineering. Mathematicians like Leibniz (1646-1716), Daniel and Jacob Bernoulli and Leonard Euler (1707-1783) made further progress in the theory of beams and their elastic line. In 1791 Jacob Bernoulli discovered that the curvature of a beam in pure bending is proportional to the value of the bending moment.
Figure 2.4: Galileo’s cantilever beam embedded in a masonry wall.

Figure 2.5: Hooke’s spring.
Daniel Bernoulli, 50 years later, simplified considerably the mathematics of his brother’s theorem, giving it an “engineering” approach. In the first half of the 18th century the first engineering schools based on the application of the infinitesimal calculus to technical objects arose in France. In 1729 Bernard Forest de Bélidor published the first standard Code of Practice for Civil Engineering. In this book the findings of science are made into rules of design where mathematics found a practical application.

Fifty years later, Charles Augustin de Coulomb (1736-1806) in his work Mémoire analyzed the earth thrust, the arch and the beam through infinitesimal calculus in a relatively clear way which made his work more readable and more successful than other previous publications. In this book structural analysis exhibits for the first time the characteristics of a scientific subject.

2.2 Discipline Creation Period (1825-1900)

In the Discipline Creation Period the isolated discoveries of the Preparation Time were integrated with the elasticity theories developed in France in the first half of the 19th century.

Henry Navier (1785-1836), professor at the Ecole des Ponts et Chaussé, was also concerned with the strength of materials, analyzing numerous wood and steel constructions.

His idea was that the ultimate interest of an engineer is to ensure the safety of a structure under specific loads, not to investigate the behaviour of a structure near collapse. To this end, an engineer must calculate the stresses in a loaded structure, so as to ensure that they are below the elastic limit of the materials. While equilibrium equations are sufficient to determine internal and external forces in simple structures, the statically determinate or isostatic ones, they cannot be used alone to analyze hyperstatic structures. This problem, which soon became an important issue in the field of the theory of structures, was studied by Navier, who developed a theory to handle the second type of structure. All the scientific apparatus was available for this analysis: Jacob Bernoulli had stated that the elastic curvature of the beam was proportional to the value of the bending moment at each point, and Daniel Bernoulli had shown how to set up the differential equation of elastic bending to determine the deflected shape of a beam. Navier used the elastic equation combined with the equilibrium equation to analyze a cantilever beam supported by a rigid prop at its end, which is a hyperstatic structure and then, generalizing this simple case, he developed the linear elastic theory, treated in his book Résumé des Lecons (1826). With this work Navier indicated a scientific way to model and numerically analyze a structure using simple calculation tools and material properties. Following the methods dealt with in Navier’s book, an engineer could, in principle, optimize its model and build cost-effective structures capable of safely carrying the applied loads.

Navier was also concerned with hyperstatic trusses, which were of interest in the 19th century due to the construction of steel bridges. Although his method for solving trusses was correct, the resulting calculations were prohibitively complicated due to the large number of equations to be solved.
In the second half of the 19th century, methods were devised to reduce the size of the problem, among them some based on graphics. The method’s most important exponent was the Swiss professor Karl Culmann (1821-1881), who developed the truss theory and refined graphical analysis methods. This theory was essential for the construction of steel truss bridges in the second half of the 19th century.

Culmann’s method was useful for isostatic trusses but less appropriate for hyperstatic ones. In the late 19th century Maxwell, Castigliano, Otto Mohr and Müller-Breslau further developed the linear elastic theory of beam structures with the aim of simplifying the complex mathematical calculation needed for hyperstatic structures. From Maxwell’s work comes the reciprocal theorem, which avoided some of the complex mathematical calculations in the solution of hyperstatic trusses. Castigliano formulated the theorem which bears his name. Although the analysis could be simplified using Castigliano’s method, the number of equations to be solved remained large. The force method developed by Müller-Breslau (1851-1925) based on the principle of virtual forces allowed one to reduce the number of equations to be solved for hyperstatic structures.

2.3 Consolidation Period (1900-1950)

This period was characterised by the spread of reinforced concrete. Its invention in Germany in the second half of the 18th century represented a revolution in the field of construction and at the same time had an impact on structural analysis, leading to the development of new theories. Since 1915 the theory of frame structures and some 10 years later, the theory of two dimensional structures like plates, shells and folded structures
Chapter 2. Structural Analysis: A Brief History

Figure 2.7: Heinz Isler: Shell model of office building roof, Burgdorf, Switzerland, 1965.

were developed. The deformation method for hyperstatic structures replaced, in some cases, the force method. In the 1920s with the increased number of high rise buildings, the deformation method was enhanced by the iteration method of Hardy Cross (1930), more suitable for high-grade statically indeterminate structures.

By the 1940s, in addition to the use of concrete and hot-rolled steel sections, cold-formed steel members began to be widely used in building construction, particularly in roof decks, floor decks and wall panels. Steel roof decks were successfully used in folded-plate and hyperbolic-parabolic roof construction. Compared to other materials, such as concrete, cold-formed steel members possessed many qualities: lightness, high strength and stiffness, easy erection, transportation and handling, the combination of which often resulted in cost-saving. However, the slender nature of such structures causes local and global buckling phenomena, which are of major concern for design. Although rolled steel members have been used in construction since the second half of the 19th century, only from 1940 were buckling problems investigated firstly in the USA and later in other countries.

In the same period the ultimate strength of structures, in particular hyperstatic ones, became the objective of analysis and this led to the development of the plasticity theory. The first papers on plasticity theory date back to a congress in Berlin in 1936, which was held in view of the growing use of steel for industrial and large commercial and domestic buildings. So far they had been designed on the basis of the elastic theory. This is based on the assumption of a perfect structure where small errors in manufacturing and construction, temperature variation, settlements of supports, etc. are often enough to invalidate the elastic calculation. The conclusion was that the elastic stresses are not always relevant to the prediction of the strength of a structure. Jon Backer, an expert member of the Steel Structures Research Committee set up by the British steel industry in 1929 was the first to have the “idea” of plastic design. He worked mainly on continuous beams and portal frames. Backer’s approach to plastic analysis was of a statical nature: the resistance moment acting where plastic hinges have developed is introduced into the statics equations. In this way the collapse mechanism of fairly simple structures could be forecast accurately.

In 1949 Prager and his colleagues developed the fundamental theorems of plasticity theory.
Chapter 2. Structural Analysis: A Brief History

The so called lower-bound theorem, however, had already been stated and proved in 1936 by the Russian scientist Gvozdev. According to his theorem, if a set of forces within the structure are in equilibrium with the external loads and do not violate the yield conditions, then the corresponding value of the load acting on the structure is a lower-bound, i.e. safe estimate of the collapse load. The theorem explains why the elastic theory results in a safe, but often uneconomical design. More importantly, plasticity theory made engineers think about the actual behaviour of a structure under collapse loads. The experimental work of this century has showed that unknown and unpredictable “imperfections” - a small settlement in a foundation, a slip in a connection, a differential rise in temperature - can produce large changes in the actual stress state so that the calculations made to find this state are possibly unreliable. It has been an important contribution of plasticity theory to prove that imperfections have, in fact, no influence on the ultimate strength of ductile structures. The theory was developed largely with reference to steel construction but can also be applied to concrete structures reinforced with ductile steel.

2.4 Integration Period (1950-1975)

Until the mid 20th century, despite the use of simplified calculation methods like the force method, the displacement method and the Hardy-Cross method, it took a long time to analyse structures even of medium complexity, mainly due to the difficulty of solving linear equation systems. In the late 1950s the advent of computers and the development of the Finite Element Method (FEM) completely revolutionized structural analysis. The FEM was developed in the field of aeronautic engineering, where calculations of high-redundancy systems were the norm. The creators of FEM defined it as a “method of analysis for highly redundant structures which is particularly suited to the use of high-speed digital computing machines” [3]. The main persons concerned with the initial development of the FEM were Argyris (Stuttgart and London), Clough (Berkeley) and Zienkiewicz (Swansea). The first FEM programs SAP, ADINA, ANSYS, NASTRAN, MARC, etc., were used only by specialists in big companies and computing centers which could effort expensive mainframe computers. They were large, slow and with no graphical user interface. The FEM combined with the computer has lead to modern computational mechanics, of which structural analysis is a part. Through digital computing, non-trivial calculations concerning dynamics, collapse mechanisms, materials and geometrical nonlinearities as well as ultimate loads could also be routinely performed.

2.5 Diffusion Period (1975 until now)

While the FEM theory has been to a large extent developed in the 60s and early 70s, the access to suitable hardware remained difficult. However, with the advent of the personal computer in the 80s and the Internet in the 90s, the work of a civil engineer related to structural analysis has been totally revolutionized. Today one is no longer required to manually perform or even be able to perform structural analysis computations, as was the case in the past. Today, thanks to developments in computer graphics, a large variety of
user-friendly FEM computer programs are easily available, which in practically no time can any kind of calculation. A structural engineer has now merely to enter a suitable structural model into a FE program and specify the loads to be carried. The computer will then immediately provide the internal structural forces and the corresponding stresses. If the structure does not satisfy the safety criteria, the computer can, in some cases, itself make changes to the design until safety criteria have been fulfilled.

The friendliness of today’s structural analysis programs could lead to the wrong belief that no specialized personnel is required to operate them. In fact, experience shows that the user of an FEM package must be able to properly model a structure, to know the fundamental theories of FEM and have a feeling for structural behaviour in order to properly assess the validity of the computer results. Without these fundamental requirements, dangerous errors can easily occur.

2.6 Information and Communication Technology (ICT) in the Civil Engineering and Architecture Today

The increased memory of personal computers, their high processing capabilities and the user-friendly interface of their applications caused most architectural and construction firms to use computers and software which enabled them to save time and improve the quality of their work, which is known as Computer Aided Design (CAD).

Since the 1990s, when computer networks began to interconnect and communicate with each other, an important role has been played by electronic mail (e-mail) and the World-Wide-Web (WWW). In fact, construction as a multi-organizational process is heavily dependent on the sharing and exchange of large amounts of information. Therefore, the successful completion of a project depends on the accuracy, effectiveness and timing of such an exchange between the project team members, which can be greatly facilitated by the use of the Internet. Through Internet-related software, many corporations can now publish and transfer information in a user-friendly way within a company or among the members of a project team. Currently, several construction firms use the Internet to create project-specific web sites. These web sites are used to access a wide range of construction documents like drawings, specifications, requests for information, budgets and the minutes of meetings. Web-interfaced databases are also used to develop an electronic archive of information as the project is carried out. Here, a web server keeps track of the construction process through digital images and documents. This can also result in facility management tools used to administer and buildings after they have been completed.

With the dramatic development of global networking tools, many construction companies have teams all over the world, which through the Internet, can retrieve stored digital drawings and documents. Rapidly developing video conferencing tools are also changing the way construction projects are run. Virtual conferencing tools are now being bundled with standard computer software packages so as to enable project teams to collaborate,
exchange drawings and solve problems without having to travel to the job site.

2.7 Creativity in Engineering Education

The final goal of civil engineering education is to adequately prepare students for their future profession, so as to meet the requirements of the construction industry, which is today completely based on ICT, as was briefly explained in the previous section. This means that schools and universities should train students to be familiar with the latest technology, networking applications, digital teleconferencing, e-mail, computer-aided applications, databases, etc, so that they are proficient by the time they start working in practice [4].

In addition, the rapid growth of technology and the increase in competitiveness has turned creativity into a major asset. The link between growth of technology and the demand for creativity can be motivated as follows: engineers, who do not have to spend most of their time doing tedious tasks, which are now automatized, can devote their time to more challenging and creative aspects; as a result the possibility offered by the new technology, such as animations, simulations, 3D, virtual reality representations, etc., can enhance creativity. This has therefore become something highly valued which companies look for in individuals. The ability to be creative is closely linked with the ability to be good designers. It has to be stressed here that being a good engineer means being able to design and not to analyze! Therefore it is mandatory that civil engineering schools, aided by the new media, promote design classes and creativity from the early years of the curricula. Squeezing design experience into the final years of study might well be too late [5].
Chapter 3

Teaching the Fundamentals of Structural Analysis and Design

3.1 Traditional Teaching

In this section, the traditional approach to teaching structural analysis is briefly discussed in relation to the undergraduate level of the architecture and civil engineering faculties. This approach is largely a product of the historical development of scientific thinking and of the growing role and influence of engineering for the design and construction of buildings. This has led to the introduction of sophisticated mathematical models into the construction process. As a result, these methods have become dominant for teaching structural analysis to both architecture and civil engineering students.

If this mathematical approach is appropriate for students, who are used to scientific subjects, it could be less adequate for students who have difficulty to study in a rational way. This problem is particularly evident when teaching structural analysis to architecture students. In fact, in contrast to engineering students, they are less interested in mathematics and are used to learn in a visual, qualitative and creative way. Furthermore, they are often not sufficiently motivated to learn structural analysis, mostly because in general they do not have related fields in their later studies.

Thus there is, sometimes, frustration and dissatisfaction when examining the knowledge gained at the end of structural analysis courses. Students often do not understand the basic structural concepts or how structures work, e.g. they have difficulty in grasping whether a bar is in tension or in compression or why the diagonal in a truss may need to change direction at the mid-span; or, they do not know where a simple structure like a continuous beam is most stressed, how the length of a cantilever beam influences bending moments and displacements, what is the direction of the support reaction when a continuous beam fixed at one end is loaded at its free end, how stresses are related to structural dimensions, etc. Furthermore, it is also observed that some students learn by rote the rules taught by their teacher during the lesson and apply them as a recipe without considering whether this makes sense.
Facing this problem, it is important that teachers in architecture adopt alternative ways of teaching structural analysis, e.g. considering practical, qualitative and visual aspects, in order to meet the students’ needs and increase their interest. Students, in fact, should be aware that designing structures and qualitatively understanding how these work is fundamental in their education.

Considering now civil engineering students, they should not only qualitatively understand how structures behave when loaded, but also be able to quantitatively analyze them in order to predict section forces and stresses so as to compute the needed section dimensions. Thus, the content of their structural analysis courses covers more aspects than those of architecture students: They should learn more about complex structures and different analysis types, like stability, $2^{nd}$ order theory, dynamic analysis or the principles of plasticity theory and its related theorems.

Although civil engineering students are supposed to have a strong scientific and mathematical foundation before attending structural analysis courses (thanks to subjects such as mathematics, physics and mechanics), examination of tests and homework reveals problems similar to those of architecture students. It has been observed that, in some cases, students learn by rote the techniques taught by their teachers, and, realizing that similar problems are set in the exam, they become proficient in dealing with these pictorial representations. It emerges, however, that the analytical technique has been learnt parrot fashion, and students may be competent at applying it, but if the formulation of the solution requires the use of previous analysis techniques, they get into difficulty because they have to apply knowledge in a context different from the original one. In this case, there is evidence that some students learn structural analysis as a series of disjointed mathematical techniques. The difficulty for them is to recognize the pictorial abstract representation of the problem and associate it with the correct technique and move through the mathematical process to achieve a numerical solution. A physical understanding of the problem and an intuitive interpretation of the theoretical calculations are rarely present.

Yet many of our great engineers have emphasised the importance of such understanding. According to Nervi (1956):

“The mastering of structural knowledge is not synonymous with the knowledge of those mathematical developments which today constitute the so-called theory of structures. It is the result of a physical understanding of the complex behaviour of a building, coupled with an intuitive interpretation of theoretical calculations”.

Conceptual understanding of structural behaviour is a qualitative approach, in contrast to a quantitative approach. It involves the ability to visualize and sketch the deformations and the section force distributions in a structure, to develop a feel for section dimensions and material properties and to understand how a parameter’s change influences structural behaviour.

Developing such a qualitative understanding is particularly important today due to the requirements civil engineers have to meet in engineering practice. This is the consequence of the advent, in the second half of the $20^{th}$ century, of the computer and the development of the Finite Element Method (FEM), which has progressively found application in a multitude of computer programs for structural analysis. In pre-computer days one of the
most important tasks of a structural engineer was to be a good “calculator”. The young graduate became proficient in dealing with the analysis techniques gradually and only when his analytical skills were sufficiently trained did he gain a conceptual understanding of structural behaviour. Today young engineers are expected to perform the analysis of complex structures using the computer analysis packages that are found everywhere in industry. Therefore, they should primarily know how to properly model real-life structures for the computer to provide meaningful results and qualitatively assess the validity of the output results. In this context, the task of modern engineering education is to adequately train students to meet these needs.

Emphasizing the importance of qualitative understanding when teaching structures to civil engineering students, however, should not lead to the wrong belief that the quantitative understanding is not important. On the contrary, qualitative analysis must not be considered a substitute for numerical analysis but a necessary complement to it so that the two approaches constitute a complete whole giving an ability to evaluate structural performance [6].

3.2 Alternative Approaches

As previously mentioned, the goals of students in civil engineering and architecture are different and this justifies the different contents of their structural analysis courses. Nevertheless, both types of students need to develop a better understanding of structural behaviour. For this reason, developing innovative approaches to the teaching of structures at the undergraduate level of both architecture and civil engineering is not only desirable, but necessary. Facing this problem, teachers follow different ways. In some cases, they set up a series of “hands-on” experiments, that are designed to provide first hand experience for observing structural behaviour. In other cases they use graphical methods coupled with numerical techniques in a highly visual context. Finally, some teachers, taking advantage of the development in digital animation technology, use computer packages to teach structural principles in a more motivating, appealing and therefore efficient way. Examples of the different approaches already mentioned are discussed below.

3.2.1 “Hands on” Experiments

Especially in the case of students in architecture, who are more inclined to a visual approach, some teachers believe that conducting experiments with real small scale structures can help them better understand the fundamental principles of structural behaviour. This belief is shared by many schools, among them the Architecture Department of the ETH of Zurich, where “hands-on” experiments have been designed for the first structural analysis course [7]. Some of these face-to-face class demonstrations are summarized below.

**Simple beam structures.** A typical example is a wooden beam simply supported at both ends with one horizontally moveable support and loaded in the middle. Students observe how the beam bends when the moveable support moves. In a further step, by
increasing the loads, they notice the linear proportionality between displacements and loads. Then, by using beams made of a different material, with the same sections, support and load conditions, differences in their elastic behaviour can be shown.

**Simple frames.** A standard experiment shows the comparison between the behaviour of a three-hinged, a two-hinged and a fixed wooden frame vertically loaded in the middle of the cross bar and then only horizontally loaded in one corner. In both cases students can observe that the deformations of the three frames are different under the same load conditions. In fact, in the case of the vertical load, the deformations of the fixed frame are much smaller than those of the other two frames. The largest deformations can be observed in the three-hinged frame. Then, considering the horizontal load, the deformations of the hinged frames are more or less the same whereas the fixed frame deforms least.

**Arches.** An experiment on a wooden arch loaded vertically is performed with the aim of demonstrating that the arch acts in compression and that an interior chain connected to and retaining the base supports acts in tension.

**Trusses.** In a typical experiment a wooden truss is set up firstly without diagonals. Students should observe that the truss is unstable and it becomes stable only if diagonals are inserted in each rectangular field. Then, when some wooden diagonal bars are replaced with steel wires, students can observe that local instability occurs if the steel diagonals are stressed in compression. Displacements of trusses of different height can also be measured and compared.

Although demonstrating the structural behaviour using physical models is very helpful, some obvious limitations have to be mentioned. First of all, the number of experiments is limited to a few examples; secondly, the time required for setting up each experiment is considerable; and finally, students tend to be passive during the demonstration.

Considering this, some teachers, try to involve students in building “hands-on” experiments by themselves in order to increase their interest and motivation. For example, in a structural analysis course at Imperial College, London, students are given the task of building a bridge of given length using the least material possible. The final experiment is conducted by the professor who has to walk over the student’s bridge. Two prizes are awarded for the least weight and deformation solutions.

Also, in the College of Architecture in Madrid, Spain, the students develop each year a project where they have to build a structure with small timber bars cut by themselves using only glue to join the elements.

The process of production, assembly and construction, can sometimes be guided by computer software. This is for example the case of a structural analysis course at the University Jaume I de Castellon, Spain. An educational project is assigned where students are asked to build their physical model using bars and plastic joints and the commercial computer program SAP2000. Firstly, students have to choose the type of structure to build, to evaluate its model (truss, frame, etc.) and then to sketch the new structure based on the previous one. After that, they have to prepare a physical model of the structure using plastic beams and joints. Once the small structure is built, students have to analyze it using the computer program, from which they obtain the deformed shape, the maximum
displacement, normal forces and bending moment. Based on the results obtained, they improve the structure and decide which elements could be removed without influencing the overall performance. The last two steps have to be repeated until the optimum design of the structure is reached.

In another educational project students are motivated to understand the failure mechanism of a balsa-wood structure. In this case a load is applied up to structural failure. The structure has to have a span of 1.2 m, a maximum height of 1 m and a depth of 30 cm. Students have to use a specific balsa wood plate 2mm thick, which can be cut and joined with glue. The winning group is that which obtains a better ratio ‘ultimate load/weight of the structure’.

In this project, students propose the geometry and an ultimate load is applied at the midpoint. Then, they analyze the structural model with the program SAP2000 and, based on the results, they continuously modify the model adding or deleting bars, changing section properties and modifying the geometry. Finally, they physically realize the designed model and at the end of the experiment they explain to the others the behaviour of their structure and the improvements made.

Results achieved with these projects are both positive and negative. On one side, students acquire interest in structural performance, looking at real-world structures such as bridges, industrial buildings, and so on. They also seem to better understand the theoretical principles if they can apply them to concrete experiences. Furthermore, they learn how to work in a group when building physical structures and to present and justify their ideas when explaining their project to other students. On the other side, the time required for setting up the experiments is considerable. Furthermore, the material used is limited to wood and plastic whereas it would be desirable to work with the materials (steel, aluminium, reinforced concrete) encountered in practice [8]. Then, finally, the use of a commercial program such as Sap2000 could prove to be too difficult. In fact, as it designed to deal with practical engineering problems, it is too general and therefore not suitable for teaching purposes.

Also in the Civil Engineering Department of the Federal Institute of Technology, Zurich, students perform experiments similar to those already described above using the software EasyStatics as an aid in the design (see Chapter 6). The main advantages of such a computer program is that, in contrast to commercial programs, it was conceived exclusively for educational purposes.

### 3.2.2 Graphic Methods

In order to produce engineers who better understand the structural behaviour, are more creative and therefore better prepared for design work, some structural analysis teachers are reconsidering the pedagogical approach of famous professors of the past, like Wilhelm Ritter (1876-1956, Federal Institute of Technology, Zurich, Switzerland) and Pierre Lardy (1903-1958, Federal Institute of Technology, Zurich). Among their students were Robert Maillard (1872-1940), Othmar Ammann (1879-1965), Heinz Isler (1926-) and Christian Menn (1927-), who went on to become widely recognized as the greatest designers of the
20th century. The link between the achievement of these four designers and what they were taught is considered by most people as compelling evidence of the power of education to influence the practice of structural design.

In the pedagogical approach of Ritter and Lardy much importance was attached to the use of simple graphical methods and a critical study of existing structures. Both favoured simple methods of analysis, including graphical methods, with which forces in structures are calculated by drawing lines on paper corresponding to the magnitude and direction of the vector representing the forces. These methods are intended to allow designers to visualize the flow of forces throughout a given structure, and provide a direct link between structural behaviour and the structural shape. The use of visual methods for the analysis and design of structures not only was considered helpful for the conceptual understanding of structure but was also believed to provide students with the opportunity to develop creative skills. Ritter and Lardy also placed considerable emphasis on the description and critical discussion of real structures, structural systems and detailing. With these studies students should observe the structural system of the selected examples and consider the aesthetic aspects of design in relation with the structural behaviour. This was done because both professors believed that with the analysis of real structures students were provided with a sufficient body of knowledge, i.e. a starting point, which could help them in the creative process of designing new structures. In this process, structural analysis was always closely connected with design and never taught separately. Ritter and Lardy, in fact, regarded structural analysis as a tool serving the needs of design, rather than an end in itself [9].

Although some ideas of Ritter and Lardy can still be considered effective, for example the need of coupling structural analysis and design and fostering visual understanding and creative and critical thinking, we consider that introducing their graphical methods into modern engineering courses would be a retrograde step. These methods, in fact, were certainly helpful in the past, and were introduced in order to simplify complex and time-consuming algebraic calculations needed at that time. Nevertheless, such methods still require a lot of time to be performed. In fact, firstly the structure to be analysed has to be drawn followed by the whole graphical construction. In addition, in the case of variants, parts of the structure and of the graphical representation have to be erased and redrawn again. To do all this the time spent can be considerable. Furthermore, the cases in which graphical methods can be applied are limited. In fact, they can be used only to analyse arches, beams and cable structures. With today’s software analysis packages, which can immediately analyse any type of structure and immediately display the results in an appealing graphical user-interface provided with plenty of visual clues, the use of such graphical methods is, in our opinion, no longer justified.

### 3.2.3 Computer Simulations

In order to improve the students’ visualization of mechanics principles and the conceptual understanding of structural behaviour, to enhance their motivation and to foster their creativity, in recent years many engineering instructors have taken advantage of the great potential offered by modern software technology. Its role has also been recognized
by modern didactic studies to be conducive to the learning process (see 3.3.2). Modern technology, in fact, allows interactivity, which makes students assume an active role in the learning process, allows simulations, animations and virtual reality, which helps students visualize abstract concepts, allows them to simulate the performance of design, and provides knowledge anytime and anywhere, so that students can always review lessons on their own without being physically present in the classroom. On the side of instructors, technology also helps one to concentrate on core aspects when teaching structures to students. For this reason, many engineering instructors have attempted to improve students’ learning by incorporating computer based, multimedia examples and modules in the classroom [10].

Since the end of the 1980s increasingly sophisticated interactive educational tools have been developed which simulate the behaviour of structural components after any parameter change following a mouse action and which can therefore assist in the visual understanding of structural concepts, especially at the introductory levels. In these computer programs visual analysis is generally supported through certain visual aids. These aids support the visual representation of concepts based on calculated values. For example compressive and tensile forces in a truss are displayed by drawing the truss elements using the blue and red colours, respectively, and assigning them a thickness proportional to the force value. By using the mouse or text fields students can change the force magnitude, supports and forces location, structure geometry or section and material properties and after any change results such as displacements, bending moments, normal and shear forces are immediately refreshed, recomputed and redisplayed [11]. Some examples of this kind of structural analysis learning tool are Structural Gizmos [12], Deflect [13], Grips [14] and EasyStatics.

It has to be mentioned here that these computer tools do not have to be considered as a replacement of the traditional class course but as a necessary complement to it. In fact, on-line simulations, examples and demonstrations available on the web should be given in addition to books and face-to-face lessons in order to help students to learn more efficiently and review concepts outside their class. A challenge that must be addressed in the successful use of the new media, however, is overcoming the students’ tendency to passively receive instruction. To be effective software must use the capability of the computer to engage the attention and stimulate students, both visually and through interaction and feedback, somehow like a video game.

### 3.3 Didactic Theories

In order to improve the student’s understanding of structural behaviour so as to adequately prepare them to meet their future needs, teachers should firstly know how they actually learn and consequently find out the best didactic methods and teaching strategies to use.

In the following sections the most recent didactic theories of behaviourism and constructivism are described. For most of the 20th century education has been strongly influenced by the first theory, whereas the second one is now considered by many educational psy-
chologists to be more adequate to the learning process.

### 3.3.1 Teacher-Centered Instruction and Behaviourism

According to the behaviourism theory the learning is considered as a sequence of stimulus from the teachers and response actions in the learners [15]. The teacher, sole content provider in the classroom, determines what students should know and makes sure they learn in a step-by-step manner [16]. The method assumed is transmissive and teacher-centered. Students rely on lecturers to feed them with information, are passive recipients of knowledge, and thus often lack “ownership” of what they learn. Individual subjects are taught in an encapsulated manner. The student’s main task is memorization and absorption of facts delivered to them by an authority, the teacher, who is regarded as “the sage on the stage”. Then, they “regurgitate” what they have memorized in a test or examination which is used as the measurement of successful learning. This method does not encourage active participation of students and does not encourage them to become independent and autonomous [17]. Furthermore, it promote individualism, competition and assumes that all students learn in the same way [18].

Nowadays, employers complain that young people educated according to this method are not adequately prepared to meet the needs of engineering practice [19]. In fact, this has experienced a process of rapid change, which is a consequence of the recent technological revolution. Automated processes have mainly replaced the traditional work of engineers, who are now required to deal with different tasks and therefore should develop different skills. Today engineers are compelled to update their school knowledge and acquire and work with new information. They are required to have a solid training in science and technology, to have knowledge about computer hardware and software, to be able to communicate, to develop projects, to analyse problems, to work in a team, to show initiative, entrepreneurialism, creativity, flexibility, motivation, and interest in life-long-learning [20].

This change in the engineering work must inevitably affect engineering education with the consequent fast expansion of the basic engineering knowledge and the obsolescence of much of what was learnt at engineering schools in earlier days. Also the prevailing instructional teacher-centered model, based on knowledge transmission and heavily on analytical problem-solving methods learned by rote, needs to be replaced by another model of instruction [18].

### 3.3.2 Student-Centered Instruction and Constructivism

The learning-centered model, also called constructivism, is nowadays considered to be more adequate to develop the skills required by modern industry, such as critical thinking, ability to solve problems, to work in a team and to communicate. This model is based on several aspects, the most important of which are listed and explained below:

- active construction of knowledge;
• learning by problem solving;
• strategies for investigating students’ alternative conceptions;
• social aspects;
• scientific observation and the predict-observe-explain (POE) strategy;
• use of demonstrations inside a POE procedure;
• “multiple intelligence”;
• new role of the teacher in instruction.

Active Construction of Knowledge. The constructivism learning method is based on the assumption that students actively construct (rather than acquire) their own knowledge, strongly influenced by what they already know and this construction of knowledge occurs in a social context. The educational psychologist Vygotsky proposed that all learning takes place in the “zone of proximal development”. This “zone” is the difference between what a student can do alone and what he or she can do with assistance. By building on the student’s experience and providing moderately challenging tasks, teachers can provide “an intellectual scaffolding” to help the students learn and progress through the different development stages [15].

Learning by Problem Solving. In order to promote the development of critical thinking and problem-solving skills, the constructive learning approach emphasizes the student’s ability to solve real-life problems relevant to their future practical work. Learning should be problem-based (PBL). Contrary to the conventional model, which uses problems after their content has been introduced, PBL uses the problem to challenge, motivate, focus, and initiate learning [21]. This implies that education should be more deductively-oriented than inductively-oriented, more process-oriented than product-oriented and more practice-oriented than technical skill-oriented [22]. In a constructive environment students typically work in cooperative groups rather than individually; they tend to focus on projects that require solutions to problems rather than on instructional sequences that require the learning of certain content skills [23]. These concepts are well summarized in the sentence of Confucious (450 BC):

“Tell me and I will forget. Show me and I will remember. Involve me and I will understand.”

Recognition of the Previous Ideas. Fundamental for the constructive learning theory is that knowledge is not transplanted to a person’s mind as if the mind were a blank slate waiting to be written on [24]. Alternatively, learners make sense of the world by interpreting new information in terms of what is already known. It is in fact widely accepted that students often come to classrooms with a range of strongly held ideas which are often contrary to accepted scientific views. These alternative understandings of science have been referred as alternative conceptions. These can form a barrier to learning and hence effective teaching in science requires these strongly held ideas to be made explicit and reviewed. The elicitation of student ideas is central to any teaching approach informed
Chapter 3. Teaching the Fundamentals of Structural Analysis and Design

by constructivism ([25]. Thus, the role of the teacher is to recognize the conceptions students bring to the classroom and provide experiences that will help them to build on their current knowledge of the world. In this sense constructivist teachers act as a guide mediating between the student’s everyday world and the world of science.

**Strategies for Investigating Students’ Alternative Conceptions.** Student interviews and oral examinations are acknowledged as the most effective way to probe students understanding. These, however, are time-intensive, thus difficult to apply to large numbers of students. Multiple-choice tests have been used widely as an alternative. These tests are cheap, easy to administer, mark and interpret and can be used with much larger groups of students. Also computers have been used to investigate students’ alternative scientific conceptions. Some of these studies have used computer simulations, where students can see the discrepancy between their prior conceptions and what they see after running the simulation.

**Social Aspects.** One of the problems with early cognitive science theories was the lack of consideration given to the social dimension of learning. Social constructivism acknowledges that learning is a social activity in which learners are involved in constructing consensual meaning through discussions and negotiations with other students and teachers [26]. From a social constructivist perspective, the development of understanding by writing and discussion of ideas with fellows is an essential part of learning and involves articulation, reflection, elaboration, negotiation and consensus-seeking: “Accordingly, students should be encouraged to be involved in putting language to ideas, testing their understanding with peers and listening and making sense of the ideas of other students. Students can articulate their own views, exchange ideas and reflect on other views, reflect critically on their own views and, when necessary, reorganize their own views and negotiate shared meaning” [27]. Articulation is a first step to elicit and then clarify one’s own ideas. Until ideas are publicly articulated, neither teacher not student can assess their meaning and validity [28]. Articulation of students’ ideas sharpens their conceptions and leads to a recognition of new connections [29]. The process of reflection involves actively monitoring, evaluating, and modifying one’s thinking: “Human reflection is the key to understanding and creating a new world in which we coexist with others” [30].

This type of thinking requires both individual and collaborative reflection [31]. On a social level processes such as explaining, clarifying, elaborating, justifying, evaluating, analyzing, synthesizing, questionning, and restructuring occur as students reflect on what they have learned and they interact with other students. Student discussions make ideas visible to others and give learners opportunities to share multiple viewpoints, listen carefully to others’ ideas and promote mutual respect between peers [29]. If a person is involved in a group discussion, it is necessary to listen to the views of others and determine whether or not the others agree. Feedback from the listeners becomes a crucial stage in the process of reflecting on one’s own and the ideas of others. Student negotiation is concerned with creating opportunities for students to explain and justify to other students their ideas; listening to, understanding and reflecting on the viability of the ideas of others and subsequently reflecting on the viability of their own ideas. This process is recognized as a key part of learning under a social constructivism framework. As students construct shared knowledge in peer groups, teachers take on the role on supporting students’ interaction
Chapter 3. Teaching the Fundamentals of Structural Analysis and Design

with their peers and the physical world. They assist students reconstructing their existing conceptions and determine the viability of the new ideas [32]. Teachers need to promote mutual respect among students by explaining to them the basic notion of constructivism that different students can hold different perspectives about a concept at the same time [30]. Students need to learn how to generate ideas in discussions, and accept and elaborate on others’ ideas without criticizing them. In a social constructivist learning environments there is in fact the need of an open discourse. In addition, peers’ discussion becomes an opportunity for learners to talk science and learn the language of science [33]. Indeed, learning science means to achieve a certain level of competence in talking science [34].

Scientific Observation. Scientific observation is a complex process. It is widely accepted that observation is interpretation [35]. This interpretation arises through one’s prior experience of the world; what one observes depends on what one already knows. Hence, “Looking at is not a passive recording of an image like a photograph being reproduced by a camera” [36]. The quality and the usefulness of observations also depend on the language available to the observer. Indeed, the language used in students’ observations provide an assessment of how well students can use relevant canonical discourse relating to the phenomena studied [37]. One aim of scientific education is to teach students to be observant and be objective and precise in their reporting and recording of events. From a constructivist perspective, the articulation and writing of observations also provide an important insight into students’ strongly held ideas and beliefs [36]. Therefore using POE tasks is an instrument to elicit students’ scientific ideas. Unlike other probes such as interviews, POE tasks take less time to complete, they can be used with large groups and they do not require special teacher training to administer and interpret results [38].

The Predict-Observe-Explain (POE) Strategy in Science Education. The POE procedure is based on the classic model of research where a hypothesis is stated and reasons are given for the way this may be true, relevant data are gathered and results are discussed. A POE task involves students predicting the results of a demonstration and finally explaining any discrepancies between their predictions and observations. The procedure generally uses observable, real-time events as stimuli to provoke student thinking about concepts. Demonstration Inside a POE Procedure. Demonstrations are an extremely valuable teaching strategy. They can be used to introduce or review topics, set a problem and connect concepts with real-life examples. They can be stimulating and motivating for students and allow teachers to focus student thinking and observation on specific outcomes of events. The visual aspects of demonstrations provide a complementary way of understanding a concept. Multiple visual examples presented in demonstrations promote the use of correct analogies in forming and transferring concepts. A demonstration can be embedded in a POE procedure. The commitment to a prediction enhances students’ understanding the demonstration while reasoning provides a further focus for the observation and builds motivation. The observation phase can initiate discussion and foster valuable learning, while the explanation phase provides a significant opportunity for discussion. These positive learning effects are enhanced when students are required to write their responses to each components of a POE task. Writing predictions, reasons and observations increases the level of commitment, encourages the formation of links between new and old concepts [39].
“Multiple Intelligence”. Following Carl Jung’s theory of “multiple intelligence”, researchers in education developed different models for learning styles in an attempt to classify stereotypes. A modern teacher should be aware of these studies and prepare a variety of set activities within a course to be able to reach all students in their preferred style. There are different models of learning styles. For example, the elder-Silvermann Learning Style Model [40] divides students into the following categories:

- sensing learners (concrete, practical, oriented toward facts and procedures) or intuitive learners (conceptual, innovative, oriented toward theories and meanings);
- visual learners (prefer visual representations of presented material—pictures, diagrams, etc.) or verbal learners (prefer written and spoken explanations);
- inductive learners (prefer presentations that proceed from the specific to the general) or deductive (prefer presentations that go from the general to the specific);
- active learners (learn by trying things out, working with others) or reflective learners (learn by thinking things through, working alone);
- sequential learners (linear, orderly, learn in small incremental steps) or global learners (holistic, systems thinkers, learn in large leaps).

Most of the time in engineering teaching programs, the didactic strategy favours only a few styles. For instance, lectures only favour students who are abstract and reflective. Studies have shown that 20% to 40% of engineering students do not fall into this category. Therefore a rich set of activities within a course should be planned to address their best learning mode [41].

Role of the Teacher. In a student-centered approach the role of the teacher changes: from “the sage on the stage” to the “guide on the side”. Teachers should now provide the appropriate tools and environment for helping students construct their knowledge. In order to do this they have to conceive their courses based on a weakly structured problem since students have to be active learners, value the students’ previous knowledge, stimulate their initiative, motivate them, provide them with open-ended problems, assist them in delineating issues, formulating problems, exploring alternatives, making effective decisions, etc.

3.4 Use of Technology in Science Education

In the following sections the use of technology in relation to the didactic theories of behaviourism and constructivism is described. It will be shown that, when used in a constructivist environment, technology better helps students to become critical thinkers, able to deal with open-ended problems, to understand concepts and to work and collaborate with others.
3.4.1 Use of Technology to Deliver Information

Today, as a rule, educational technology is being used as a medium to deliver information which can be “stored” in the computer. To this category belong the drill and practice and many tutorial systems, also called instructions-based systems. In the drill-and-practice systems, in accordance with the so called stimulus-answer model, the knowledge is transmitted by means of a series of instructions. These systems are mainly aimed at having the student acquire specific knowledge and basic skills. By means of some questioning strategy, play elements are used for motivating and encouraging students. These systems typically lead to the acquisition of passive knowledge. An example of a drill and practice system is the online structural analysis course Calice \(^1\) used at the civil engineering departments of the ETH Zurich.

The tutorial systems are primarily designed for distance learning courses based on the Web. They include instruction content on a given topic. Importance is ascribed to factors such as reinforcing memorization, presenting objectives, specifying prerequisites, as well as eliciting and assessing performance. The questions asked require the application of the concepts or rules covered in the instructional sequences. Feedback is often diagnostic, identifying errors and providing hints to the right answers. An example of a tutorial system is the online structural analysis course i-Structures \(^2\) used at the Civil Engineering Department of the EPFL Lausanne. In these systems students are taken through predefined paths. The whole traditional lecture is subdivided into topics. For each of these there is a theoretical explanation, a text with multiple choice tests, exercises requiring manual calculations and the insertion of the results into text fields. Once students have answered the quizzes and have filled in the text fields they can see how many correct answers they have given. If they are not satisfied with the points scored, they can go back and repeat the exercise until the correct answers are given. For increasing the learners’ motivation animated and colourful pictures are present in the learning environment. If compared to a traditional course in the classroom, these systems offer also the possibility of being available at any time and anywhere.

Despite all their advantages, these systems present the limitations which are inherent in teaching methods that want to steer the learning process by external stimuli. Like in lectures without technology, students are guided through the learning process, following only one predefined path which leads to the unique correct result. In such a way they are not required to be creative and active in order to find by themselves the right way by choosing different alternatives. Furthermore, the activities done with these systems give limited opportunities of problem-solving and cooperative work. Generally, learners are considered as passive recipients of instructions with an emphasis on memorizing facts and isolated skills. Authoring systems typically support the development of this type of teaching material. The overall goal is the acquisition of knowledge. With these information-based systems students are supposed to learn from the technology, which contains and delivers knowledge. In such a way rote and not conceptual learning is favored.

---

\(^1\)Calice: www.ibk.baug.ethz.ch/Vo/calice

\(^2\)i-Structures: http://i-structures.epfl.ch
3.4.2 Use of Technology to Build Competencies

With the emergence of activity-based theories in the educational field, the way to consider technology for teaching and learning purposes has changed. Knowledge is not longer considered as a reflection of an external reality but rather as a function of perceptual process. Learning environment, that support activity-based, explorative interaction with the environment are based in the constructivistic approach discussed in the previous chapter. In order to construct knowledge students should actively work with computers (rather than learning from computers). In a constructivist learning framework, pedagogy and content matter most and technology and media are only vehicles which help in the process of construction rather than transmission of knowledge. This means that it is not necessarily the computer by itself that improves the learning but more how the students and the teacher use it. Hence, constructivist learning can take place with the support of computers if teachers adopt a constructivist approach. Thus, like before, they continue to perform a central role in mediating the quality of students’ learning [42]. In the information-centric approach media and technology are given directly to students to use for expressing and representing what they know. Learners themselves only use media and technology as tools for analyzing the world, accessing and interpreting information, organizing their personal knowledge, and representing what they know to others [43].

Next we consider how computer supported technologies can be used to create different learning environments such as collaborative learning, situated learning, learning through video-based laboratories, simulations and micro-worlds. The meaning of these concepts is explained below.

Collaborative Learning. There are three other types of technology that support collaborative learning. The first category includes those types of programs that provide the environment for collaboration. Included here are word processors (e.g. Word), spreadsheets (e.g. Excel), databases (e.g. Access), drawing programs (e.g. Corell Draw), desk top publishing programs (Microsoft Publisher), Multimedia Presentations (e.g. PowerPoint). Each of these tools provides the environment for students to produce a product that they can share with the class or publish in a newsletter or on the World Wide Web. Using these programs students must make important decisions on what information to convey and how to convey it. Learning to choose material, to work together, and to value each others work for their multiple intelligence are all goals of cooperative learning that teachers can focus on. To the second category of technology supporting collaborative learning belong all those programs that will provide the resources needed to complete the research for class presentations, e.g. the World Wide Web. The third kind of technology that supports collaboration among students consists of chat rooms and conferencing facilities for synchronous collaboration and other systems such as e-mail for asynchronous collaboration. In addition, web facilities can support and keep track of synchronous dialogue among students that then serve as a public archive of conversations: “conversation can be stored, reflected on and reacted to, creating a common knowledge base that is open to review, comment and manipulation” [44].

Also EasyStatics and its e-learning platform promote collaborative learning. The program, in fact, allows students and teachers to generate EasyStatics files including com-
ments upon the represented structural model or to export *EasyStatics* pictures to include in any word processor or presentation file. The e-learning platform can be used among the users to deliver and receive messages and documents. These can be *EasyStatics* files, text files, presentation files, etc. (see Section 3.5 and Chapter 4).

**Situated Learning.** Modern design of interactive multimedia can be created to facilitate learning in environments that “situate” the content to be learned. These programs adopt the view that learning and thinking are fundamentally situated. In these programs students can situate their learning, can create their own meaning and understand phenomena as an alternative to those generated by their teacher. With these programs students are encouraged to be active, constructive and cooperative as they conduct these investigations. The program *EasyStatics* can be also used by teachers to generate demonstrations and exercises connected to a real context, which is significative for learning (see Section 3.5).

**Video-based Laboratories.** Interactive digital video presentations can be used to make observations, measurements and gather data about events. Computer digital video systems allow students and teachers to capture videos of experiments they perform themselves. When connected to spreadsheets, students can then use the interactive video clips to gather data and make graphs and other representations to analyse and model their data. Many studies have shown these video-based laboratories to be motivating and authentic learning experiences for students [45], [46], [47]. Indeed video based laboratories are considered as facilitating a constructivist learning environment by promoting open-ended exploration in an authentic learning context; this is particularly so when the learner chooses and captures his or her own film clips. Video can help to expose students to real phenomena and overcoming the traditional barriers showing difficult, expensive or time-consuming demonstrations not normally possible in the laboratory [48]. For example, video can slow down or speed up time, be played in the forward or backward direction, either through filming techniques or using motion whilst viewing the clips. Digital video clips also allow students to observe accurate and reliable replications of demonstrations and enable students to enjoy a continuous, seamless experience [49]. Hence, interactive digital video makes possible the detailed observation of laboratories of real-life events and is considered an important technology in the area of computer-based learning in science [50]. Such real-life scenarios can make science more relevant to students’ lives [51] and help students build links between their prior experiences and abstract models and principles [52]. “This ease of access to any part of the video changes its function from a linear element used to introduce or enhance instruction to an integral resource that can be explored and analyzed in detail” [53].

**Simulations and Micro-Worlds.** Simulation programs represent a part of reality, emulating physical systems and processes and allowing experiments that are normally impossible, dangerous, inaccessible, etc. Simulations which can be reviewed at any time aid visualizing abstract concepts: “New technologies, such as computer simulations can help to make the reasoning of students explicit and help them to visualize the consequence of their thinking” [54]. Students can adjust variables and observe effects as they use the simulations to enhance the inquiry process. Specific types of simulations are the micro-worlds where users can explore problems, experiment, test, revise and hypothesise [50]. They often
use images that represent a concept (semantic icons) which can be manipulated by the learner on the screen (e.g. by a mouse click or point etc.). They are imaginary worlds where students can investigate scientific problems, hypotheses, design alternatives, test their ideas and use feedback to reflect on ideas [54]. Problem-based learning is therefore facilitated by micro-worlds. An example of these special type of simulations can be considered the program *EasyStatics*. In fact, it allows students to observe the structural behaviour by modifying structural parameters using mouse clicks or text inputs. In such a way students can match their prior ideas (see Section 3.3.2) with the simulated situation. Many benefits can be ascribed to the use of educational multimedia. First of all using multimedia, e.g. visualization, modelling and multiple representations, enhances learning by providing opportunities to construct and articulate meanings that in the traditional class would be limited to rendering and expressing in words only. In addition, multimedia use is believed to foster aspects of intrinsic motivation such as fantasy, challenge and curiosity [55]. In agreement with this, multimedia presentations are considered engaging because they are multimodal. In fact, multimedia can stimulate more than one sense at a time, and doing so, may be more attention-getting and attention-holding. Multimedia gets students’ attention and often has a real-world connection. Students can learn in the context of real-world problems and address the question, how these problems can be solved. Furthermore, multimedia accesses multiple intelligence. Sounds access student’s rhythmic intelligence. The video clips access the visual spatial intelligence. Questions access logical and intra-personal intelligence [43].

Research and evaluation shows that using multimedia in the teaching and learning environment enables students to become critical thinkers, problem solvers, more apt to seek information, more collaborative and more motivated in their learning process [56]. Furthermore, if used in a constructivist setting, multimedia increase self-determination and intrinsic motivation which are linked with the notion of learner control: “The more learner controlled the instructional systems are, the more generative they are; that is they require learners to generate or construct their own knowledge” [57]. The notion of learner control refers to students taking control over some aspects of their computer-mediated learning environment. It may imply control over the pace of information presented, the order or sequence of information presented, the timing of information presented or the quantity of information presented [58]. It may also refer to control over difficulty levels, content, context, and the methods of presentations [59]. Learner control of the content and sequence in a program can encourage self-management of learning [60]. In this context, where students have a high degree of freedom, there could be the danger that they are not capable of making good use of the control they are given. To avoid this, guidance from the teacher is always necessary.

### 3.5 Learning with *EasyStatics*

The program *EasyStatics* belongs to the type of technology used as a vehicle to build competencies (see previous Chapter). In this context, learning with *EasyStatics* and not from *EasyStatics* means that the program does not directly provide the lectures needed for
a structural analysis course, but is an instrument teachers can use in order to generate their own lectures. For this purpose they are completely free to set up class-demonstrations and exercises using the program’s functionalities, but according to their own preferences.

For students, EasyStatics represents a structural micro-world or virtual laboratory (see Section 3.4.2) where they can experiment with “virtual” structures so as to learn quasi by playing. The type of structures one can analyze is limited to plane frames and trusses. The latter have been introduced only with didactic purposes due to the fact that they are ideal structures and therefore do not exist in practice.

The EasyStatics “micro-world” is mostly based on the visual representation of structural behaviour. Consequently, not all the multimedia capabilities mentioned in Section 3.4.2, such as audio and video tools, were included in the program.

The functionalities of EasyStatics, explained in detail in the Appendix, have been developed to support the constructivist teaching approach (see Section 3.3.2) while observing the following requirements:

- user-friendliness and customization of the graphical user interface;
- exploration, observation, reflection and negotiation;
- situated and problem-based learning;
- demonstrations, exercises and collaborative learning.

These concepts as implemented in EasyStatics are discussed below. Details of their practical realization are found in the Appendix. A number of ready-made examples on the use of EasyStatics as a teaching tool are given in Chapter 5.

### 3.5.1 User Friendliness and Customization

Essential features that IT tools should have in order to motivate students and teachers are intuitiveness and user-friendliness. For this reason, the graphical interface of EasyStatics has been developed so as to be:

- *customizable* by choosing the language in which texts are written, by changing the color of the graphical objects and by selecting only the functionalities one wants to work with and hiding the others (see Appendix Section A.10);

- *self-explaining*, i.e. users should immediately recognize what to do in order to perform a certain operation. Online instructions should appear whenever something is selected (see Appendix Section A.13);

- *highly interactive*, i.e. to any command the system must respond in real time.
Regarding the first point, the possibility to customize the user interface by selecting only the needed functionalities is considered particularly important for two reasons. Firstly, it allows teachers to build their course step by step and, therefore, to focus only on certain aspects avoiding students being distracted by additional features; secondly, because EasyStatics is conceived as a tool for both civil engineering and architectural students, who, as written at the beginning of this chapter, have to focus on different aspects when taught on structural behaviour. Because the structural analysis course content for architects is limited, teachers do not need all the features offered by EasyStatics. For this reason, they can make any feature not needed become invisible.

Considering then the last point, in order to promote intuition on structural behavior, it is necessary that, after any parameter change (e.g. a change in the geometry, the load and the support conditions, the material or the section properties), results such as displacements, bending moments, shear and normal forces, etc. are immediately recomputed and refreshed. This is obtained simply by clicking, moving, dragging and dropping the mouse or editing a value in a text field.

### 3.5.2 Exploration, Observation, Reflection and Negotiation

In this section the functionalities are described which help students to actively explore and observe structural phenomena, reflect on them in order to take structural design decisions, and negotiate with other students and teachers when they have to justify what they have done.

#### Visualization of the global and local instability mechanisms

At the beginning of the design process, students are supposed to enter the geometry of the structure and to adequately define support conditions. As long as these do not guarantee structural stability, EasyStatics draws the global instability mechanism (see Figure 3.1, top) so as to allow students to reflect on the degrees of freedom which need to be blocked. Also when students add internal hinges to the structure, local instability can occur and the related mechanism is shown (see Figure 3.1, bottom). They can therefore realize that hinges make a structure weaker and that only a limited number of these can be introduced without making the structure become unstable (see Appendix Section A.6).

#### Definition of the structural model

After the geometrical model of a structure with adequate support conditions has been defined, loads can be applied (see Section A.1). By pressing and dragging the mouse in specific positions (see Appendix Section A.2), the related graphical objects can be moved, their inclination can be changed and their length, which is related to the load’s value, can be varied. After any change, the analysis is performed again and results are recomputed and instantly redrawn. This occurs not only when load conditions are modified, i.e. loads are added, deleted or moved, but also when, by using the mouse, the geometry of the
model is modified, a hinge is added, moved or deleted or supports are placed, rotated or removed. Much can be learned when observing the relations between changes in the structural model and results. For example, students can understand the principles of equilibrium by observing how changes in the loads modify values and directions of the support reactions. To complete the model definition, material and section properties have to be assigned to the structural members.

**Definition of the section and material properties**

Because *EasyStatics* has been designed for didactic rather than for commercial purposes, students can select only a limited number of typical section and material types (see Appendix Section A.8). After a section type has been chosen, one can interactively modify its geometry, the material properties and, for reinforced concrete, the upper and lower reinforcement contents. As soon as the modified section and material properties have been assigned, the analysis is performed again and the results are recomputed and refreshed, so as to allow students to immediately see the consequences of their changes. Performing a 1$^{st}$ order analysis, teachers can, for example, increase the stiffness of some part of a structure and show the output deformations to their students (see Figure 3.3). Other situations in which the section panel can be used for didactic purposes, especially for civil engineering students, are in the cases of 2$^{nd}$ order theory, stability phenomena, dynamic and plastic analysis. For example, by increasing the section stiffness of structural members, students can observe that the buckling load of the whole structure increases; by decreasing the mass density of the structural elements they notice that the eigenfrequencies increase; by increasing the reinforcement in reinforced concrete members or the yield stresses they observe that the ultimate load increases and the collapse mechanism changes. Other instructive examples of the use of the section panel are found in Section 5.6.
Figure 3.2: Continuous beam with modified load and support conditions

Figure 3.3: Comparison between displacements of beams with different stiffness.
Load cases

Particularly for engineering students, to allow them to understand the principles of load superposition, *EasyStatics* gives the possibility to have two load cases, the effects of which can be seen separately or combined. The limited number of load cases is justified by the fact that *EasyStatics* is not meant to be used for a proper dimensioning according to the building codes, but only for didactic purposes.

Force diagrams representation

After teachers have theoretically explained what are the concepts of displacements, bending moments, shear and normal forces diagrams, *EasyStatics* can be used to display them and their mutual dependency, e.g. the relation between the curvature of the deflected shape and the bending moments, between these and the shear forces, etc. (see Figure 3.4). In order to distinguish the force diagrams, these are displayed using different colours. Two different colours are also used to represent compressive and tensile forces. This is important in the case of truss structures where students can see at a glance which bars are in tension and which are in compression.

Quantitative representation of values

If a teacher wants to focus on quantitative aspects of structural analysis and relate them to the graphical visualization or check the validity of calculations manually performed, *EasyStatics* displays the numerical values of all results as soon as the mouse runs over
Figure 3.5: Comparison between the bending moments of a simple beam under different supports condition.

the related diagrams (see Appendix Section A.11).

Copy option

As an aid to observation and reflection processes, *EasyStatics* allows users to generate up to four independent copies of a model (see Section A.14). On each copy, different results, such as displacements or force diagrams, can be separately shown (see Figure 3.4) so as to give the possibility to reflect on their relation. In the context of 1st order theory, through simultaneous observations, students can see, for example, that when the bending moment is linearly distributed the shear force is constant or that different slopes of the bending moment diagram correspond to discontinuities in the shear force. As another example, one can enter the model of a simple horizontal beam, generate more copies, assign different support conditions and compare the relative bending moment distributions (see Figure 3.5). Another situation in which the generation of more copies of a structure is particularly useful, is in the case of dynamic analysis if one wants to see the different structural eigenmodes and eigenfrequencies. Also in the field of plasticity theory, engineering students can take advantage of the generation of more copies in order to see the differences between a structure analysed according to plastic analysis and to elastic analysis. More examples of the didactic use of the copy function can be found in Chapter 5.
Chapter 3. Teaching the Fundamentals of Structural Analysis and Design

**Compare option**

A visual clue, which can help students to reflect on structural behaviour, is given by the compare function. By maintaining the same scale of the output results after any change, one can see the mutual dependency between loads, displacements and force diagrams. For example, when the compare function is active, if one increases the value of a load, simply by stretching with the mouse the related graphic object, displacements, shear forces, normal forces and bending moments change proportionally so that valuable comparisons can be made.

**Combined use of compare and copy options**

Again, in order to foster reflection and observation the compare function can be combined with the generation of more copies of a model, so that comparison between more systems can be simultaneously made. A situation where one can benefit from both the compare and copy functionalities, is by observing what happens to the displacements and force diagrams of a structure, if the material and the section properties of the structural members change, as in the real examples reported in Section 3.2.1. Also the differences in the behaviour of the three frames described in the same section can be better understood using *EasyStatics*. Particularly for engineering students, another instructive use of these functions is when 2nd order effects are shown and comparisons between results obtained by 1st and 2nd order theory are made. Amplified or reduced deformations and section forces diagrams are displayed if initial compressive normal forces or tensile forces, respectively, are considered, i.e. if a 2nd order analysis is performed. Other similar examples can be found in Chapter 5.

**Stress Panel**

Once the distribution of the normal stresses and their relation to bending moments and normal forces have been theoretically treated, the stress panel (see Section A.5) allows one to visualize these relations. Students can select some bars and for these see the corresponding normal stress distribution. The colours along the elements’ axis and the element height indicate that the stress values change along these directions. As in the case of displacement and force diagrams, *EasyStatics* allows one to see the numerical stress values whenever the mouse runs over the graphical representation of the beams displayed on the stress panel. In order to understand the link between normal forces, bending moments and normal stress distribution students can open an internal panel representing in a reduced scale the whole structure, the selected elements and one of the active force diagrams (bending moments or the normal forces). Another helpful functionality allows one to display compressive, tensile and total resultants along the selected element. Again students can think about the connection between bending moments, normal forces, stress distribution and tensile or compressive force resultants. They can observe that the compressive and tensile resultants are both within the element section, that the total resultant approaches the element the more the normal forces increase and that the compressive and
ten 
tension resultants disappear in order to display only the total resultant. In fact, many 
examples can be explored, from a horizontal beam in pure bending to arch structures, 
loaded mostly by compressive normal forces (see Figure 3.6). Indeed, equilibrium con-
siderations can be made considering external forces and internal total stress resultants. 
Several examples of the use of the stress panel are found in Section 5.2.4.

Truss Model

In the case of reinforced concrete structures EasyStatics allows students to select one 
structural element and to see how forces are distributed inside it by generating a truss 
model (see Section A.5 and Figure 3.7). Tensile and compressive forces are shown with 
the same colours as in the case of truss structures, so that students can understand where 
and in which way reinforcement and stirrups are needed. In addition, they can observe 
the bending moment as well as the normal and the shear force distributions along the 
element and correlate them with the force distribution in the truss model. As in the case 
of the stress panel, an internal panel can be opened on the truss model-panel in order to 
see where the selected beam lies inside the structure. Some examples where this option 
is used can be found in Section A.5.

Check Line and M-N interaction diagram

Using EasyStatics students can see at a glance if the members of a structure are under- or 
over-dimensioned. This is obtained through the so-called check line (see Appendix Section 
A.9). If its colour is red this indicates that the structural section is under-dimensioned, 
whereas if it is blue the structural members are over-dimensioned. Moving the mouse on 
this line, the possibility to see the numerical values of the check line is useful for assessing
how far the section dimensions are from an adequate dimensioning (the numerical value should be small, but near the unit). Finding the appropriate section dimensions is possible by opening the material panel. This not only includes several section types, as written before, but also the M-N interaction diagram related to the section corresponding to the position on the selected bar. The position and colour of a cross symbol corresponding to the M-N values (outside the diagram and red if the bearing capacity of the section is not sufficient, inside the diagram and blue in the opposite case) indicate if the section is properly dimensioned or not. Students can interactively change the section parameters and see in real-time how the M-N diagram and the position and colour of the cross symbol change together with the colour and position of the check line on the main screen. This option if particularly helpful to get a “feel” for structural dimensions and to relate them with the external loads.
Chapter 3. Teaching the Fundamentals of Structural Analysis and Design

Comments, pictures, state-files and HTML editor

As mentioned in Section 3.3.2, one aim of scientific education is to teach students to be observant, objective and precise in their reporting and recording of events. Only when teachers read students’ observations about the phenomena studied, can they assess the quality and usefulness of these observations. For this reason, *EasyStatics* provides students with special internal editors with which they can write what they have observed or justify their structural design decisions. In order to save more structural scenarios with relevant comments students can also generate different computational states (see Appendix Section A.17). For the purpose of reporting and recording events, students can select a part of the screen they want to describe, then save this as a picture and embed it in any text editor, such as the intern HTML editor (see Appendix Sections A.18 and A.19). The documents generated by such text editors can then be used in exercises and class presentation in order to justify and negotiate ones own ideas.

3.5.3 Situated and Problem Based Learning

As mentioned above, it has been shown that students learn better with problems that are rooted in reality. When using *EasyStatics* a teacher can assign a real problem starting by displaying the picture of a real structure on the background of the computer screen (see Appendix Section A.12). On this background picture students draw their design.

To cite an example of situated and problem-based learning, one exercise assigned in the “Computer Static” course for engineers at ETH Zurich has proved to be instructive. Students were asked to design and dimension a bridge crossing a valley, the profile of which was sketched on the background image (see Figure 3.9). Self weight, service loads and a concentrated load corresponding to the weight of a truck had to be considered. Wood, reinforced concrete or steel could be used. Students proposed different types of structure, i.e. truss or frame, different forms, such as arches, simple continuous beam supported by many piles, etc. as well as different materials.

For each alternative they worked with the check line option (see Appendix Section A.9) and the M-N interaction diagram (see Appendix Section A.8) in order to properly dimension the structure. For each alternative they wrote down their considerations showing disadvantages and advantages of the proposed alternatives.

3.5.4 Demonstrations, Exercises and Collaborative Learning

Copies, compare option, comments, states-file, HTML editor, the export of pictures and the possibility to send and receive files to and from the *EasyStatics* server (see Chapter 4) foster collaboration among teachers and students. As mentioned in Section 3.3.2, teachers can use demonstrations in order to introduce and review topics in a stimulating and motivating environment, set a problem and connect concepts with significative real-life examples, promote visual intuition and provide significant opportunity for discussions. After teachers have shown a demonstration in the face-to-face class, they can make it
available on the World Wide Web (see Chapter 4). In such a way, students can review important concepts on their own, anywhere and at any time. At the same time, when they have to solve exercises, they can write down their observations or justify their decisions with fellow students and teachers using the EasyStatics functionalities which allow them to generate copies, compare situations, write comments, produce state-files or save pictures and embed them in the built-in HTML editor. Then, using the EasyStatics teaching platform they can send the generated EasyStatics files or any related document to their teachers or use them in discussion with other students. One example of the collaborative use of EasyStatics can be found in Chapter 6, when a practical experiment carried out by students is described.

### 3.6 Programming Challenges

The current state of EasyStatics is the result of an iterative development process which has led four more or less year. At the beginning the main questions concerned how the program should look to its user and which functionalities had to be implemented. However, when the time to effectively implement them came, in some cases, difficulties arose and alternative ways had to be found. The initial considerations as well as some problems encountered and their solutions are described below.

**Choice of the programming language:** The programming language had to meet the following requirements:

- the software should be platform independent. No student should be discriminated, regardless which operating systems she or he is using, be it MS Windows, Macintosh OS, Unix, Linux, etc.;
• the programming language should be modern, with a good documentation and rich libraries;

• the software should eventually run as Web application, e.g. as an applet\footnote{An applet is an application which can be downloaded from a Web server and run on a computer by a Web browser, such as Netscape Navigator or Microsoft Internet Explorer.} on the Web.

Considering these requirements, the Java programming language, an object-oriented language similar to C++, was chosen. Java source code files are compiled into a format called byte code, which is then executed by a Java interpreter. Compiled Java code can run on most computers as Java interpreters and runtime environments, known as Java Virtual Machines, exist for most operating systems, including UNIX, the Macintosh OS, and Windows. A drawback of interpreted programs is that they generally run slower than compiled ones. In the case of EasyStatics, this was not a matter of concern because the program was designed for simple structures only. A time performance problem was encountered only when the program had to run as an applet. This reacted too slowly, which was against the request for fast response times. For this reason, the idea of having the program also as an applet was abandoned. Nevertheless, EasyStatics was designed to communicate with the Internet and to exchange data with the server through the use of a special java package. This allows one to directly open EasyStatics files from the server and to store them on it.

Operations on the graphical objects. In order to make the program easy to use and intuitive as possible, it was initially designed to allow one to perform certain operations on different graphical objects without distinguishing the kind of object dealt with\footnote{These operations regard, for example, the moving and deleting of graphical objects, which can be hinges, loads, supports, elements and nodes.}. This was only possible if the graphical objects were not too close to each others. Whenever this was the case, the latest drawn graphical object would be selected, which is not necessarily the desired one. Due to this inconvenience, it was preferred to abandon the idea of generality and to allow operations on a graphical object only if the corresponding button on the left of the screen (see Appendix Section A.1) is selected.

Also for moving graphical objects along the structural model it was not reasonable to have all of them behaving the same way. Therefore, some limitations were necessary. For example, a concentrated load or a hinge can move from one element to another, whereas this is not allowed with supports and distributed loads. Supports are always associated with a node and moving them is possible only by moving the corresponding node. It was also decided to restrict the movement of a distributed load only to the associated element. In fact, if such a load could move from one element to another, problems would arise when several elements converge in the same node as it would be difficult to know on which element one wants the load to move. Furthermore, if the load would encounter another distributed load, the handling would be complicated. For this reason it was decided to limit the mobility of the distributed loads and also to have only one distributed load on an element.

Load scale definition. In order to get a feeling for the loads acting on a structure and
relate them to the results, the loads are drawn proportional to their value. However, with the default scale given at the program start, the value of a load could be too large, in the second case too small. In the first case, the length of the associated graphical objects could exceed the screen dimensions or become invisibly small. To avoid that, one is allowed to redefine the load scale by entering in a text field the load value on which the scale has to be computed.

**Representation of the support reactions.** For giving a feel of force equilibrium also the support reactions should be drawn proportional to their value. This could lead to graphical objects too large or too small. Having this permanently displayed on the screen would, in some cases, disturb the user. For this reason it was decided to introduce a “result” button (see Appendix Section A.4), which, if selected, causes the reactions to be proportionally shown in the direction of the resultant. If the button is deactivated only the support directions are shown.

**Subdivision of one element into more elements.** Allowing one to subdivide an element by introducing more nodes is not straightforward. In fact, each node introduction requires the global matrix to be reassembled considering the new topology, i.e. assembling the local matrices of all old and new elements. The characteristics of the original element, i.e. material and section properties and, if present, loads and hinges, should be passed to the new ones. Although these operations are relatively complex, they are necessary, e.g. when one wants to observe the effect of element subdivisions in the case of 2nd order, plastic or dynamic analysis. This allows also more flexibility in defining the structure’s geometry.

**Results representation.** Displacement, bending moment, shear and normal force diagrams are represented in *EasyStatics* using a scale computed on a certain number of pixels representing the maximum value shown. This number is defined as a constant in the program, which depends on the actual screen resolution but cannot be changed by the user. Shear and normal forces were originally drawn using the scale of the concentrated loads. Although this was instructive on simply loaded structures, in many cases the related diagrams became so large that some parts of the model were covered. For this reason also these diagrams were represented with an independent but equal scale based on a predefined number of pixels.

**Saving *EasyStatics* files.** For saving a complete *EasyStatics* model description one possibility was to generate a text file containing all the relevant data to be parsed lately, when the file was reopened. This, however, would have been quite complicated. The preferred way to save all the data was by *object serialization*, which is the process of saving an object’s state to a sequence of bytes, as well as the process of rebuilding those bytes into a live object.

**Saving the graphic layout.** This information is also saved and read in serialized form in a *EasyStatics* file. However, when such a file is reopened one can reactivate the missing button or, more generally, choose his own setting.

43
Chapter 4

*EasyStatics* in the context of E-Learning

The program *EasyStatics* is embedded in the *EasyStatics* e-Learning environment, which can be reached at the address [www.easystatics.ethz.ch](http://www.easystatics.ethz.ch). In order to understand the criteria followed during its implementation, a definition of the e-learning concept is first given. Then, commonly used e-Learning platforms are analyzed, considering their advantages and limits.
4.1 The Value of E-Learning

In its broadest definition, e-learning includes instruction delivered via electronic media including the Internet, audio/video tape, interactive TV and CR-ROM. These technologies have caused dramatic changes in the learning and teaching process, also strongly influenced by the World Wide Web, which eliminates the dependency on time and space. However, the real value of e-learning lies not only in its ability to train just anyone, anytime and anywhere, but also in the ability to train people to gain the right competencies when they are needed. There are efforts to move e-learning towards the total automation of administration of the teaching and learning process by means of software generally known as Learning Management Systems (LMS).

In order to be effective, a LMS, must be based on sound pedagogical principles. Unfortunately, there has often been a mismatch between the abundance of features in an e-learning platform and sufficient explanations on the pedagogy underlying the inclusion of these tools. Also lacking are guidelines on how to design, develop, deliver, and manage pedagogically sound e-learning materials. Therefore, in the context of e-learning, methodology rather than technology is what really counts. This implies that the following phases of the instructional process have to be considered when developing e-learning courses: analysis of the needs, design of the course, delivery of didactic material, understanding the content and assessing the acquired competencies. Only a precise analysis of these aspects allows one to find out [61]:

- which didactic materials have to be delivered through the Web and which aspects should be treated in the traditional classroom;
- which e-learning systems have to be included in a LMS in order to substitute the outdated components with the updated ones;
- which support service should be set up in order to adequately prepare tutors and online experts to help students during their learning process;
- how can the learning progress of students be assessed effectively?

4.2 E-Learning Platforms

Technically, an e-learning platform can be described as software installed on a Web Server which communicates with a database and provides several functions for instructors and students [62]. An e-learning platform can also be considered as a container which provides a set of tools able to manage every aspect of the users’ administration needs (students, tutors and teachers), the didactic materials and the course realization.

As stated earlier, in order to be successful an e-learning platform should be rooted in strong pedagogical foundations. However, it should be clear which purposes it serves. This means that it should be considered if an e-learning platform is used to
provide distance courses, as a support to the face-to-face class, in blended mode or simply as an additional tool for a traditional course. Furthermore, when choosing an internet platform, the technical competencies of the people administering it should also be considered. If these have little computer knowledge an e-learning platform with elementary functionality is the preferred choice. Otherwise, an e-learning platform with more complex functionality can be used, which is more flexible (see Subsection 4.2.2).

4.2.1 Characteristic Attributes of E-Learning Platforms

Pedagogically desirable attributes will be discussed using the following five parameters: content development and course structure definition, delivering content, data elaboration, assessment, students and classes organization [62].

Content development and course structure definition. The content development process must follow a systematic instructional development methodology. Using a systematic approach to the content development ascertains the congruence of the learning material with the predetermined learning objectives and influences the structure of the on-line course. This approach consists of the following steps:

- Analysis of the background of the learners, particularly their academic performance.
- Analysis of the task for determining the level of detail and the depth of content. This implies making a list of general topics to be covered by instruction, to outline the course content and to identify the task that learners should be able to perform.
- Defining objectives by identifying terminal objectives, intermediate objectives, enabling objectives and analyzing these instructional objectives to identify types of learning involved, i.e. collaborative learning, situated learning, individual learning, etc.
- Selecting instructional strategies which implies selecting instructional activities and media elements.
- Preparation of preliminary draft material by constructing a concept map, possibly developing and validating course evaluation questionnaires.
- Formative evaluation by gathering information about weaknesses in the material. This is made principally by revising instructional material, by selecting a representative group of students (e.g. three learners, one high achiever, one average learner and one low achiever) and observing how they interact with the instructional product and respond to the course evaluation questionnaire.
- Production of the first version of the e-learning module.

This methodology appears to be linear but it is actually iterative in practice. Content developers will often find themselves moving back and forth repeatedly between functions.
Sometimes plans are finalized for one function but after moving on to the next function, earlier plans have to be modified.

An emerging tendency in the design of e-learning platforms is to organize the pedagogically relevant contents in smaller, manageable chunks known as learning objects (LO). The main advantage of producing learning content in this way is the flexibility it offers. Once a collection of objects has been produced these can be re-used in different combinations to construct different learning experiences. Re-using existing learning objects allows everyone to develop professional material with little effort reducing costs and saving development time.

Learning Objects can be regarded as e-learning’s equivalent to traditional classroom lessons. Often, they are organized in several smaller components. If learners do not achieve the minimum scores stipulated by the LO, then they will be directed to a more basic LO in order to acquire all the knowledge required to master the current LO. Once learners have reached the minimum requirements, they can solve a related practical exercise. This is finally followed by a post-test that assesses the learner’s mastery of the LO’s knowledge. Then the learner is directed to the next LO in sequence [61].

**Delivering Content.** Learners and instructors should be allowed to access and offer content in the form of bare LOs ‘just in time learning’, which may be the best option to bridge specific performance or knowledge gaps. In such a way, the students are not forced to study in a linear way. But the easiest system is to allow the students to skip information included in a lesson, in order to move to the tests. In some cases, after the results of specific preliminary tests, lectures can be presented in a different way, for example, supplying only some parts and eliminating those already covered by the student. The most complex platforms can determine the lecturer’s assignment following the student’s individual needs (language, experience, etc.). Students interact with the e-learning platform only when they connect to it for accessing the didactic material. The other functions, in fact, are more the task of the teachers. Generally, a student can access his or her assessment tests, lectures already worked through, lectures still to be read, homework to do, etc. Normally, the platform provides students with communication tools, such as mails and chats in order to exchange messages with teachers and other students of the same class [62].

**Data Elaboration.** A platform should give the possibility to elaborate the collected data in statistical reports. In order to do this, the system should register the number of completed, passed and failed lessons, the time devoted to each lesson, the didactic objectives reached or failed, the beginning or ending time of a lesson, etc. This function has two goals. On one hand, it allows one to observe the efficiency of the students; on the other, it allows one to evaluate the efficacy of the course and eventually to correct design defects [62].

**Assessment.** Most e-learning platforms include test builder tools that automate the process of authoring questions. Most of these tools offer easy-to-use templates for
Chapter 4. *EasyStatics* in the context of E-Learning

authoring automatically scored questions. Unfortunately, these tools do not include other types of questions like projects, structured subjective questions, and case studies, which are also important for a valid and reliable assessment of the learners. In order to assess them by means of these latter types of questions, instructors have to post the message on a bulletin board. Students then complete their assignments and submit their work to the instructors via e-mail or upload it as a Web page for the instructor to assess manually. [61].

*Students and classes organization.* Most e-learning platforms allow students to enter one or more on-line courses. In some cases, students are registered directly by their teachers. Generally, teachers fill in the first and last name of their students through a tailored tool, which is contained in the e-learning platform. Then, an automated process generates passwords and usernames, which are immediately sent via e-mail to each student. Then, a registered user can change his or her password once logged on the e-learning platform. If students are not registered by their teachers, e-learning platforms allow them to register themselves directly after entering the platform as a guest [61].

4.2.2 Easy to Use and Flexible Platforms

Today, there is a large number of e-learning platforms to choose from for setting up an e-learning course. Our evaluation of different Internet platforms has shown that a single “best” platform does not exist. Every product does have its specific strength, but is lacking in other aspects. A direct comparison is therefore nearly impossible. However, an interesting result is that the products can be naturally divided into two main classes: those with only elementary functionality, which are simple to use, and those with complex functionalities, which are more flexible and extensible.

*Platforms with elementary functionalities,* among which there are, for example, the commercial platforms WebCT and Blackboards, are meant for course designers who need to be able to efficiently create courses with just a basic knowledge of Web technology. No programming or advanced Web-authoring skills are required. These platforms have the drawback that it is difficult or even impossible to integrate ambitious interactive and dynamic learning content into them. In addition, there is no possibility for the user to modify the source code in order to correct mistakes or add functionalities. Another disadvantage of the use of commercial platforms is that their cost can be considerable.

*Platforms with complex functionalities,* on the other hand, can be customized in terms of presentation and functionality. They have a fully accessible and documented programming interface, they allow the addition of new functionality to the server and they give full layout and navigation control to the course designers. However, they require skilled course designers with knowledge and experience in programming, Web data standards and communication protocols. A minimum of four weeks of introductory training should be allocated for an IT-specialist to get along with such a platform. Examples of extensible platforms are the open-source platforms OLAT (ETH Zurich) and Caroline (MIT, USA).

It has also to be said, that often the choice between commercial and open-source platforms
is a choice between operating systems (Linux vs. Windows).

### 4.2.3 Choosing the Most Adequate E-Learning Platform

Before choosing or implementing an e-learning platform, Leonardo Borselli \(^1\) suggests that one considers the following ten questions:

1. Will the platform be used to provide courses completely via Web or as a support to the face-to-face class, in blended mode or simply as an additional tool for a traditional course? In the first case the commercial platforms are more adequate. If the answer is that the platform will be used for a blended mode, then the platform should be flexible, like the open source platforms. They should have the possibility to embed any kind of tool in order to fulfil a certain need during the course whereas less important is the monitoring of activities, because a test will take place at the end. Finally, if the platform may be used only as a communication tool between teachers and students, this must be simple for both to use.

2. Who will manage the system? Are there competent people who are able to manage the platform? If the answer is negative, then it is necessary use a commercial platform otherwise, if somebody is able to customize the environment, an open source platform can be used.

3. How many students are there? The commercial platforms are usual for a large number of students whereas the open source ones are suited to a smaller number of student.

4. Should the platform monitor the didactic activities made by students with absolute certainty or some tolerance is admitted? In the first case, a platform is needed which includes this feature. In the second case the services provided by a normal log of any web server can be used.

5. What is the security level? Is the risk that external users can enter the online course acceptable? Admitting that complete security is impossible, it is necessary that the platform provides security services which are a compromise between a certain level of security and a major system access.

6. When choosing a platform, does one use the available functionalities or favour the real needs of the users? A platform with too many features is less intuitive. This could discourage its use and a user, probably, will not try to use the additional features.

7. Do students work in computer environments with the same or with different configurations? In this case, some platforms could have features which do not allow the use of some functionalities with some configurations. This is often the case with

---

\(^1\)Responsible of the “Nucleo Informatico e Telematico - Dipartimento di Ing. Civile, Universit degli Studi di Firenze”.
platforms requiring a particular browser or different plug ins. If the aim is to have a
platform suited to a heterogeneous environment, for example when student are con-
ected from home, this aspect should be primarily considered during the material
preparation.

8. Do the users use the same language? Are there sensorially disabled users? In this
case the platform should be customized for the different purposes. Generally, all
the platforms allow this but the difficulty and cost level for the adaptability are
different.

9. Are there users with a reduced bandwidth? In this case, it is necessary to provide
alternative modes to access and select the content. Therefore, it should be allowed
to use the platform without pictures and multimedia contents.

10. Is it necessary that the platform accepts documents of any kind or only with a
defined format? Should the system follow a standard for the storing the documents?
Should the platform allow search functionalities? If these requirements are made,
for an open sources platform this is a marginal but solvable problem, whereas in
the case of commercial platforms the needed feature is announced but it could be
provided too late.

4.2.4 The Lifespan of E-Learning Platforms

E-learning platforms, like computers, quickly become outdated. Thus, the platform
should be updated and sometimes old files must be converted. A platform may last two
years before the next version is needed. Each new version generally introduces more
errors than those eliminated, with the result that the program becomes more complex
and, paradoxically, slower and less compatible with the old files. In addition, users are
often required to learn more functions and new locations of the menus. Therefore, if
on one hand commercial platforms change rapidly, on the other the effort needed to
understand how they work is considerable. This is the reason why a new platform should
always provide substantial benefit.

In order to find a solution to the problems already mentioned, integrated solu-
tions, i.e. organization of services tailored to the specific needs rather than to the
abundance of available technological tools should be implemented. Obviously, in order to
create an adequate e-learning platform, these services should consider all the requirements
reported in this section.

4.2.5 Basic Components of an E-Learning Platform

In order to create the integrated services which are expected from an e-learning platform
the following three essential components are needed:

- a Web Server which can be accessed by all users connected to the Internet;
• a database for storing and managing different types of information. This information can regard users (e.g. their password, username, address, etc.), messages among users, forums, chats, questionnaires, documents and exercises related to the lectures, points scored by each students in a exercise, etc.;

• an interface which allows communication between the Web Server and the database and between the client and the Web Server.

Web Server

The *EasyStatics* e-learning platform is based on the Apache Web Server, which is open-source and run on the most used operating systems (Windows, Unix, MacOs, Linux). A Web Server follows a strict client-server architecture and its function is to provides access to distributed documents. For the communication between the client (browser) and the Web server the Hypertext Transfer Protocol (HTTP) is used. The most common format to present information on the Web is the HyperText Markup Language (HTML), a standardize language for creating hypertext documents which are also called static HTML pages. Additionally to such documents, Web servers can as well provide access to other information sources such as DBMSs (Database Management Systems). For this purpose the Web server communicates with application programs via the Common Gateway Interface (CGI). This mechanism allows clients to submit queries that in turn are processed at the server site.

Database on the Web

A database on the Web follows also the client/server architecture. The server stores data and manages the accesses to them whereas the clients connect to the server and access data. A database includes a system of files, which are structured in a very efficient way, in order to retrieve the needed information in a short time. These files can be shared among many users.

The *EasyStatics* e-learning platform is built on a *MySql* database, which is a Relational Database Management System (RDBMS) based on tables. The rows of these tables represent relations which are identified through a key value.

Common Gateway Interface

The oldest and probably most widely used approach to access databases from the Web is based on the Common Gateway Interface (CGI). CGI is a standard that allows Web servers to access arbitrary external applications, so called CGI-scripts, which can be written in different languages, such as *PHP*, *Perl*, *C*, etc. - in the *EasyStatics* e-learning platform the CGI scripts are written in Perl. A CGI script is executed at the server site. It is started by the Web server upon client request. The CGI script receives the parameters supplied by the user and hidden variables, generates HTML pages
dynamically (e.g., on the basis retrieved from a database by the CGI script), and delivers them to the Web server. The Web server then transfers the generated HTML pages to the WWW client.

In order to understand the relation between clients, Web server, CGI and database, the figure 4.2 is explanatory.

![Diagram of Web Server, Database, CGI and Static HTML pages.]

Figure 4.2: Use of Web Server, Database, CGI and Static HTML pages.

### 4.3 EasyStatics E-Learning Platform

In the course on “Computer Statics” at the Civil Engineering Department of the ETH we have used EasyStatics intensively as an educational tool for teaching structural analysis and design. In order to allow students and teachers to share EasyStatics documents, exercises and information we created an Internet environment which embeds also the EasyStatics program.

According to the considerations written in the previous sections, before developing the e-learning platform, we considered which services it should provide, the characteristics of the class going to use it and our technical competencies in administering it.

Our goal was not to generate a complete on-line course but to provide an additional instructional tool to enrich the face-to-face class so as to help students access information outside the classroom, anywhere and at anytime. It was assumed that teachers could upload the necessary documents on the Internet (theory documents, EasyStatics demonstrations and exercises), download the exercises made by their students and communicate with them through an internal messaging system.
Students had to work with the *EasyStatics* program and then access the Internet only for downloading and uploading material and for writing or reading messages to and from their teacher or other students of the same group. Regarding the assessment of students’ learning progress, we believed that this could not be made by means of multiple choice, which is often used with e-learning platforms. We were more confident that a valuable assessment could be obtained by reading the comments written by students in an *EasyStatics* file or by directly interviewing them. This choice was possible mainly because of the small number of students attending the course (around 35) and because of the possibility that the assistants could monitor them directly. This is also the reason why we did not want the Internet platform to provide systems for monitoring the students’ activities (see Section 4.2).

Considering the technical competencies for managing the e-learning platform, we were confident that we had the necessary know-how to implement in a short time an environment tailored to the needs already mentioned. For this reason, the possibility of using a commercial platform was ruled out. In fact, as previously mentioned, this is expensive and has many functionalities we do not need. Alternatively, we could choose an open source platform. However, this has the disadvantage that, in order to have a personalized environment with only the functionalities tailored to our needs, its source code needs to be edited. Therefore, the required time for both learning languages such as *MySql, PHP, Perl*, etc. and for understanding the source code written by other people (not an easy task!) is considerable.

The concept on which the *EasyStatics* platform is based can be seen in Figure 4.3.

![Figure 4.3: Relationship between the Server and the Users.](image-url)
4.3.1 Files Managed by the *EasyStatics* E-Learning Platform

Three different kinds of file are stored in the database:

- **EasyStatics files**: files with suffix “.est” are executable *EasyStatics* files. They include one or more computational states with the related comments (see Section A.17).

- **Documents**: these refer to lecture brochures in general, like MS-Word, PDF, Power Point files, pictures, etc. Obviously, in order to open these documents students must have the necessary software installed on their computers. HTML reports also belong to the documents that can be created with the HTML editor (*SimplyHTML*) embedded in the *EasyStatics* program (see Section A.19).

- **Messages**: these are text files similar to e-mails, which can be sent between the members of the same group.

4.3.2 Users and Users’ Rights

The e-learning platform distinguishes between three kinds of users: teachers and their assistants, students and guests. A group is composed of a teacher (including teaching assistants) and his or her students. A special role is played by the administrator of the database.

**Database administrator**

The administrator is responsible for the establishment and organization of the database on a server. He or she can create new groups and register the teachers of each group, assigning them a user name and a password. Several groups can be accommodated on the same server. The online publication of freely available *EasyStatics* demonstrations, which is available to all users including guests, is done by the administrator.

**Teachers**

A teacher can

- register teaching assistants in the database and delete them. Registered assistants then have all the teacher rights listed below;

- store *EasyStatics* state-files and document files on the server;

- make *EasyStatics* state-files both visible (thus downloadable) and invisible to his or her students;

- download *EasyStatics* state-files sent by students and assistants;
Chapter 4. *EasyStatics* in the context of E-Learning

- send messages to the whole group, which appear immediately as soon one enters the *EasyStatics* platform, or to a single member of the group;

- read student messages and

- delete all types of messages.

**Students**

A student can

- download *EasyStatics* state-files to his or her computer and execute them locally. Furthermore, because *EasyStatics* can connect with the server directly, i.e. without going through the Internet portal, a student can open an *EasyStatics* state-file sent individually to him or made available by the teacher to the whole group, during program execution (see Section A.20);

- download to his or her computer the document files available to the group;

- send edited *EasyStatics* state-files (e.g. solution of exercises) to the teacher or to another student (this can also be done directly while using *EasyStatics*);

- delete his or her own *EasyStatics* state-files and HTML reports;

- send messages to his or her teacher, assistants and other students of the same group;

- read messages, which were sent either personally to him or her or to the whole group and

- delete personal messages from the server.

**Guests**

A guest can only install the program *EasyStatics* locally, download *EasyStatics* demonstration state-files and execute them.

### 4.3.3 User Interface

The user interface of the *EasyStatics* platform was designed to be as easy to use and as intuitive as possible. This means that, when a student or a teacher enters it, they should immediately see the program’s functionality and understand which menu to select in order to perform a certain operation.

When the *EasyStatics* home page is shown (see Figure 4.1), users can decide whether they want to work in English or German.
Chapter 4. *EasyStatics* in the context of E-Learning

On the top of the home page there are labels embedded in a horizontal toolbar, which represent links to other HTML pages and are always visible. These labels are:

- *About EasyStatics* for reading information related to the *EasyStatics* project.
- *Examples* for downloading *EasyStatics* demonstration files.
- *Installation* for downloading the *EasyStatics* program.
- *News* for any information about events regarding *EasyStatics*.
- *Contacts* for reaching the people involved in the *EasyStatics* project.

The central area of the home page is reserved for the login area where students or teachers input their password and username in order to enter the course in which they are enrolled. Immediately below the login window, there is an area specifically for guests, which allows them to freely access the *EasyStatics* demonstration files.

### 4.3.4 Welcome Page for Registered Users

![Welcome page](image)

**Figure 4.4: Teacher welcome page**

When registered teachers and students have correctly written their password and username, they access the *welcome page* of their online course. In the centre of the page there is a list of messages sent by the teacher to the whole group (see Figure 4.4). The messages are chronologically ordered and identified by a subject. Clicking on the “read” button the related text appears.

On the left (see Figure 4.4), a list of menus appears, which is always visible from any location of the Internet platform. The number of menus varies, depending on whether the user is a student or a teacher.
4.3.5 Teacher Environment

When teachers access their online course, they can perform different operations by selecting the menus on the left (see Figure 4.4). These menus together with the associated operations are described below.

- **Enter users.** A teacher can enter students by writing their first and last name in the reserved fields or sending a text file containing a list of the students’ first and last names to the database. This file is then parsed by a CGI script, which enters all the users into the database assigning them a password and a username.

- **Users list.** A list of the registered students with their first name, last name, username and password appears. Each entry of the list can be deleted individually.

- **Theory files.** A list of downloadable theory files sent by a teacher to the whole group and chronologically ordered is shown. Each file can be deleted and made visible or invisible to students.

- **EasyStatics files.** Three separate lists of downloadable EasyStatics files are visible, which are chronologically ordered and identified by the file name. In the first list there are the EasyStatics demonstration files provided by the administrator. The second list contains the files sent by a teacher to the whole group. These files can be demonstrations shown in the face-to-face class or exercises. Like the theory files, they can be deleted and made visible or invisible. In the third and last list, there are personal files sent by students. The username of the sender appears beside the name of the file. These files can be deleted by the teacher.

- **HTML files.** The same as above.

- **Read messages** Messages appear subdivided in two parts and chronologically ordered. In the first part, one finds the group messages. In the second part there are the messages sent by the students of the group. Each message is identified by the username of the sender and by the subject. Each message can be deleted.

- **Write messages** An e-mail-like window appears where a teacher can write the subject and the text of the message and select to whom to deliver it: a message can be sent to the whole group or to a particular student selecting him or her in the list containing all the usernames.

- **Upload files.** The page is subdivided into more sections, corresponding to the different types of file to upload. The first section is reserved to the uploading of theory files whereas the second and third section are for uploading EasyStatics and HTML files, respectively. In this case, teachers can decide either to send an EasyStatics or HTML file to the whole group or to an individual student selecting him or her in the list of all the usernames. In the case of HTML files, because they may contain links to pictures, these are to be uploaded separately. With the CGI script they are automatically placed in the right folder in the server.
4.3.6 Student Environment

As mentioned above, the operations a student can perform when using the *EasyStatics* platform are fewer than those performed by a teacher, therefore, the number of menus is reduced in this case. These menus are:

- **Theory files.** A list of downloadable chronologically ordered theory files sent by a teacher to the whole group.

- **EasyStatics files.** The same as in Section 4.3.5. The only difference is that students cannot delete the group files.

- **HTML files.** Same as above.

- **Read messages** The same as in Section 4.3.5. The only difference is that students cannot delete group messages.

- **Upload files.** The same as in Section 4.3.5. Differences are that students cannot upload theory files and cannot send files to the whole group.

4.3.7 Database Structure

The administration of files and users as described in Section 4.3.1 and Section 4.3.2 together with all the operations one can perform when using the Internet environment are based on the *MySql* database (see Section 4.2.5). This includes the following tables:

- **Groups**, which contains only one kind of information, i.e. the name of the group (*GroupName*).

- **Users**, which contains information related to the users, i.e. the user name (*UserName*); the name of the group (*UserGroup*); the password (*Password*); the permission (*Permission*), which is different depending on whether the user is the administrator (*root*), a teacher or a student; and the user last and fist names (*LastName, FirstName*).

- **ExecutionStates**, which contains information related to *EasyStatics* state-files, i.e. the identification number (*ID*), the name of the file (*StatesFileName*); the name of the sender (*WriterName*); the name of the receiver (*ReceiverName*); the name of the group to which the file belongs (*GroupName*); the server location where it is stored (*StatesFilePath*); the accessibility (*Access*), which says if the file is accessible by everybody (*public*), by all the members of the group (*group*) or if it is sent to a particular person (*personal*); the activation state (*Active: yes or no*); and the time at which the file was stored (*CreationTime*).

- **Documents**, which contains information related to all kinds of documents including *EasyStatics* pictures. The same explanation used for the *ExecutionStates* is valid here.
• **Information**, which contains information related to the theory files provided by the teacher to his students, i.e. the identification number (ID); the name of the file (FileName); the description of the file (FileDescription); the name of the group to which the file belongs (GroupName); the server location where the file is stored (FilePath); the activation state (Active: yes or no); and the time at which the file was sent (CreationTime).

• **GroupMessages**, which contains information related to the messages a teacher writes for the whole group, i.e. the identification number (ID); the name of the sender (SenderName); the name of the group sending the message (SenderGroup); the text of the message (MessageText); and the time at which the message was sent (CreationTime).

• **Messages**, which contains information related to the messages a user writes to another user of the same group, i.e. the identification number (ID); the name of the sender (WriterName); the name of the receiver (ReceiverName); the name of the group to which the message belongs (GroupName); the text of the message (MessageText); and the time at which the message was sent (CreationTime).

These objects can be represented in tables as shown in Figure 4.5.

![Database structure](image.png)

Figure 4.5: Database structure

The relations existing between these tables (which are identified by a line connecting the tables) can be described as follows.
• A document containing EasyStatics pictures or an EasyStatics file can be received by many users and a user can receive many documents or EasyStatics files, therefore, the relations “is received by” between Documents and Users or ExecutionStates and Users are both N:N relations.

• A document or an EasyStatics file can be sent only by a user but a user can send many documents or EasyStatics files. Thus, the relation “is sent by” between Documents and Users or ExecutionStates and Users are both N:1 relations.

• A message can be received by many users and a user can receive many messages. Therefore, the relation “is received by” between Messages and Users is an N:N relation.

• A message can be sent only by a user but a user can write many messages. Therefore, the relation “is sent by” between Messages and Users is an N:1 relation.

4.3.8 Course Preparation

In general, a course is based on several lectures, each of them comprising some theoretical background accompanied by EasyStatics demonstrations and exercises.

Distribution of document files

Document files stored on the server are available to all members of the group through the e-learning platform. These contain all kinds of theoretical or complementary information for a lecture or exercise. They are created using either an arbitrary program such as MS-Word, PowerPoint, Acrobat Reader or with the HTML editor embedded in the program.

Preparation of EasyStatics demonstrations and homework

The teacher defines a problem using the editor embedded in the EasyStatics program. As soon as the exercise has been prepared, it can be sent to the server by selecting the menu item save document on the server. In general, EasyStatics demonstration files and exercises are made available to the whole group or, when required, to an individual student. This can be done both through the e-learning platform or directly from the EasyStatics program, which allows a connection with the server without passing through the e-learning platform (see Appendix Section A.20).

All the students of the group can open the exercise directly from the program by selecting the menu item open from server (see Section A.20) or downloading it locally from the e-learning platform and opening it later with the program.
Chapter 4. *EasyStatics* in the context of E-Learning

**Solving *EasyStatics* homework**

Under the menu *EasyStatics* state-files, students have a list of all the *EasyStatics* state-files which have been sent to the whole group or just to them personally. They can open a selected *EasyStatics* file directly from the program by selecting the menu item open from server (see Section A.20) or downloading it locally to their computer and opening it later in the program. Using the program, students can edit the exercise, add their own comments and delete or create new computational states. When the exercise is completed, it can be sent directly to the teacher, tutor or fellow students by selecting the menu item send to server and choosing the recipient from a list of all group members (see Section A.20). Alternatively, they can save the exercise locally and send it to the server through the e-learning platform (Menu send files). Students can also document the completion of their homework by producing an HTML file using the internal editor. The HTML document, possibly containing pictures extracted from the program *EasyStatics*, can be saved locally and then sent directly to the server by selecting the menu item save document on server and then proceeding in the same way as in the case of *EasyStatics* state-files. Otherwise, it can be sent to the server via the Internet portal (menu send files).

**4.3.9 Experience Made**

At the end of the course the students were asked to express their opinion regarding the *EasyStatics* e-learning platform. Their experience with it was generally positive. They found the platform easy to use and intuitive. In particular, they appreciated the possibility of accessing information anywhere and at any time. They connected to the platform mostly for uploading and downloading files whereas they hardly used the internal messaging systems. This is probably because, when they wanted to communicate with their teaching assistants or their teacher they went directly to them or used their personal e-mail system.

As instructors, we experienced the advantages of using the *EasyStatics* e-learning platform as a complement of the traditional face-to-face class. However, we found also some limitations. One of them concerned the organization of the students’ files on the e-learning platform. In fact, these were grouped all together in a list and when their number became considerably large, it was difficult for us to manage them. Also time-consuming was erasing students’ files from the e-learning platform. This because each file needed to be deleted individually. Another disadvantage was that many students preferred to communicate through their personal e-mail system and only few of them used the messaging system of the e-learning platform. Consequently, we had to check both the personal e-mails and the messages on the e-learning platform, which caused the communication to be inefficient.
Chapter 5

*EasyStatics* as a Tool for Teaching

In order to be a tool for learning structural design the *EasyStatics* program has to help students answer the following questions:

- what are the elastic deformations of the structure caused by the loads?
- what are the section forces associated with these deformations?
- what are the stresses caused by these section forces?
- Can the bars of the structure with their assigned sections withstand these section forces?
- What is the influence of 2nd order effects on structural behaviour and how great can compressive normal forces become before buckling occurs?
- in which cases can time-dependent loads lead to dangerous dynamic effects?
- how far can external loads be increased without the structure’s collapsing and how would the structure collapse if the loads reach this limit?

To answer to these questions mechanical models, which approximate physical reality, are needed. The FEM is ideally suited both to build such models and to analyze them numerically.

In the following sections, after considering the main principles of the FEM, it is shown how the program *EasyStatics* can be used to set up exercises and demonstrations related to the different analysis types. Their theoretical aspects will be briefly explained before discussing related exercises¹.

It should be clearly stated, however, that *EasyStatics* is not meant to be used for understanding these theoretical and computational aspects. If these are mentioned here is only because we consider them as essential for all students willing to use modern software instruments for structural design. In fact, the theoretical background found in the

¹For more detailed explanations about the theory behind the analysis types, see [63]
next sections, which is basic, straight forward and also quite easy to understand as many details are left out, represents the absolute minimum which any student - in our opinion - must understand. Obviously, this requires adequate teaching, independently of EasyStatics.

5.1 Basics of The Finite Element Method

In the linear elastic case the FEM works as follows:

1. The structure to be analyzed is first subdivided into finite elements of simple geometric shape, e.g. straight prismatic bars. The elements are joined to each other or supported at the nodes.

2. At each node displacement parameters are introduced which represent the primary unknowns to be numerically determined. In the case of plane frames two displacements in the horizontal and vertical directions and the nodal rotation are considered, whereas in the case of plane trusses only the rotation is left out.

3. In order to determine these unknowns, nodal equilibrium conditions are formulated as linear equations. Nodal equilibrium is satisfied when the resultants of all the forces in the horizontal direction, of all forces in the vertical direction and, in the case of plane frames, of all moments acting on each node vanish.

4. The fundamental idea of the FEM is to determine the numerical coefficients needed to formulate nodal equilibrium conditions by considering each element separately. These coefficients are first “locally” determined for each single finite element taking into account its geometry, material properties and loads.

5. From these local element coefficients a “global” equation system is then assembled whose equations represent nodal equilibrium conditions for the unknown nodal displacement parameters.

6. This global system of linear equations is solved numerically so as to determine all nodal displacements. In two cases (see Sections 5.3 and 5.4) eigenvalue problems, which arise from the formulation of the same kind of nodal equilibrium conditions, are to be solved.

7. Once the nodal displacements are known, secondary quantities, like section forces or support reactions, are determined.

In the rigid-plastic case, as discussed in section 5.5, the steps 1 to 5 are similar, but instead of a system of linear equations, a maximum or minimum problem whose variables are subjected to linear inequality constraints is obtained. This is called a Linear Program (see [63] for more details.).

The problems of finding the unknowns of a system of linear equations, computing eigen-modes or linear programming optimization are, nowadays, of little interest for students
attending courses on structural design. In fact, only specialists need to be concerned with them. However, as future users of FE programs, students must understand the meaning of these problems and their relation to structural mechanics and the FEM.

While using *EasyStatics*, students have to be aware that only in the case of statically loaded plane frames and trusses by 1st order theory the FEM provides exact results. In fact, the FEM is, in most cases, an “approximate” numerical method which provides “exact” results only when using an infinite number of elements. However, by refining the element subdivision, satisfactory convergence to the theoretically exact results is easily obtained. In the following sections exercises will be suggested where by subdividing the original elements into shorter ones more realistic results are obtained (see Sections 5.3.6, 5.4.3 and 5.5.5).

### 5.2 Static Case - 1st Order Theory

Traditionally, students beginning a structural analysis course first deal with statically loaded structures analyzed by the linear elastic 1st order theory. In fact, this is fundamental to understand all the other analysis types explained later.

According to this theory, a linear elastic material behaviour under slight deformations is considered. It is also assumed that the normal forces which in the undeformed reference configuration might already be present in the structure have no influence on its mechanical behaviour (see Section 5.3).

#### 5.2.1 Global and Local System

As mentioned before, the displacements of the loaded structure are the primary unknowns to be numerically determined. For this, a linear system of equations has to be solved, which is built by formulating nodal equilibrium conditions. For plane frames the two forces resultants in the horizontal and vertical directions and one moment resultant are considered at each node. For trusses the moment resultants are not needed.

Each of these resultants $R_n$ is expressed as a linear function of the nodal displacements $U_m$ and of the loads as follows:

$$R_n = - \sum m K_{nm} U_m + F_n$$

or, in matrix notation

$$\mathbf{R} = - \mathbf{KU} + \mathbf{F}$$

where the global system coefficients, $K_{nm}$ and $F_n$ respectively, are the components of the *global stiffness matrix* $\mathbf{K}$ and the *global load vector* $\mathbf{F}$.

A coefficient $K_{nm}$ represent the nodal force (or moment) resultant $R_n$ due to the nodal displacement (or rotation) $U_m = 1$ assuming all other $U_m$’s and all loads to vanish.
A coefficient $F_n$ represents the force (or moment) resultant $R_n$ due to the external loads acting directly on the node and on its surrounding elements assuming all $U_m$’s to vanish. In *EasyStatics* loads can be concentrated forces, concentrated moments and uniformly distributed loads acting on the bars in any direction.

In order to determine the numerical coefficients $K_{nm}$ and $F_n$, which relate the nodal resultants $R_n$ to the nodal displacements $U_n$ at the global system level, the same kind of numerical coefficients $k_{ij}$ and $f_i$ relating the nodal resultants $r_i$ to the nodal displacements $u_i$ of each single bar element are first determined locally.

The following linear relations hold for each element:

$$r_i = -\sum_m k_{ij} u_j + f_i = 0$$

or, in matrix notation

$$r = -ku + f$$

where $k$ is the *local element stiffness matrix* and $f$ the *local element load vector*.

A coefficient $k_{ij}$ represents the force (or moment) resultant $-r_i$ acting on the element node due to the nodal displacement (or rotation) $u_j = 1$ assuming all other nodal displacements to vanish.

A coefficient $f_i$ represents the force (or moment) resultant $r_i$ due to the external loads acting on the element assuming all the nodal displacements to vanish.

### 5.2.2 Global System: Assembling and Solution

Once the coefficients $k_{ij}$ and $f_i$ of the local stiffness matrix $k$ and the local element load vector $f$ have been determined for each element, these can be assembled by a simple summation procedure so as to obtain the global coefficients $K_{nm}$ and $F_n$ of the global stiffness matrix $K$ and the global load vector $F$, respectively.

If the nodal displacement parameter $U_n$ is free, i.e. there is no support in the direction of the displacement $U_n$ so that its value is not prescribed, the nodal equilibrium condition $R_n = 0$ must be satisfied. This leads to a global system of equilibrium equations to be solved for all free nodal displacements $U_n$:

$$R_n = -\sum_m K_{nm} U_m + F_n = 0 \quad \text{where } U_n \text{ is not prescribed.}$$

or, in matrix notation:

$$K U = F$$

A prescribed displacement $U_n$, e.g. at a fixed supported node, does not represent an unknown to be determined. The corresponding equilibrium equation does not have to be considered, but the $K_{nm}$ coefficients can be used for determining the not vanishing support reaction $R_n \neq 0$.  

66
It has to be noted here, that if the structure is not properly supported or too many hinges are prescribed, instabilities occur. That means that the global stiffness matrix $K$ is singular so that the above mentioned linear equation system cannot be solved as equilibrium cannot be fulfilled. In this case EasyStatics shows one possible rigid-body displacement mechanism of the unstable structure.

### 5.2.3 Element Results

Once all displacement parameters $U_n$ are known, the following secondary quantities along the element’s axis can be determined:

- the axial displacements in the longitudinal and transverse directions;
- the distribution of the normal force $N$;
- the distribution of the bending moment $M$;
- the distribution of the shear force $V$;
- the stresses $\sigma_T$ and $\sigma_B$ at the top and bottom fibers of the sections.

### 5.2.4 Exercises

In this section, some exercises are suggested which should help students with an adequate knowledge of theoretical background to develop a feeling for structural behaviour. With EasyStatics this can be done by experimenting, like in a virtual laboratory, with different situations.

Of course, the exercises discussed below are just a few examples. It is the actual teacher who has to design its demonstrations and exercises in a way that suits him - EasyStatics is just a tool to do this in a way which takes into account today’s state-of-the-art of structural analysis.

**Comparison Between Different Types of Frame**

As a first exercise students can be asked to compare the behaviour of different types of frame under horizontal and vertical loads. A statically determinate three-hinges frame (the top right of Figure 5.1), a simple statically indeterminate frame (top left) and a frame with built-in supports (bottom) are considered. The compare option (see Appendix Section A.15) is activated. The teacher first prepares an EasyStatics computational state by dividing the screen into three parts, one for each type of frame under vertical loads, as shown in figure 5.1. Students can then be asked to qualitatively sketch the displacements and the bending moments of the frames and to report whether there is a difference in their behaviour. These questions can be written by using the intern editor provided by EasyStatics (see Appendix Section A.16). Student are expected to notice that the frame
with three hinges has greater vertical displacements than in the two other cases.

The same can be asked with respect to horizontal loads in a second computational state (see Figure 5.2). In this case, students should notice that the fixed frame has maximal horizontal displacements which are much lower than in the other two cases.

Figure 5.1: Frames under vertical loads.

Figure 5.2: Frames under horizontal loads.
Multi-Stores Frame

In this exercise a multi-stores steel frame under wind load and self-weight is considered (see Figure 5.3). The vertical beams are HEB profiles with decreasing cross section from the bottom to the top, whereas all the horizontal beams have identical IPE profiles. Students could be asked to suggest different ways for reducing the maximal horizontal displacements.

These could be

- introduction of bracings;
- fixing the supports of the structure;
- increasing the stiffness of some bars.

The simplest way to compare different suggestions is to subdivide the screen into four parts, each showing a model which can be changed at will (see Figure 5.3). The activation of the compare function allows to immediately see the differences between the models.

Figure 5.3: Possible solution to the exercise multi-stores frame under horizontal loads
Truss Structure: Culmann Assumption

In this exercise a truss structure (see Section A.7 and Figure 5.4) is considered, e.g. with

- the bottom bars made of steel profiles HEB 300;
- the upper bars made of steel profiles HEB 200;
- the diagonal bars made of steel profiles HEB 100;
- a concentrated load of 20 kN acting in the middle of the bridge;

Students can be asked to

- identify which bars are in tension and which are in compression;
- qualitatively sketch the displacements;
- show how the normal forces change in the bottom and upper chord if the height of the truss is increased or decreased;
- discuss what happens to the normal forces if both supports are horizontally fixed;
- test the validity of the Culmann hinge-connection assumption\(^2\), by observing whether the displacements and the normal forces change considerably if the system is modelled as a frame instead of a truss and if the self-weight is considered by means of distributed loads instead of concentrated nodal loads only.

For the two first points, students can be additionally asked to find an analogy between the forces distribution and the displacements of the truss, and those of an horizontal simply supported beam.

The exercise can be solved as follows: in one computational state, two models of the structure are generated (see Figure 5.4). One model is higher than the other. Students should firstly observe that

- the bottom bars are in tension;
- the upper bars are in compression;
- the two diagonals near the concentrated force are in tension and so are all the diagonals with the same inclination as the two considered diagonals.

Then, by activating the compare function, they can observe that the higher is the truss, the smaller become the normal forces in the bars and the vertical displacements.

\(^2\)Over 100 years ago Culmann introduced his truss-assumption that bending can be neglected even when the bars in reality are not hinge-joined
Figure 5.4: Displacements and normal forces in two trusses with different height.

In another computational state students can enter two beams under the conditions mentioned above so as to obtain displacements and section forces.

In two other computational states, the stress fields of the two beams can be shown.

From these computational states, students can observe that

- the higher is the cross section, the smaller are the vertical displacements (see Figure 5.5);
- the tension and compression resultants are found near the top and bottom edge, respectively (see Figure 5.6);
- the higher is the cross section, the smaller are the values of the tension and compression stresses and their resultants (see Figure 5.7).

By comparing the two systems (truss and beam), students can observe that the behaviour of the simply supported beam is similar to that of the truss.

In another computational state (see Figure 5.8), students can compare two truss models:

- one model is the assigned truss with one support horizontally free;
- the other model is the same truss with both supports horizontally fixed.
Figure 5.5: Displacements of two horizontal beams with different height are compared.

Figure 5.6: Stress fields of the horizontal beam with a lower height.

In the second case an arch effect is observed: the upper cord is in compression, whereas the bottom chord is almost not loaded.
In a final computational state, students can compare displacements and normal forces distributions, first in the case of the structure considered as a truss, then as a frame. They should observe that Culmann’s assumption is, to a large extent, valid.

Figure 5.8: Displacements and normal forces in a truss model with one support horizontally free and a truss horizontally fixed at both supports.
Figure 5.9: Comparison between the structure considered as a truss and as a frame.

Arch Effect

In this exercise two structures are considered. One is a horizontal simply supported beam with a concentrated load in the middle, the other is the same beam but with the mid-point slightly higher than the two end supports. Students could be asked to consider:

- how normal forces and bending moments change from the first to the second system;
- how normal forces and bending moment change in the double inclined beam if one of the horizontal supports is left out.

By comparing the bending moments and the normal forces of the two structures, students should observe that in the double inclined beam there is an arch effect, which causes a great reduction of the bending moments and large normal forces (see Figure 5.10 and Figure 5.11).

If students look at the stress distribution inside the two beams, they can observe that in the horizontal beam there are both the compression and tension resultants (see Figure 5.12), whereas in the double inclined beam there is only a total compression resultant inside the beam (see Figure 5.13).
Chapter 5. *EasyStatics* as a Tool for Teaching

Figure 5.10: Bending moments of an horizontal and a double inclined beam fix at both ends.

Figure 5.11: Normal forces of a horizontal beam and a double inclined beam fix at both ends.
Figure 5.12: Total normal forces resultant of the horizontal beam.

Figure 5.13: Total normal forces resultant of the double inclined beam.
Concrete Beam: Truss Model

In this exercise a rectangular concrete beam simply supported on the left side and built-in on the right under self-weight is considered. This can help students to understand why and where steel reinforcements and stirrups are necessary.

In one computational state, students can generate two models of the beam. In one model the bending moments are shown, in the other the shear forces. Students should consider the following:

- where the bending moment is positive, the tension is in the lower part of the beam; where the bending moment is negative the tension is in the upper part. Longitudinal steel bars have to be introduced where tension forces develop. Their required sectional area is more or less determined by the bending moment and the normal force and, in general, only little influenced by the shear force;

- shear forces cause traction in vertical direction. Stirrups must therefore be introduced the cross sections of which are proportional to the shear forces. Stirrups are particularly needed in the vicinity of the supports where these forces are large.

Students can show the truss model generated by EasyStatics to check whether their assumptions are true. In addition, they can observe the bending moment and the shear distribution along the selected bar and correlate them with the forces distribution in the truss model (see Figure 5.14).

![Figure 5.14: Cantilever beam's truss model.](image-url)
Influence lines

Although influence lines are seldom used in practice today, they still represent a useful tool for didactic purposes.

Using EasyStatics, three copies of a structure can be created showing the influence lines for bending moments, shear and normal forces, respectively (see Figure 5.15)\(^3\). Consider for example the model on the top left. The maximum/minimum value of the influence lines gives the position where a concentrated moving load produces the maximum/minimum bending moment at the section marked with a circle on the element.

Figure 5.15: influence lines for bending moments, shear and normal forces at the marked section.

It can be shown (Landt’s theorem) that the influence line for a section force is given by the structure’s displacements in the direction of the load (which in EasyStatics are always assumed to be vertical), due to a concentrated internal unit deformation associated with this section force. Influence lines are determined simply by computing the structure’s displacements caused by such unusual load cases.

\(^3\)The program allows only the representation of these kind of influence lines.
5.3 2nd Order Theory - Stability

The so-called 2nd order theory considers the influence on bending behaviour of initial normal forces, i.e. of normal forces which are assumed to act on the structure in its undeformed reference configuration. With the FEM, these are determined with the help of the geometric stiffness matrix, first developed by Argrys around 1960 (see [63]).

5.3.1 Normal Force Influence on Bending Behaviour

If initial normal forces $N$ are present in the undeformed configuration, the displacements are not only proportional to the external forces, but also depend on these initial normal forces $N$. The greater are compressive forces $N$, the greater are the bending displacements due to $F$. In fact, if the normal compressive forces $N$ approach a critical limit, the displacements due to the external forces tend to infinity. This means that the structure, due to large enough initial compressive normal forces, loses all its load carrying capacity. This phenomenon is known as instability or buckling. If the forces $N$ were to change their sign and become tensional the deflections due to the external forces would become smaller.

When students deal with 2nd order effects, they should consider

- how displacements and section forces are to be determined taking into account the influence of initial normal forces;
- how large the initial forces can become before reaching instability.

This can be done with the help of the geometric stiffness matrix $K_G$.

5.3.2 Element Geometric Stiffness Matrices

While the elastic stiffness matrix $k$ is proportional to the elastic stiffness, i.e. to Young’s modulus, the geometric stiffness $k_G$ is proportional to the initial normal force $N$, which is assumed to be constant, i.e. independent of the nodal displacements.

At the element level the coefficients $k_{Gij}$ of $K_G$ are proportional to $N$ and must be added to the coefficients $k_{ij}$ of $k$ in order to take into account the 2nd order effects due to initial normal forces $N$:

\[ r_i = -\sum_j (k_{ij} + k_{Gij}) u_j + f_i \]

or in matrix notation:

\[ \mathbf{r} = -(\mathbf{k} + \mathbf{k}_G)\mathbf{u} + \mathbf{f} \]

It is important that students understand the physical meaning of the coefficients of the matrix $k_G$.
• $k_{Gi j} =$ coefficient to be added to $k_{ij}$ for taking into account the influence of the initial force $N$ on the element’s bending behaviour. $k_{Gi j}$ is proportional to $N$.

$k_{G}$ only concerns the nodal forces in the bending direction (this is valid also for truss elements). If $N$ is positive, i.e. if it corresponds to a tensile force, the coefficients $k_{Gi j}$ are such that the bending stiffness of the element, and therefore of the global system, is increased otherwise, in the case of compressive forces, the opposite occurs.

5.3.3 Structural Analysis by 2nd Order Theory

Structural analysis by the 2nd order theory simply means that, instead of $k$ alone, the sum $k + k_{G}$ of the elastic element stiffness matrix $k$ and the geometric stiffness matrix $k_{G}$ is used for assembling the global system of nodal equilibrium equations. This leads to the following global system of equations for the nodal force resultants $R_{n}$ associated with non-prescribed nodal displacements $U_{n}$:

$$R_{n} = -\sum_{m}(K_{nm} + K_{Gnm})U_{m} + F_{n}$$

or in matrix notation: $R = -(K + K_{G})U + F$

where $K_{G}$ is the *global geometric stiffness matrix* of the system. As already mentioned, it is important to note here that the initial normal forces $N$ in all elements are assumed to be independent on the nodal displacements $U_{n}$ obtained by 2nd order analysis.

In *EasyStatics* the initial normal forces $N$ in each element are first determined by 1st order analysis, then, in a second step, they are used for determining the elements $k_{G}$’s as needed for 2nd order analysis.

The determination of the nodal displacements $U_{n}$ is only possible if the system does not become unstable due to excessively large compressive forces.

5.3.4 Linear Stability Analysis

Because the element geometric stiffness matrix $K_{G}$ depends linearly on the initial normal forces $N$, it is possible to introduce the same factor $\lambda$ for the initial normal force of each element. This, in matrix notation, leads to the equations:

$$r = -(k + \lambda k_{G})u + f \quad \text{assembled to:} \quad R = -(K + \lambda K_{G})U + F$$

When, with $\lambda = \lambda_{CR}$, the structure is unstable, the matrix $(K + \lambda_{CR}K_{G})$ is singular. In this case displacements $U$ are possible without any load $(F=0)$.

For determining $\lambda_{CR}$ the following condition is used:

$$\sum_{m}(K_{nm} + \lambda_{CR}K_{Gnm})U_{m} = 0 \quad \text{or:} \quad (K + \lambda_{CR}K_{G})U = 0$$
This homogeneous system of linear equations for the unknown nodal displacements \( \mathbf{U} \) and for \( \lambda_{CR} \) has a solution different from \( \mathbf{U} = 0 \) only for some so-called eigenvalues of the critical load factor \( \lambda_{CR} \). These values cause the matrix \( (\mathbf{K} + \lambda_{CR} \mathbf{K}_G) \) to become singular.

Referring to the above mentioned condition, only the smallest value of \( \lambda_{CR} \) is here of interest. In fact, this represents the lowest possible initial normal force intensity leading to instability. Above this limit, stable equilibrium is no longer possible. Its associated eigenvector \( \mathbf{U} \) describes the buckling shape of the structure due to the initial normal forces \( \lambda_{CR} \mathbf{N} \) in all elements.

The information provided by \( \lambda_{CR} \) is useful for assessing the theoretical safety margin of a slender structure with respect to buckling.

### 5.3.5 Further Considerations

When *EasyStatics* analyzes a structure according to the 2\(^{nd}\) order theory, a linearized three-step procedure is carried out.

In the first step, the normal forces \( \mathbf{N} \) due to the prescribed external loads are determined by 1\(^{st}\) order analysis. This defines the distribution of the initial normal forces in all elements.

The stability eigenvalue problem needed for determining the critical buckling factor \( \lambda_{CR} \) is then solved as discussed in the last section.

After \( \lambda_{CR} \) has been found by solving the eigenvalue problem, a factor \( \beta \) can be specified (default \( \beta = 1 \)) which will be used for determining the initial normal forces \( \mathbf{N} = \beta \mathbf{N} \) needed for static (or also dynamic: see next section) 2\(^{nd}\) order analysis.

A final point must be stressed: The geometric element stiffness matrix \( \mathbf{k}_G \) represents an approximation, meaning that equilibrium is only approximately fulfilled. This means that by subdividing the element into sub-elements, different critical load factors \( \lambda_{CR} \) are obtained. Convergence, however, is excellent. Even for slender frame elements with high initial normal forces a subdivision into two or three sub-elements is, in most cases, sufficient for practical purposes. This can be easily checked by an interactive program like *EasyStatics*.

### 5.3.6 Exercises

The following exercises are suggested in order to help students

- understand the effects of the initial normal forces on a loaded structure by comparing the results of 1\(^{st}\) and 2\(^{nd}\) order analysis;
- understand buckling phenomena;
- be aware that the solution is an approximation and consequently that the structure might need to be subdivided into more sub-elements.
Horizontal Simply Supported Beam

In this exercise a simply supported horizontal beam with a vertical and a horizontal load is considered. Students should compare displacements and bending moments of the structure analyzed by 1st and 2nd order theory. The effects of compressive and tensile initial normal forces on structural behaviour are to be observed.

To carry out this exercise 4 copies of the assigned model can be created. The screen is subdivided as follows (see Figure 5.16):

- top left: the structure is analysed by 1st order theory;
- top right: the structure is analysed by 2nd order theory;
- bottom left: the structure analysed by 2nd order theory is loaded with a smaller horizontal compressive concentrated load at this end;
- bottom left: the structure analysed by 2nd order theory is loaded with a tensile horizontal concentrated load at this end.

![Diagram of beam analysis](image)

Figure 5.16: Moments and displacements of a simply supported beam analysed by 1st and 2nd order theory.

Using the compare function one can observe that

- initial compressive normal forces increase vertical displacements (Figure 5.16, top right);
• the smaller the value of compressive normal forces, the smaller vertical displacements (Figure 5.16, bottom left);

• initial tensile normal forces decrease vertical displacements (Figure 5.16, bottom right);

• if the initial normal forces lie above the buckling load, the beam becomes unstable, so that equilibrium with the external distributed load cannot be fulfilled.

Considering the bending moments distribution in the case of 2\textsuperscript{nd} order analysis, one can observe that equilibrium conditions are not satisfied. This is because the geometric stiffness matrix $K_G$ represents an approximation. However, convergency may be reached by subdividing the element into sub-elements (see Figure 5.17).

Figure 5.17: Convergence by subdividing the beam into more elements.

In a successive computational state, the display has been subdivided in order to show the following:

• top left: buckling shape and load of the unmodified beam;

• top right: buckling shape and load of the same beam but with a bigger span (1 m longer);

• bottom left: buckling shape and load of the same beam but with bigger cross section dimensions (each side 10 cm longer).
The student should observe that

- in each case the buckling shape is approximately half a sinus-wave in both positive and negative directions \((\pm \sin(\pi x/L))^4\);
- the buckling load decreases if the span of the beam increases. That means the structure becomes unstable with a lower normal force;
- the buckling load, i.e. the load carrying capacity, decreases if the span of the beam increases.

Figure 5.18: Buckling shape and load of a simply supported beam with modified parameters.

If one compares the critical buckling load of the beam modelled with a single element provided by EasyStatics, he will find that this is about 21% greater than Euler’s theoretical value:

\[
N_{CR} = \lambda F = \frac{EI \pi^2}{L^2}
\]

\(^4\)Only the distribution of these displacements is relevant, not their sign or their numerical values which can be multiplied by any factor.

84
This is due to the fact that the geometric stiffness $k_G$ of Figure 5.15 represents an approximation. However, if the element is subdivided into two sub-elements, the error is already smaller than 0.8%. Finer element subdivisions lead to very rapidly converging results.

**Arch Bridge**

In this exercise 2\textsuperscript{nd} order effects of the same arch bridge under vertical loads with the same value but opposite direction are considered. One bridge acts in compression and the other in tension. Students should consider and comment upon the displacements and normal forces distribution of both systems (see Figure 5.19).

![Figure 5.19: 2\textsuperscript{nd} order effects on the same arch bridge under vertical loads with opposite directions.](image)

One can observe that with compressive initial normal forces displacements and section forces increase whereas tensile initial normal forces have a stiffening effect (see Figure 5.19). The stability factor $\lambda_{CR}$ of the bridge on the left is negative, whereas that of the bridge on the right has the same absolute value, but is positive.

**Multi-Stores Steel Frame**

In this exercise a slender multi-stores steel frame under 2\textsuperscript{nd} order effects has to be analyzed. HEB 240 and IPE 300 steel profiles are used in the vertical and in the horizontal beams, respectively. The frame is loaded by a horizontal wind load and by concentrated vertical unit compressive loads of 1 kN, as shown in the top left corner of the figure below.
Students can be asked to

- show the displacements and bending moments resulting from a linear elastic analysis of the 1\textsuperscript{st} and 2\textsuperscript{nd} order and then compare them. In case of 2\textsuperscript{nd} order analysis, the initial normal forces can be defined by multiplying the normal forces from 1\textsuperscript{st} order analysis with a factor $\beta$ of 50\% of the buckling load ($\beta = \lambda_{CR}/2$);

- proceed in the same way as in the previous step considering tensional initial normal forces.

To carry out this exercise, one can generate three copies of the assigned model, as shown in Figure 5.20, showing that 2\textsuperscript{nd} order effects cause displacements and bending moments to increase considerably when the forces in the columns are compressive (top right corner of Figure 5.20), whereas they decrease when they are tensional (bottom right corner of figure 5.20).

Figure 5.20: 1\textsuperscript{st} and 2\textsuperscript{nd} order analysis of a slender steel frame.
5.4 Dynamic Eigenmodes

The loads acting on a structure can vary over the time thus causing corresponding displacements with non-vanishing velocities and accelerations. When masses are accelerated, according to Newton’s law, inertia forces are generated. If the accelerations are large enough, these dynamic inertia forces have to be considered when formulating nodal equilibrium conditions. The characteristic of dynamically loaded structures is that if the load frequency \( \Omega \) is near one of the structure’s eigenfrequencies \( (\Omega_1, \Omega_2, \Omega_3, \Omega_4, \text{ etc.}) \) the amplitude of the displacements becomes large. In the ideal undamped case it becomes infinitely large. Consequently, when dynamic loads are considered, the following question is relevant to structural design:

- knowing the expected load frequency \( \Omega \), how is the structure to be designed so that \( \Omega \) does not coincide or nearly coincide with one of the structure’s eigenfrequencies \( \Omega_i \)?

For answering this question the dynamic eigenfrequencies of the structure are to be determined. For this purpose the mass matrices, which represent another fundamental concept of modern structural mechanics, come into play.

5.4.1 Inertia Forces and Mass Matrices

Considering the contributions of the element stiffness matrix \( k \) and geometric stiffness matrix \( k_G \) as previously explained, the time-dependent nodal forces \( r_i(t) \) of an element can be determined as follows:

\[
r_i(t) = -\sum_j (k_{ij} + k_{Gij}) u_j(t) + f_i(t) - \sum_j m_{ij} \ddot{u}_j(t)
\]

or, in matrix notation:

\[
r(t) = -m \ddot{u}(t) - (k + k_G) u(t) + f(t)
\]

- A coefficient \( m_{ij} \) of the local element mass matrix \( m \) represents the nodal force \(-r_i\) associated with \( u_i \) due to inertia forces caused by the masses attached to the element for a unit nodal acceleration \( \ddot{u}_j = 1 \).

Again, and by the same procedures explained in Section 5.2.2, the global equilibrium equations for the nodal force resultants \( R_n(t) \) of the vector \( R(t) \) can be assembled from the local element coefficients \( k_{ij} \), \( k_{Gij} \) and \( m_{ij} \) of the matrices \( k \), \( k_G \) and \( m \) of each element:

\[
R_n(t) = -\sum_m M_{nm} \ddot{U}_m(t) - \sum_m (K_{nm} + K_{Gnm}) U_m(t) + F_n(t)
\]

or, in matrix notation:

\[
R(t) = -M \ddot{U}(t) - (K + K_G) U(t) + F(t)
\]
where $M$ is the structure’s global mass matrix whose $M_{nm}$ coefficients are assembled in the same way as the $K_{nm}$ coefficients of the global stiffness matrix $K$ (see Section 5.2). The coefficient $M_{nm}$ of the global mass matrix $M$ is defined as follows:

- $M_{nm} = \text{n-th nodal force - } R_n$ associated with $U_n$ due to the inertia forces caused by the masses attached to the structure for a unit nodal acceleration $\ddot{U}_m = 1$.

### 5.4.2 Eigenfrequencies and Eigenmodes

As in the static cases, the nodal resultants $R_n(t)$ associated with non-prescribed nodal displacements $U_n(t)$ must vanish. That leads to the following equilibrium equations:

$$M \dddot{U}(t) + (K + K_G) \dot{U}(t) = F(t)$$

This is a system of coupled differential equations of second order in time which, however, seldom needs to be explicitly solved as only the eigenfrequencies $\Omega_i$’s and the corresponding eigenmodes are of interest for most practical design purposes.

To determine these the following assumption for the displacements has been made (see [63] for more details):

$$U_n(t) = E_n \cos(2\pi \Omega_i t) \quad \text{or} \quad \dot{U}(t) = E_i \cos(2\pi \Omega_i t)$$

With $F(t) = 0$, this assumption leads to the following matrix eigenvalue problem

$$\left( K + K_G - (2\pi \Omega_i)^2 M \right) E_i = 0$$

which has to be solved for the unknown eigenfrequencies $\Omega_i$ and eigenvectors $E_i$. These describe the shapes of the structure’s eigenmodes. Of course, the geometric stiffness matrix $K_G$ is needed only if 2$^{nd}$ order effects are to be considered. As the eigenfrequencies $\Omega_i$ and eigenvectors $E_i$ are independent of the external loads, the graphic objects representing the loads are not shown by EasyStatics when dynamic analysis is performed.

In contrast to the eigenvalue problem of stability analysis discussed in Section 5.3 where only a single eigenmode associated with the smallest eigenvalue is of interest, here a small number of eigenfrequencies $\Omega_i$ near the expected load frequency $\Omega$ are to be determined. Generally, only a few eigenmodes with the smallest eigenvalues are of practical interest. EasyStatics can only determine 4 of them.

### 5.4.3 Exercises

The exercises proposed in this sub-section should help students understand how eigenfrequencies and eigenmodes of a structure are influenced by changes in stiffness, mass, dimensions, supports conditions and elements subdivision. It should also be understood how eigenfrequencies and eigenmodes change if 2$^{nd}$ order effects are taken into account.
Multi-Stores Concrete Frame

In this exercise the first four eigenfrequencies and eigenmodes of a multi-stores concrete frame are first to be found. Then students should consider how these change

- if the elastic (or Young’s) modulus E module changes from 20 MPa to 80 MPa;
- if the field length and the columns height increase;
- if the columns are simply supported or built-in to the ground;
- if the elements subdivision is refined;
- if 2nd order effects due to compression and tension forces are considered.

Again, students can generate four copies of the given structure, then select the dynamic analysis button and show for each model the chosen eigenmode with relative eigenfrequencies (see Figure 5.21).

Figure 5.21: First four eigenmodes and eigenfrequencies of a multi-stores concrete frame.

They should observe that

- the greater is the requested number of eigenmode, the more complex is its shape and the greater its corresponding eigenfrequency;
- with increased elasticity modulus by a factor 4 from 20 to 80 the eigenfrequencies of the structure increase by a factor $2 = \sqrt{4}$ (see Figure 5.22);
- with increased field length or column height, the eigenfrequencies decrease (see Figures 5.23 and 5.24);
• with columns fixed on the ground, the system is stiffer. As a consequence, the eigenfrequencies are greater (see Figure 5.25);

• especially in the case of the larger 3rd and 4th eigenfrequencies (Figure 5.26), a finer subdivision of the system decreases them\(^5\). Students should notice that convergence is fast. For this exercise, a single subdivision of each element in two parts suffices to obtain a good approximation.

Figure 5.22: First four eigenmodes and eigenfrequencies of a multistory, multibay concrete frame with increased E module.

To notice the influence of 2nd order effects on dynamic behaviour, students can generate two copies of the assigned frame (see Figure 5.27). The two models represent the 3rd eigenfrequency and eigenmode. The model on the left does not consider initial normal forces, whereas the one on the right does. In this case, compressive normal forces are caused by the concentrated forces shown in Figure 5.27.

Students should observe that, in the case of initial compressive normal forces, the eigenfrequency decreases. The opposite is noticed considering initial tension normal forces obtained by changing the orientation of the vertical forces of the previous case (see Figure 5.28).

\(^5\)In fact it can be shown that mesh refinement leads to more flexible FE-models, thus to lower eigenfrequencies
Figure 5.23: First four eigenmodes and eigenfrequencies of a multistory, multibay concrete frame with increased field length.

Figure 5.24: First four eigenmodes and eigenfrequencies of a multistory, multibay concrete frame with increased column height.
Figure 5.25: First four eigenmodes and eigenfrequencies of a multistory, multibay concrete frame with fixed supports.

Figure 5.26: First four eigenmodes and eigenfrequencies of a multistory concrete frame modelled with subdivided elements.
Figure 5.27: 3\textsuperscript{rd} eigenmode and eigenfrequency of a multistory concrete frame considering and not considering initial normal compression forces.

Figure 5.28: 3\textsuperscript{rd} eigenmode and eigenfrequency of a multistory concrete frame considering and not considering initial normal tension forces.
5.5 Rigid-plastic Analysis

If one considers the external loads $F_n$ being multiplied by an unknown factor $\lambda$ and wants to determine the critical factor $\lambda_{CR}$ for which the structure collapses, then the rigid-plastic theory comes into play. Based on the assumptions that the structure is “ductile enough” and that 2nd order and dynamic effects can be neglected, this theory leads to the following Limit-state theorems:

- **Static or lower-bound theorem:** If the section forces everywhere satisfy equilibrium with the external loads $\lambda F_n$ and nowhere violate plasticity conditions, then these loads lie below the critical load for which the structure would collapse.

- **Kinematic or upper-bound theorem:** Consider the structure while it collapses at a constant speed. If the rate of work of the external loads $\lambda F_n$ moving with the structure and the total rate of internal plastic work dissipated into heat are equal, then $\lambda$ is an upper-bound of the true critical load factor $\lambda_{CR}$.

Both theorems can be used to numerically determine the critical load factor $\lambda_{CR}$. The static theorem also allows one to determine the section force distribution expected shortly before collapse. The kinematic theorem provides information on the way the structure would collapse, i.e. the form of the collapse mechanism.

The static theorem leads to one of the most fundamental concepts of structural design:

- if a ductile structure is dimensioned so as to resist a section force distribution which satisfies equilibrium with the external loads but is otherwise arbitrary, then safety against collapse is guaranteed.

The designer is therefore free to determine the resistances of each member of a structure in the way he finds appropriate as long as he uses a model, possibly much simplified, in which equilibrium is satisfied. This concept is explicitly or - more often - implicitly applied to all kinds of practical design problems, specially for reinforced concrete structures.

5.5.1 Plane Truss and Frame Models

In order to determine the ultimate load factor $\lambda_{CR}$, the shape of the collapse mechanism and the section force distribution during collapse, the structure is, again, subdivided into elements which are supported or connected to other elements at their nodes. In the nodes the same displacement parameters $U_n$ (two displacements and, for frames only, one rotation) are introduced with the same associated nodal force resultants $R_n$ (two forces and, for frames only, one moment) as for elastic analysis.

At the element level, three section force parameters are introduced: the normal force $N$ in the middle of the element and the bending moments $M_{START}$ and $M_{END}$ at the start and end points, respectively (see [63], Section 6.5).
Chapter 5. *EasyStatics* as a Tool for Teaching

The way the structure is expected to fail when the load factor \( \lambda \) reaches the critical value \( \lambda_{CR} \), i.e. the shape of the collapse mechanism, is described by the nodal velocity parameters \( \dot{U}_n \) which are kinematically compatible with the plastic rotations and the elongations in a number of “plastic hinges”. These are located at both ends of each element, where plastic deformations are assumed to take place.

### 5.5.2 Limit Analysis by the Static Theorem

In order to satisfy nodal equilibrium, the nodal resultants \( R_n \) have to be determined. These are now linear functions of the unknown \( M-N \) element parameters and of the unknown load factor \( \lambda \) multiplying the external loads. The \( R_n \)'s associated with all free displacement parameters \( U_n \) are then set to zero, which leads to a linear system of as many equations as in the elastic case but with different unknowns (see Section 5.2).

In addition to equilibrium, the static theorem requires the plasticity conditions for the section forces to be everywhere fulfilled. This means that the section forces \( N \) and \( M \) must always represent a point \((M, N)\) within the M-N interaction diagram, where each side of the polygon (see Appendix Section A.8) represents a linear inequality which \( M \) and \( N \) have to satisfy. In *EasyStatics* the plasticity conditions are not checked everywhere but only at the element’s ends, which are the only sections where plastic flow can occur. Here, the point \((M, N)\) is within or at the boundary of the M-N interaction diagram. Because \((M, N)\) cannot increase above these limits, this polygonal boundary defines the limit resistance of the rigid plastic section for any combination of \( M \) and \( N \).

The \( M-N \) interaction diagram used in *EasyStatics* (see Appendix Section A.8) is a polygon with 8 vertices approximating the true so-called *flow figure* which has a curved boundary. Therefore, formulating plasticity conditions at the element’s ends means that for each element 16 (2 * 8) linear inequality constraints are formulated. These depend on the section forces and on the section resistances.

As an additional simplification, plasticity conditions for the shear forces \( V \) as well as the interaction of \( V \) with \( M \) and \( N \) are not considered in *EasyStatics*.

Based on the static theorem, which states that \( \lambda \) represents a lower bound of the theoretically correct ultimate load factor \( \lambda_{CR} \) when equilibrium and plasticity conditions are fulfilled, the maximum allowable \( \lambda \) factor is sought.

This maximum problem with linear equations and inequality constraints is called a Linear Program. It can be solved for the unknown section force parameters and for the critical load factor \( \lambda_{CR} \) by the well-known Simplex Algorithm (see Appendix A of [63]).

Knowing the section forces at the element’s ends, which satisfy equilibrium, the section force distributions inside the element can be exactly determined by formulating equilibrium conditions. If only concentrated nodal loads act on the structure, the normal force distribution is constant between the element’s ends while the bending moment distribution is linear. The plasticity conditions, which are enforced at both ends of each element, will therefore be satisfied exactly everywhere in each element since the maximum and minimum values of any \( M-N \) combination can be reached only at the start and end
points. From the static theorem it follows that \( \lambda_{CR} = \text{Maximum}(\lambda) \) then represents a lower-bound of the true ultimate load factor.

However, if external loads act on the element, the distributions of the normal forces \( N \) and the bending moments \( M \) which satisfy equilibrium exactly are no longer constant or linear, respectively. It is therefore possible that the plasticity conditions are violated somewhere within the elements. One of the conditions of the static theorem therefore may not be satisfied exactly so that the computed value for \( \lambda_{CR} \) no longer represents a lower-bound of the theoretical ultimate load factor. This can be clearly shown by EasyStatics when the check line (see Appendix Section A.9) is displayed. Obviously the colour of the check line is always blue at the element’s ends because the critical load factor \( \lambda_{CR} \) has been determined so as to satisfy plasticity conditions there. Within some elements, however, its colour may be red, meaning that the plasticity conditions are not fulfilled. To satisfy these one can simply subdivide the elements where the colour is red. In fact, the finer the subdivision, the larger the number of sections where plasticity conditions are checked and enforced leading to somewhat smaller and more precise values of the ultimate load factor \( \lambda_{CR} \).

### 5.5.3 Limit Analysis by the Kinematic Theorem

As soon as the external loads, which are multiplied by the factor \( \lambda \), increase above the critical limit \( \lambda = \lambda_{CR} \), the structure collapses. Unlimited (but small!) displacements and plastic deformations (plastic elongation and rotations) at a constant speed are then assumed to take place in a limited number of plastic hinges at any end of some elements. Obviously, because of this limitation, not all possible collapse mechanisms are considered.

Linear Programming is again used to find the structure’s nodal velocities, i.e. the form of the collapse mechanism, and the critical load factor \( \lambda_{CR} \).

According to the kinematic or upper-bound theorem, kinematic compatibility and equality between the rate of internal plastic work dissipation taking place at the plastic hinges and the rate of external work\(^6\) are enforced by means of linear kinematic compatibility equations. The load factor \( \lambda \) has to be minimized: \( \lambda_{CR} = \text{Minimum}(\lambda) \).

From the duality theorem of linear programming theory it follows that, because the two linear programs related to the static and kinematic theorem can be proved to be “dual” to each other (see appendix A of [63]), the load factors \( \lambda_{CR} \) obtained from their solutions are identical. This means that the static and the kinematic theorems of plasticity theory lead to the same critical load factor \( \lambda_{CR} \). This value represents at the same time both a lower- and an upper-bound, thus the theoretical ultimate load factor, only if the conditions of both theorems are everywhere satisfied, i.e. only if the section forces nowhere violate the plasticity conditions. The conditions of the kinematic theorem are always fulfilled exactly.

---

\(^6\) **internal plastic dissipation work**: work of the internal forces to bend and stretch a plastic hinge dissipated into heat; **external work**: work of the external loads during collapse; **kinematic compatibility**: compatibility of the nodal displacements with the deformations occurring at the plastic hinges.
Due to the duality of the linear programs, when solving one problem, the solution of the other is automatically obtained. This means that the collapse factor $\lambda_{CR}$, the element section forces $N_{\text{MIDDLE}}$, $M_{\text{START}}$ and $M_{\text{END}}$ at collapse as well as the nodal velocities describing the form of the collapse mechanism are all determined at the same time.

### 5.5.4 Advantages and Drawbacks of Plasticity Theory

Simple rigid-plastic analysis allows one to obtain an estimate of both the ultimate load factor of the collapse mechanism. This is not possible using elasticity theory which can only provide information on the level plastic where flow would first occur without considering stress redistribution. However, the following drawbacks of the simple rigid-plastic theory (as opposite to the non-simple step-by-step incremental elasto-plastic analysis) should be mentioned:

- No information on displacements and section forces under normal load conditions is obtained.
- Plasticity theory is based on the assumption that structures are “ductile enough” so as to allow section force redistribution without brittle failure. However, simple rigid-plastic theory, like elasticity theory, provides no information on ductility requirements.
- While the collapse mechanism is well described, the section forces only correspond to reality in the zones of the structure where plastic flow occurs. In the zones which remain rigid, the section forces obtained fulfill equilibrium and plasticity conditions but are otherwise arbitrary. In general, and unless the structure is statically-determinate, they provide no real information on the section forces at collapse.

The ideal way to proceed for design purposes is to compare elastic and rigid-plastic solutions, for which *EasyStatics* is ideally suited.

### 5.5.5 Exercises

The exercises of this section should help to understand how changes of section properties, structure geometry and load position influence the ultimate load factor and the collapse mechanism of a structure. Students are also expected to recognize whether the computed ultimate load factor is a lower-bound, an upper-bound or the exact theoretical value. Comparison between elastic and rigid-plastic analysis are also instructive.

**Continuous Beam Under Concentrated Load**

In this exercise, a continuous beam under a concentrated load is considered. Students could be asked to find the collapse mechanism of the structure and then to compare the bending moment distributions by rigid-plastic analysis and by elastic 1st order theory.
Chapter 5. *EasyStatics* as a Tool for Teaching

Figure 5.29 shows four copies of the given structure. On the top left the results of rigid-plastic analysis without any element subdivision are shown. The ultimate load factor is infinite because the plasticity conditions for M and N of the element where the concentrated load acts are only checked in its end sections, so that the beam can carry any load like if it would be simply supported with zero moments and normal forces at its ends.

To obtain a collapse factor and see the associated collapse mechanism a node under the concentrated load must be introduced. Rigid plastic analysis then provides a solution fulfilling both equilibrium and plasticity conditions. These are checked using the check line, which is everywhere blue (see Appendix Section A.9). Because the lower-bound and the upper-bound conditions are everywhere satisfied, the computed ultimate load factor $\lambda_{CR}$ represents the theoretically true one.

If the load is moved (bottom left), with the check line one observes that the plasticity conditions are no longer fulfilled exactly. It follows that the computed ultimate load factor no longer represents a lower-bound but only an upper-bound, thus it is larger then the true one.

Considering the copy at the top right, one observes that the bending moment distribution provided by rigid-plastic analysis, although not violating the conditions mentioned above, is somewhat unexpected and does not correspond to the elastic distribution, which is shown by the $1^{st}$ order analysis (see bottom right). Only in the element where the mechanism occurs is the moment distribution of the rigid-plastic case similar to the elastic one. This is because the introduction of the node under the concentrated load forces the bending moment to be linear, as in the elastic case.
While performing 1\textsuperscript{st} order analysis, the concentrated load can be multiplied by the ultimate load factor previously determined by rigid plastic analysis. From the check line, one can observe that the plasticity conditions of the structure analyzed by 1\textsuperscript{st} order elastic theory are not satisfied everywhere. This shows the effect of section force redistribution obtained by rigid-plastic analysis.

**Concrete Bridge under Self-Weight and a Concentrated Load**

In this exercise, a concrete bridge under self-weight and concentrated load is considered (see Figure 5.30). The pier cross sections are rectangular, 80 cm wide and 220 cm high, whereas the cross section of the deck is a double T, 310 cm wide and 150 cm high.

![Figure 5.30: Collapse mechanism and ultimate load factor of a concrete bridge.](image)

One can firstly observe how the collapse mechanism and the collapse factor change if the load is moved on the deck (see Figure 5.31).

Considering the situations shown in Figures 5.30 and 5.31 and selecting the check button, one observes that the ultimate load factor corresponds to an upper-bound of the theoretical value. In fact, the plasticity conditions are violated in some places, which is indicated by the red colour of the check line, because the plasticity conditions are checked only at the element start and end sections. Consequently, only if the bending moments and normal force distributions are constant or linear along the element, the plasticity conditions are not violated. This is the case when concentrated loads act on the nodes. In this case, the ultimate load factor represents a lower-bound of the true one. Otherwise, if external loads or distributed loads act on the element, the distributions of the normal forces $N$ and bending moments $M$, which satisfy equilibrium exactly, are no
longer constant or linear, respectively. It is therefore possible that the plasticity conditions might be violated somewhere within the elements. In this case, the ultimate load factor represents only an upper-bound of the theoretical value. Following these considerations, one should subdivide the structure into more finite elements until the theoretical value is well approximated. A possible element subdivision is shown in Figure 5.32.

*EasyStatics* can also show in two neighbouring panels (see Figure 5.33) the moment distributions obtained by rigid-plastic and elastic analysis. To facilitate the comparison between the two solutions, *EasyStatics* allows one to apply the collapse factor obtained from rigid-plastic analysis to the loads used for elastic analysis. The section force distributions, which will only be equal for statically-determinate structures, can be then readily compared for assessing the influence of the section force redistribution due to plastic flow.

**Steel Frame under Horizontal and Vertical concentrated Loads**

In this exercise a steel frame is considered with HEB300 profiles in the columns and IPE300 in the horizontal beams. The structure is subjected to horizontal and vertical concentrated loads as shown in Figure 5.34. Again, students could be asked to find the collapse mechanism and the collapse factor of the given structure and to state if they correspond to an upper bound or to the true ultimate load factor.

With the check button, they can observe that the plasticity conditions are violated in the middle of the horizontal beams (see Figure 5.35). Therefore, following the considerations of the previous exercise, they should subdivide the horizontal beams by introducing nodes.
Figure 5.32: Collapse mechanism and ultimate load factor of the concrete bridge with increased concrete resistance in the element deck.

Figure 5.33: Comparison between rigid-plastic and linear-elastic analysis.

in the vicinity of the vertical loads. With a refined subdivision, one will notice that the collapse mechanism and the collapse factor change (see Figure 5.36).
It should also be considered how the collapse factor and the collapse mechanism change with different support conditions, for example without rotational supports. One observes that the collapse mechanism is completely different (see Figure 5.37). As expected, the collapse factor decreases considerably when the supports can rotate because the structure becomes weaker.

Students can also be asked to consider what happens if some cross sections change.

They should observe that the collapse factor and the collapse mechanism again change. The first decreases because the plastic resistance of the columns is decreased (see Figure 5.34 and Figure 5.35).
Figure 5.36: Collapse mechanism and ultimate load factor of a steel frame with a refined subdivision.

Figure 5.37: Collapse mechanism and ultimate load factor of a steel frame without rotational supports.

Finally, one could be asked to consider what happens if the vertical loads decrease. They should observe that the collapse factor increases with the structure collapsing laterally (see Figure 5.39).
Figure 5.38: Collapse mechanism and ultimate load factor of a steel frame with rotationally free supports.

Figure 5.39: Same as in the previous figure considering smaller vertical loads.
5.6 Exercises on Section Dimensioning

The following exercises show how the material panel explained in Appendix Section A.8 can be used. The goal here is to help students:

- dimension the structure’s members in order to resist the load;
- understand how the required section dimensions of structural members change if the material changes;
- notice how the required section dimensions change if buckling and lateral buckling phenomena are taken into account;
- notice the influence of 2nd order effects on sections dimensioning and
- observe and comment upon the M-N interaction diagram in the case of rigid-plastic analysis.

Continuous Beam: Buckling and Lateral Buckling Phenomena

In this exercise a 3 m long continuous wooden beam of rectangular section is considered. This is subjected to

- a dead load (see Figure 5.40) consisting of the self-weight and a distributed load of 6kN/m (1st load case) and
- a live load (see Figure 5.41) consisting of a distributed load of 12kN/m (2nd load case).

The dead load is multiplied by the factor $F1 = 1.3$, whereas the live load is multiplied by the factor $F2 = 1.5$.

Students could be asked to

- dimension the structure’s members in order to resist the load;
- dimension the structure’s members if lateral buckling is considered and the resisting moment is reduced by the factor $\kappa_M = 0.8$;
- ascertain how the required section’s dimension changes if steel profiles or concrete are used.

Initially a rectangular wooden section, e.g. 12 cm high and 10 cm wide, could be considered. Then, with the check line it should be checked whether the section is well-dimensioned everywhere. As seen in Figure 5.42, the section in the middle of the fields and near the supports is clearly underdimensioned.
If a section window is opened by clicking where the bending moments are greatest (i.e. in the vicinity of the internal supports), one can see the M-N interaction diagram related to the section and either a red cross or a red arrow outside the green polygon, indicating that the resistance is insufficient. The section dimensions are then changed until the cross moves to the inside the M-N interaction diagram and becomes blue (see Figure 5.43).

At the same time, with section dimensions 26 cm high and 12 cm wide one can see that the check line becomes blue all along the beam.

If $\kappa_M = 0.8$ the section dimensions are insufficient to resist to the external loads so that they must be changed: the width remains the same, whereas the height increases by 2 cm. Students should be aware that lateral buckling phenomena are generally to be considered during the dimensioning of slender structures.

It is then useful to dimension in the same way the continuous beam, considering a rectangular reinforced concrete section. Modifying the different section parameters, a sufficient resistance can be obtained with a section 30 cm wide and 20 cm high and 4 $\phi$ 12 in the upper part and 4 steel bars $\phi$ 10 in the lower part (see Figure 5.44). If a double T concrete section instead of a rectangular one is used, one will obtain a sufficient resistance with a section 20 cm high and 20 cm wide and the same reinforcement as in the rectangular section (see Figure 5.45). This means that the second option allows some material saving.

Finally, if IPE steel sections are used, one finds that sufficient resistance is obtained with IPE 140 (see Figure 5.46).
Frame under 1<sup>st</sup> Order, 2<sup>nd</sup> Order and Rigid-Plastic Analysis

In this exercise a wooden frame is considered (see Figure 5.47), which is subjected to

- a dead load consisting of the self-weight and a distributed load of 20kN/m (1<sup>st</sup> load case) and
- wind load consisting of a distributed horizontal load of 10kN/m (2<sup>nd</sup> load case).

The dead load is multiplied, as in the previous exercise, by the factor $F1 = 1.3$, whereas the live load is multiplied by the factor $F2 = 1.5$. 

107
Figure 5.44: M-N interaction diagram for the rectangular concrete beam section under a load combination.

Figure 5.45: M-N interaction diagram for the double T concrete beam section under a load combination.

Students could be asked to

- dimension the structure’s members in order to resist the load;

- dimension the structure’s members considering initial normal forces, the values of which are those of the normal forces resulting from 1st order analysis multiplied by 20% of the buckling load;

- carry out a rigid-plastic analysis and observe whether the (Md-Nd) point is inside the M-N interaction diagram.
In a first step, the same square wooden section could be assigned to all the structural members, e.g. with a height and width of 20 cm. To check whether this section is sufficiently resistant one selects the check button which shows the members of the structure which are well-dimensioned and not. The check line is red in many sections (see Figure 5.47).

It follows, that the section parameters need to be modified. A possible solution will be one where

- the sections of the lower columns are changed to 40 cm height and 36 cm width;
- the sections of the upper columns are changed to 26 cm height and 26 cm width;
- the sections of the lower horizontal beams are changed to 38 cm height and 30 cm width and
- the sections of the lower horizontal beams are changed to 28 cm height and 20 cm width.

When 2nd order analysis is carried out and the initial normal forces are multiplied by 20% of the $\lambda_{CR}$ ($\beta = 0.2$), some section dimensions need to be changed because the section forces have become larger and the section resistances are no longer sufficient everywhere: the height of the lower column sections and of the left horizontal beam in the 1st floor have to be increased by 4 cm in height.

Finally, if a rigid-plastic analysis is performed, one obtains the mechanism shown in Figure 5.48 and observes that at the point where plastic hinges occur, the Md-Nd section forces are on the border of the M-N interaction diagram. This means that the section will reach its maximum resistance there.
Figure 5.47: Wooden frame under dead load and wind load. The check line is shown.

Figure 5.48: Collapse mechanism of a wooden frame and M-N interaction diagram
Chapter 6

Academic Experience with EasyStatics

Both the EasyStatics program and its e-learning platform have been used and tested in the following courses:

- Computer Statics (6th Semester, Civil Engineering, ETH Zurich)
- One week introduction on analysis and design (Civil Engineering, ETH Zurich)
- Statics course (Civil Engineering at the Winterthur Technical School and Civil Engineering Dept. at the Technical University, Graz)
- Plane structure design course (Architecture, ETH Zurich)

The experience made at ETH Zurich with the EasyStatics program and its e-learning platform is reported below.

6.1 Computer Statics Course

This course is a 2 hour weekly semester course attended by some 40 students. In addition to the 2 hours lesson, they have to carry out the home works possibly with the help of teaching assistants.

In his class, the teacher explains some theory covering the following topics:

- Linear-elastic 1st order theory, including influence lines
- Linear-elastic 2nd order theory
- Elastic stability
- Elastic eigenmode dynamics
• Rigid-plastic theory based on the limit load theorems

In order to better understand the theory, the teacher uses EasyStatics for demonstrations, which often contain several computational states files. In each computational state the teacher modifies the model and perform calculations in real time, just like working in a “virtual structural laboratory”. He can also show the influence of a parameter change on structural behaviour, helping students to grasp the concepts of the different analysis types, to make comparisons, to write comments and so on. The comments are included in each computational state so as to allow students to later review the teacher’s explanations.

After the class lesson, documents showing these demonstrations are available through the EasyStatics e-learning platform. This helps students to solve the homework exercises assigned every week. Here students have to make comparisons by using more copies of the structure, link concepts, suggest further possible solutions stimulating their creativity, write considerations with the internal editors or create text files using text programs, such the HTML editor included in EasyStatics, where EasyStatics graphics can be imported.

Once the exercises have been solved they are made available to the teaching assistants through the Internet.

Finally, during the oral examinations, students are asked about the theoretical concepts, taught with the aid of EasyStatics, and the experiences they made.

### 6.2 Introductory Course

Every year, the Civil Engineering Department of ETH Zurich organizes a one week introductory course of structural analysis and design for prospective new students. The course has to show within a few days what civil engineering is all about. For this purpose a project is chosen where students have to practice empirical, theoretical, written and oral skills. All this should be coupled with some entertainment so as to motivate them.

#### Project Definition

Students had to build a one meter long and 10 cm wide bridge made of glass or concrete. EasyStatics had to be used for the design concept. In a final step, the bridge is loaded up to failure with a testing machine that measures the ultimate load. This and the failure mechanism were then to be compared with the program results. The winning group is the one that is able to build the bridge with the highest load carrying capacity. Finally, each group has to explain the designed structure to the other groups.

At the beginning of the course the teacher gives some theoretical background knowledge. Concepts like equilibrium, bending moment, axial forces, loads, supports and the main characteristics of glass and concrete (good resistance of glass and concrete in compression; good resistance of concrete in tension if reinforcing steel bars are present) are briefly explained.
Students can then start to work. They choose the material and use *EasyStatics* to define the bridge geometry. They then make the structure stable by introducing supports, while the program informs them if instability occurs. As a further step, students apply the loads and observe the normal forces and bending moment distribution using *EasyStatics*. Then, depending on whether students work with glass or concrete, they try to optimize the structure’s geometry considering the different characteristics of the material, while displacements and stress distributions are shown in real time after any change.

An example of structure geometry chosen by one of the groups is shown in Figure 6.1. Here the results are reported that were obtained in winter 2004.

![Concrete bridge: Geometrical model of the glass bridge with normal forces.](image)

Figure 6.1: **Concrete bridge**: Geometrical model of the glass bridge with normal forces.

Once the geometry is defined, students work with the material windows (see Appendix Section A.8) and modify the section properties. Then, thanks to the check line (see Appendix Section A.9) they can see which part of the structure is either under- or over-dimensioned. Students can also check the stress distribution of some part of the structure. In fact the stress panel (see Appendix Section A.5) shows the highest and lowest stress of the selected bars to see if the stresses exceed the material strength.

Once the model is conceptually designed, students build it in the laboratory by cutting and gluing bars previously prepared or by building a carton shuttering, putting steel wires where necessary and pouring mortal (see Figure 6.2).

Students then use *EasyStatics* and start the rigid-plastic analysis which shows the ultimate load and the failure mechanism (see for example Figure 6.3).

Finally, students load their models using the testing machine until failure occurs so that they can compare the real ultimate load and the collapse mechanism with those obtained with *EasyStatics* (see Figure 6.4).

At the end they have to prepare a poster, possibly using *EasyStatics*’s HTML editor, so as to show their model, justify their design decisions and compare the expected and the obtained results. The posters are then presented orally during the closing session.
Figure 6.2: Concrete bridge: scaffolding.

Figure 6.3: Concrete bridge: failure mechanism.

Figure 6.4: Concrete bridge: real failure mechanism under the testing machine.
6.3 *EasyStatics* Assessment

The today’s state of the program is mainly a result of the interaction between program’s developers and program’s users, i.e. teachers and students using *EasyStatics* within a structural analysis course. These, using the program since its first prototype, have continuously tested it and provided a valuable feedback by communicating whether they had problems in its use, by saying which functionality was unclear and by eventually suggesting improvements of the user interface or new features. Additionally, at the end of a course, students and teachers were asked to complete the following questionnaire:

- **TECHNICAL ASPECTS**
  - Is the general design of the screen appropriate?
  - Is the use of those windows/buttons/colors appropriate?
  - Is the placements of menus and buttons appropriate and intuitive?
  - Is the program interactive enough?
  - Is the program user-friendly and intuitive?
  - Could the program interest and motivate you?
  - Which functionalities do you think are not useful and which new one should be introduced?
  - Have you used the direct connection between the program and its e-learning platform? Do you found it useful?
  - Do you need an additional *EasyStatics* manual or is the on-line help of *EasyStatics* sufficient?
  - Do you have suggestions for improvements?

- **DIDACTICAL ASPECTS**
  - Do you think the program is a valuable tool for better understanding structural behaviour?
  - Do you think the program is helpful for teaching structural behaviour?
  - Which additional functionalities can be introduced to improve students understanding?
  - Do you think to use the program again?

The feedback from the questions has allowed us to constantly adapt the program to the users’ needs and consequently to make it become an useful tool for learning and teaching purposes. However some improvements could still be made, e.g. undo and zoom functions.

Considering the didactic questions the response is generally positive. Teachers think that through the program and its functionalities (e.g. the immediate representation of the result after any parameter change, the possibility to generate more independent copies of
a model, the comparison between more models, the computational states for presenting different situations, etc.) they can focus on the key aspects of structural behaviour and through simulation they can better keep the students’ attention and motivate them. Generally, teachers who have used the program, see its benefits and are interested in adopting it also in the future.

Students find the program easy to use, appealing, intuitive and an helpful tool for understanding abstract concepts. They can learn how to model a physical structure, how section forces influence the model geometry and the section properties and improve their qualitative understanding figuring out, for example, how section forces diagrams are and checking their assumptions with the program results. Then, thanks to the easy generation of more structural alternatives and the possibility to compare them considering advantages and disadvantages, they can experiment what really means being a designer. Furthermore, particularly in the introductory course, they learn to work in a group, to discuss decisions, to critically interpret the computer results and to present and justify their ideas.

Finally, regarding the connection with the EasyStatics server, both teachers and students, have appreciated the possibility to work off-line and to be on-line only for downloading and uploading EasyStatics files and information.
Chapter 7

Conclusions and Future Developments

Some final considerations concerning the EasyStatics project are given below followed by suggestions for future developments.

7.1 Conclusions

A software and an Internet platform have been developed with the aim to provide undergraduate students in civil engineering and architecture with modern tools for learning structural design. These have been developed according to the constructivist theory (see Chapter 3), which is nowadays considered to be central to the learning process. The main purpose of EasyStatics is not only to enrich traditional structural analysis courses but, more importantly, to provide students with tools that better prepare them to meet the requirements of their future profession. With these thoughts in mind, many functionalities have been implemented (see Appendix) in order to make it an adequate tool for learning structural design.

However, the aim of the EasyStatics project was not to provide a ready-made course on structural design but only to be a useful tool to set up lectures, the content of which is the teacher’s task. Only some suggestions on how they can use the program to prepare exercises and demonstrations are found in chapter 5 together with minimal theoretical background information.

The experience we have been able to gain so far as well as the comments we have received from both students and teachers are positive.
7.2 Future Developments

To set up a structural design course using *EasyStatics*, the *Electronic Tutorials System* ([www.evim.ethz.ch](http://www.evim.ethz.ch)) developed at the Institute of Computational Science of ETH Zurich under the supervision of Prof. Hans Hinterberger is considered with interest. E-Tutorials are hypertext-based tutorials, where instructions are conceptually structured (*application guides*) so as to appropriately link theory, examples and step-by-step instructions (tutorials).

This structure is made in order to learn how to use an application software (in our case the *EasyStatics* program) to solve a problem and to develop skills while applying concepts. The interface is subdivided into three windows: one for running the application software, used for solving an assigned task, another with instructions to guide the users through an exercise and a third in which support is displayed (see figure 7.1).

![Figure 7.1: E-Tutorial interface.](image)

After finishing the tutorial students are expected to interact with their teachers, who should verify that concepts and theory have been correctly understood and applied in the right context. In such a way, the role of the teacher changes from information broker and example problem solver to personal coache. This helps students to learn more, because they transfer concepts directly into a practical activity. The method used, which has proved to be highly motivating for students in different disciplines, can be defined as *blended learning*, a mix of best old and new practices in education, where not the mixing but the right recipe is of importance [64]

The *application guides system* could also be used to set up structural design courses using *EasyStatics*. This represents the application software to be opened on the top left of the E-Tutorial interface (see Figure 7.1) and to be used to solve structural design exercises. Step-by-step instructions and some theory can guide students while carrying out the exercises (see Figure 7.1, right) and, if problems arise, show how similar tasks are solved (see Figure 7.1, bottom).
Chapter 7. Conclusions and Future Developments

We think this would be a promising way to extend the *EasyStatics* project from a teaching tool to an innovative course provider in the important field of structural design.
Appendix A

Appendix: *EasyStatics* Functionality

The *EasyStatics* software, which is based on the FEM (some theoretical details are given in chapter 5), is written in the platform-independent and object-oriented Java programming language. The program has been provided with several functionalities specially designed for teaching purposes which are explained in detail below. It is important to stress again here that the constructivist use of these (see section 3.4.2) is what is expected to make the program a valuable tool for teaching and learning structural analysis and design at the undergraduate level. Examples of how a teacher can use *EasyStatics* following a constructivist approach are found in section 3.5.

**A.1 Model Building**

Once the scale of the drawing area has been defined, the geometry of the model has to be entered. This is done by selecting the model button and clicking with the mouse on the screen so as to draw lines, which represent the bars of the plane frame or truss. Each line corresponds to a “finite element” (see Chapter 5).

After the geometry has been defined, a structure needs to be made stable through the introduction of supports. This is done by selecting the support buttons and clicking with the mouse on the nodes. Furthermore, by double-clicking on a support, an internal panel appears where the user can modify the direction of the support and enter a prescribed displacement or a rotation. The direction in which a support acts can be also changed using the mouse (see Section A.2).

Once a structure has been defined, it has to be loaded. This is done by selecting a load button and clicking with the mouse on the part of the structure where the load acts.

When the self-weight button is selected, this is automatically assigned to all bars.

The length of the graphical load objects is proportional to their value. This and the load direction can be changed by double clicking with the mouse on a selected load and
Figure A.1: Model buttons.

entering the desired value in the internal panel appearing on the screen. The load’s value and direction can also be modified using the mouse (see Section A.2).

*EasySatatics* allows one to enter loads which can belong to two (and only two) independent load cases. They can be distinguished by different colours and multiplied by the factors F1 and F2 for the 1st and 2nd load case, respectively (see Section A.8). When the results of the combination of the two load cases are shown, the user can see the superposition of the loads acting on the structure, but cannot modify them. To do this, each load case has to be considered separately.

A user can also introduce rotational hinges within the bars. This is done by selecting the hinge button and clicking with the mouse where the hinges have to be applied. No more than two hinges may be introduced on a single element. In fact, three or more of them would cause a local instability. Of course, global instability can still occur, but in this case the program will show how the structure would collapse (see Section A.6). For trusses, hinges are ignored, as truss bars are treated as frame bars with two hinges at the start and the end.

As soon as the elements are defined, material and section properties can be assigned to each of them (see Section A.8)
Appendix A. Appendix: EasyStatics Functionality

A.2 Moving Objects

An important feature of EasyStatics is the possibility to move (as well as add and delete) objects while performing any kind of analysis. By pressing and dragging the mouse, nodes can be repositioned, supports can be rotated, loads and hinges can be moved along the axis of the bars. In the same way, the magnitude and direction of every kind of load can be changed. This capability is particularly instructive. In fact, because the results appear immediately, i.e. in real time after any change, it is easy to understand the influence of a parameter on structural behaviour.

Consider, for example, a beam fixed at its ends and subjected to a distributed load (see figure A.2). When the mouse is moved on the upper part of the graphics object representing the distributed load, a small rectangular region appears. By pressing and dragging the mouse on this area, the length and the direction of the graphics object are continuously modified and associated with numerical quantities. If the compare function is active (see Section A.15), one can see how the immediately recomputed and displayed results change.

The possibility of clicking and dragging the mouse on the graphics objects, instead of editing in a text field the associated numerical values (see Section A.1), makes the program intuitive, appealing and game-like.

Figure A.2: Displacements of a beam built-in at its ends subjected to a varying distributed load.
A.3 Analysis Types

<table>
<thead>
<tr>
<th>1st Ord.</th>
<th>Stability</th>
<th>2nd Ord.</th>
<th>Rigid-Pl.</th>
<th>Dynamics</th>
<th>Infl. lines</th>
</tr>
</thead>
</table>

Figure A.3: 1st order, stability, 2nd order, rigid-plastic, dynamic and influence line buttons.

*EasyStatics* allows a user to perform the following analysis types, which will be treated in some detail in Chapter 5. These are considered to represent the basics of structural analysis at undergraduate level.

- Linear-elastic 1\textsuperscript{st} order theory, including influence lines
- Linear-elastic 2\textsuperscript{nd} order theory
- Elastic Stability
- Eigenmode dynamics
- Rigid-plastic theory based on the limit load theorems

Usually one first focuses on 1\textsuperscript{st} order elastic analysis. 2\textsuperscript{nd} order analysis helps to understand the effects of initial normal forces on structural behaviour (see Section 5.3). In both, 1\textsuperscript{st} and 2\textsuperscript{nd} order analysis, by using the stress field option, students can see the stress distribution as well as the compression, tension and total force resultants of the selected elements (see Section A.5).

One can also use the truss model option to obtain a truss model of a selected beam (see Section A.5), which is especially instructive for reinforced concrete structures.

Once a student has understood how a structure behaves in the static case, dynamics can be considered. *EasyStatics* can compute and display the four lowest eigenfrequencies and their associated modes.

Finally, one may be interested to know how much a structure can be loaded before collapsing and what the collapse mechanism would be. Rigid-plastic analysis provides this information.

In every analysis type, the possibility to show the results in real time after any change is considered to be of great importance in developing the necessary feel for structural behaviour.

A.4 Results

*EasyStatics* provides its users with several types of results. Displacements, bending moments, shear and normal forces, check lines, stress fields and truss model (see Section A.5) can be shown.
Bending moment, shear or normal force diagrams can be combined with displacements and the so called “check line” (see Section A.9). In order to see moments, shear and normal forces together, several copies of a model can be created (see Section A.14).

![Figure A.4: Displacements, bending moments, shear and normal forces, check line, stress fields and truss model buttons.](image)

When either the stress field or the truss model option is selected, the usual working area is replaced by a panel, where the corresponding output information is shown (see next section).

## A.5 Stress Fields and Truss Model

Specifically for didactic purposes EasyStatics allows two additional output options for structures analyzed by 1st or 2nd order theory. The stress field option shows, using a colour scale, the linear distribution of the axial stress $\sigma$ over the height of the section of one or more selected bars. From the linear stress distribution within the section, both the positive and negative stress resultants $R^+ > 0$ and $R^- < 0$ within the section as well the combined stress resultant $R^+ + R^- = \text{normal force } N$ can be computed. These resultants are displayed considering their eccentricities with respect to the axis of the bar. Students should observe that the resultants $R^+$ and $R^-$ always lie within the section of the bar, whereas the combined stress resultant $N = R^+ + R^-$ can lie outside the screen when, for small $N$'s, the eccentricity $M/N$ is large or, for $N = 0$, infinite.

In the second option, which helps to understand how reinforced concrete works, the truss model of a single plane frame element is built and displayed. As shown in Figure A.5 the element is modelled by several top and a bottom truss bars of equal length connected in the diagonal and transverse directions. The subdivision is such that the angle of the diagonals is approximately $45^\circ$. The height $y$ is preset to $y = 0.8h$, where $h$ is the section’s height. The orientation of the diagonal bars is such that, like concrete, they only act in compression. The bars in the transverse direction act in tension, like stirrups. According to statical equivalency criteria, the axial forces in the top and bottom bars are determined from the bending moment $M$, the normal force $N$ and the shear force $V$ in the middle of each field. The forces in the transverse bars are determined so as to fulfil nodal equilibrium.
Appendix A. Appendix: *EasyStatics* Functionality

A.6 Global and Local Instability

If, due to either insufficient supports or too many hinges, the structure is unstable, the instability mechanism is shown (see Figures A.6 and A.7).

While in some cases, as in Figure A.6, one immediately observes that supports are missing, in other cases (Figure A.7), particularly when hinges are introduced in a plane frame, it is more difficult to see the reason for the instability. Trusses can easily become unstable considering that they are treated as a frame but with hinges at the start and end of each bar.

Figure A.6: Instability mechanism.
A.7 Plane Frame or Plane Truss

EasyStatics offers the possibility to model the bars of a structure either as truss or as frame elements (see Figure A.8) and to switch at any time between the two models. Truss elements are treated as frame elements with hinges implicitly introduced at their start and end sections. Additional hinges, if any, are ignored.

Real life structures never consist of hinge-joined bars only. This assumption was introduced by Karl Culmann (1821-1881) in order to simplify manual computations for structures built so as to act mainly in tension and compression. As such it makes no sense today. Not for practical, but for didactic purposes, however, Cullmann’s assumption is still valuable (see also the exercise suggested in the Section 5.2.4).

A.8 Section Dimensioning

The section panel activated by the section button is used to assign each structural member one of the following section types and materials:

- steel profiles of the standard types IPE, HEA, HEB and HEM as well as half profiles IPET, HEAT, HEBT;
- steel tubes;
- rectangular, I- and T-shaped-concrete sections with top and bottom reinforcement
- wooden rectangular sections.

Simply with mouse clicks, the types of the steel profiles and the dimensions of steel tubes, concrete and wooden sections can be chosen at will.
The section panel (see Figure A.9) shows on the left the geometry of the section, on the right the so-called *M-N interaction diagram*. This allows one to assess the section resistance of an element against any combination of the bending moment (M) and the axial force (N). The M-N interaction diagram is represented by a green polygonal shape with 8 vertices in an M-N coordinate system. If M and N in a selected section are inside the green polygon, the section resistance is considered as capable of withstanding these section forces. Otherwise the section strength is exceeded.

The M-N coordinates of one vertex are determined by considering a corresponding strain distribution. From this, the normal stresses can be found and consequently the values M and N of the vertex.

For steel and reinforced concrete sections ideal rigid plastic material behaviour is assumed (see Section 5.5). The section stresses above and below the neutral axis across the section are therefore uniformly distributed and equal to either the compressive or the tensile yield stresses of the material (see Figure A.10). The tensional resistance of the concrete is assumed to vanish.

In the case of a wooden section linear-elastic material behaviour is assumed. In this case, the M-N interaction diagram is built with the assumption that the linearly distributed
Appendix A. Appendix: EasyStatics Functionality

Figure A.9: Section panel.

Elastic normal stresses $\sigma$ reach the limit tensile $f_T$ and/or compressive $f_C$ stresses only at the upper and/or lower edges of the section.

Concerning the strain distribution over the section, different assumptions are made depending on the section shape. The strain and the rigid-plastic stress distribution for I-profiles of steel are shown in Figure A.10: strain distribution as a line diagram, compression in dark gray, tension in light gray.

According to the building codes, particularly in the case of high sections such as steel profiles, the resistance to bending moments and normal forces should be reduced. This is ascribed to buckling and lateral buckling phenomena, which are due to the slenderness of the structural members. In the section panel the factors $\kappa_N \leq 1$ (buckling) and $\kappa_M \leq 1$ (lateral buckling) can be entered for this purpose (defaults: $\kappa_N = \kappa_M = 1$). A red line shows how the surface enclosed in the M-N interaction diagram decreases for $\kappa_N < 1$ or $\kappa_N < 1$.

To properly dimension a section it has to be ascertained that the “design” section forces $M_d$ and $N_d$, due to a combination of the load cases 1 and 2, lie inside the allowable green polygon of the interaction diagram. The load combination is obtained by multiplying the section forces of the load cases 1 and 2 (if self-weight is considered, it belongs to load case 129.
1) by the load factors F1 and F2 specified by the program user:

\[ M_d = F_1 \times M_1 + F_2 \times M_2 \]
\[ N_d = F_1 \times N_1 + F_2 \times N_2 \]

where \( M_1, N_1, M_2 \) and \( N_2 \) represent the bending moments and the normal forces due to the load cases 1 and 2, respectively.

By keeping the section window open, one can see whether the bending moments and the normal forces in the selected section are inside or outside the area enclosed in the M-N interaction diagram. If necessary, the section dimensions and/or the reinforcement can be changed until an adequate resistance is obtained.

In addition to \( F_1, F_2, \kappa_N \) and \( \kappa_M \) the following values found in the section panel, can be modified:

- in the case of a steel section, the yield stress \( f_Y \) in both tension and compression;
- in the case of a concrete section, the yield stress \( f_Y \) in tension and compression of the steel bars and the compressive strength of the concrete \( \beta_C \);
- in the case of a wooden section, the allowable normal stress in tension, \( f_T \), and in compression, \( f_C \);
- the Young’s modulus \( E \) and the weight of the bar per unit meter;
- the number of steel bars and their diameter in the upper and lower part of a reinforced concrete section.
A.9 Check Line

With the so-called check line EasyStatics also allows one to judge at a glance if and where a structure is over- or under-dimensioned for the chosen load combination. This line diagram represents the value of the check factor along all bars of the structure. Where the carrying capacity of the sections has been exceeded, the line is red, otherwise is blue.

The figure A.11 shows how the value $c = a/b$ of the check factor is determined. The distances $a$ and $b$ are measured perpendicular to the relevant side of the interaction diagram, where each side represents an inequality constraint for $M$ and $N$. If $c$ is smaller than 1, the point $(M_d, N_d)$ lies inside the polygonal shape of the interaction diagram (blue check line), otherwise with $c > 1$ the section forces do not satisfy the plasticity conditions of the section (red check line). Its distance from the axis of the bars shows the amount by which the section is under- or over-dimensioned. Numerical values of the check factor can also be displayed (see section A.11).

This unique feature of the EasyStatics program has been proved to be extremely useful for assessing at a glance the quality of different design proposal.

![Figure A.11: Determination of the check factor.](image)

A.10 Customizing the Graphical User Interface

EasyStatics allows its users to personalize the graphics interface in three ways:

- by selecting the language in which texts are written (one can choose between English, German, French or Italian);
- by changing the colours of the graphical objects (loads, supports, elements, nodes, displacements, normal forces, shear forces and bending moment diagrams, etc.) and
Appendix A. Appendix: *EasyStatics* Functionality

- by hiding all unnecessary menus and buttons.

The first two points are related to the appearance of the program, so that users can work in a “customized” environment that they will prefer. The last point is related to teaching. The possibility of hiding buttons and menus allows teachers to set up their courses providing students only with those functions they deem necessary to perform a suggested task. This avoids students being distracted by other features, the meaning of which is still unknown or which are not related to the current exercise or demonstration.

For example, if students start to learn how a simple structure can be modelled and analyzed by 1st order theory, the teacher can create demonstrations and exercises so that only the button of the 1st order analysis together with some results and model buttons are visible.

### A.11 Show and Freeze Numerical Values

*EasyStatics* not only allows one to see the graphical representation of displacements, section forces, check lines, support reactions, loads, etc., but also the numerical values associated with them. These are automatically shown whenever the mouse is moved over the corresponding graphics objects. To “freeze” the labels where the numerical values are written, the mouse is to be clicked. This is an important feature when one wants to explain some results in a text document containing pictures extracted from the program (see Figure A.12).

![Figure A.12: Picture extracted from *EasyStatics*.](image)

### A.12 Sketching the Model on a Background Image

*EasyStatics* allows one to display an arbitrary picture on the background of the computer screen. With an internal browser, the user can choose such a picture which is to be saved locally on the computer (see Figure A.13). This option is intended to allow a teacher to show students a real structure, and based on it, to build its model. In fact, it is important that students correlate abstract models to the reality they represent.
A.13 Help Functions

*EasyStatics* helps its users by means of explanatory texts in the chosen language appearing on the screen.

When a button is selected by clicking on it, a short self-explanatory text appears (see Figure A.14). At the same time, at the bottom of the screen a text explains more in detail the functionality of the button or how a user has to deal with the graphics object associated with it. *EasyStatics* also helps its users by providing text written directly on the drawing area.

A.14 Copies of a Model

For didactic purposes it is important to compare different design alternatives. *EasyStatics* offers the possibility of generating up to four copies of a model on the computer screen. Each of them is independent of the others, which means that any change to one model does not influence the others. To see how the copy option can be used, the exercise in Section 5.2.4 can be considered.

A.15 Compare Function

When the compare function is active, the scale of the results is not recomputed after a change. This functionality, which extends to all models shown on the screen, allows a
user to see how changes influence structural behaviour.

To see how the compare option can be used, the exercise in Section 5.2.4 can be considered.

### A.16 Internal Editor

*EasyStatics* offers the possibility of writing comments on the computer screen. This is done by opening one or more internal text areas (see Figure A.15) functioning as simple editors. A text can be written using different colours, fonts and sizes. When an *EasyStatics* states-file is saved, everything on the drawing area, including comments, is saved. This is useful when teachers want to provide students with exercises and demonstrations. Students can, in the same way, explain their results and motivate their design choices.
Figure A.15: Internal editor with some questions
A.17 Computational States

What appears on the display of *EasyStatics* as well as the corresponding data, builds a so-called *computational state*. The program allows one to generate and store several independent computational states. They are progressively numbered so that one can navigate between them. It is an important, and possibly unique, feature of *EasyStatics* that these states are not static pictures. On the contrary, one can normally work in each computational state. This means that elements, loads, supports and hinges can be added, deleted and moved, section properties and material can be modified, different analysis types can be performed, models can be created or deleted, comments can be introduced or modified, etc.

When an *EasyStatics* document is saved, this includes all defined computational states. Such states files allow teachers to prepare exercises with different computational states, which can be later modified by the students as an exercise.

A.18 Exporting Pictures

Selected rectangular areas of the screen can be isolated and exported as pictures, which can later be introduced into a generated HTML document as discussed in the next section, or in any text document where pictures can be inserted, e.g. MS-Word documents.

A.19 Internal HTML Editor

*EasyStatics* offers the possibility of activating a simple HTML editor (*SyplyHTML*) directly from the program. The editor can be used by the teacher to prepare text documents containing text and pictures created in the way explained in Section A.18. Such text files offer the advantage of providing reports of demonstrations and exercises. In fact, this possibility represents the main output capability of the program. It can help students to experience the real work of civil engineers, who have to write reports, explaining their assumptions and structural decisions and document the results obtained.

Reports can obviously also be written using any text document, like MS-Word, where *EasyStatics* pictures are included. The decision to include the HTML editor is connected to the idea of providing users with a complete package and to create a text with the most common Internet format.
A.20 Connection with the EasyStatics Database Server

The program EasyStatics is embedded in an e-learning platform (see chapter 4) where teachers and students can communicate by downloading and uploading different types of documents, including EasyStatics files.

In order to send and open EasyStatics demonstrations and exercises without leaving the program, i.e. without passing through the e-learning platform, EasyStatics offers a registered user the possibility of connecting directly to the server. Depending on whether users are teachers or students, they can send and open an EasyStatics states-file with the same restrictions explained in Chapter 4. For this purpose, as soon the menus send document on server or open document from server are selected, an internal window appears with a list of all the members of the same course, so that one can choose the person to send an EasyStatics file to or from whom to receive one. If a user has teacher permissions, he or she can send an EasyStatics file to the whole group. Not only EasyStatics files can be sent from the program to the server but also documents of any other format.
Bibliography


[77] Kurt Gramoll and Rob Abbanat. Interactive multimedia for engineering statics. Aerospace Engineering, Georgia Tech, Atlanta, GA.


