A Non-Linear System for Semi-Rigid Steel Portal Frame Analysis

A. A. Del Savio^{*}, S. A. L. de Andrade[^], P. C. G. da S. Vellasco⁺ and L. F. Martha[^]

* PhD Student, Civil Engineering Department,

PUC-RIO - Pontifical Catholic University of Rio de Janeiro, Brazil.

[^] Associate Professor, Civil Engineering Department,

PUC-RIO - Pontifical Catholic University of Rio de Janeiro, Brazil.

[†]Associate Professor, Structural Engineering Department, UERJ - State University of Rio de Janeiro, Brazil.

Abstract

This paper describes in detail a structural system, FTOOL/SRC, conceived to perform non-linear analysis of semi-rigid steel frames. The system can be used to validate the semi-rigid connection approaches by means of a parametric analysis in a simple and compact form. It is based on an efficient graphical interface and efficient external solver FEMOOP [2]. A connection finite element, developed by Del Savio [1], was implemented enabling plastic hinge analysis to be performed on the structural connections and bars. The main objective of this paper is to demonstrate, through a series of validating examples, the performance of the FTOOL/SRC system and its implemented strategies, focusing on the newly conceived object oriented data structure.

Keywords: structural engineering, semi-rigid connections, steel structures, nonlinear analysis, connection model, object oriented programming philosophy.

1 Introduction

The continuous search for a more realistic structural modelling has long been pointing out to an appropriate consideration of the related non-linear effects that significantly affect the structural behaviour. Various analysis techniques, with different refining strategies and associated complexities, can be adopted according to the non-linear response of a structure model. Hence a suitable modelling analysis can be selected for each required structural behaviour type.

Generally non-linear effects are related to the structural elements geometrical and material properties. Recent investigation trends indicated that the non-linear semirigid connection response also played a fundamental role in the structural behaviour. This paper will describe in detail a structural system, FTOOL/SRC, developed from an earlier program, FTOOL [3] (Two-dimensional Frame Analysis Tool Program) to perform non-linear analysis of semi-rigid steel frames.

The FTOOL/SRC system enables parametric analysis to validate the semi-rigid connection use to be performed in a simple and compact form due to the adoption of an efficient graphical interface and to the use of an external solver FEMOOP [2] (Finite Element Method – Object Oriented Program). The implemented interface combines, in a user-friendly environment, the pre-processing, structural analysis (FEMOOP [2]) and the post-processing phases.

A connection finite element, Del Savio [1], was implemented in the earlier, FEMOOP [2], system and is employed for the non-linear structural semi-rigid analysis. This element enables a plastic hinge analysis to be performed on the structural connections and bars for any type of loading.

The most significant developments of the FTOOL/SRC system were related to three fundamental issues: First the implementation of its graphical interface, enabling non-linear analysis and the semi-rigid connection data input to be performed. Second an evolution from the previous data structure, centred on a complete topological planar subdivision representation, HED (Half-Edge Data Structure), to a new object oriented data structure. Third the implementation of a connection finite element, Del Savio [1], in the FEMOOP system [2].

2 Implementations

2.1 FEMOOP [2]

The program FEMOOP (Finite Element Method – Object Oriented Program), created in 1991, was based on the object oriented philosophy associated to the finite element method. Since the program have been widely used enabling its expansion to solve thermal analysis, non-linear analysis and others.

2.1.1 Finite Element Programming, Martha [2]

Before presenting the class organization of the FEMOOP program it is important to note that the computations carried out in a nonlinear finite element analysis occur at three distinct levels: the structure level, the element level, and the integration point level. The structure (or global) level corresponds to the algorithms used to analyse the problem (e.g., linear static, nonlinear path-following, nonlinear dynamic, etc.). These algorithms are implemented in terms of global vectors and matrices, and do not depend on the types of elements and materials used in the analysis.

The main task performed at the element level is the computation of element vectors and matrices (e.g., internal force vector and stiffness matrix) required to assembly the global vectors and matrices, which are used by the analysis algorithms. The computation of these vectors and matrices is completely independent of the algorithm used to (globally) analyse the model.

The communication between the global and the element level occurs in two directions. The upward direction corresponds to the computation of the global vectors and matrices summing up the element contributions, and the downward direction corresponds to the extraction of the element displacements from the global displacement vector. These communications tasks are carried out using nodal degrees of freedom and element connectivity.

Finally, the computation of stress vector and tangent constitutive matrices is carried out at the integration point level. These quantities are used in the computation of the element vectors and matrices, but they do not depend on the element formulation, provided that the basic input data for stress computation are strain components.

2.1.2 Overall Organization

The overall class organization of the FEMOOP system is depicted in Figure 1.



Figure 1: Overall class organization, Martha [2].

The Control class is an abstract base class that provides a common interface for solving algorithms. The current hierarchy of the *Control* class is shown in Figure 2.



Figure 2: Control class hierarchy, Martha [2].

The Finite Element Model class represents the numerical discretization of the model into finite elements. The main tasks of the *Finite Element Model* class are to compute the nodal d.o.f., to assemble the global vectors and matrices used by the analysis algorithms, to update nodal displacements, and to print the computed results after convergence.

The Node class basically stores the nodal data read from the input file (coordinates, etc.), as well as some variables computed during the program execution, as the nodal d.o.f. and the current displacements.

Material is an abstract base class that provides a generic interface to handle the different materials (derived classes) available in the program, including some elastic and elasto-plastic materials.

Element is an abstract base class that defines the generic behaviour of a finite element. The main tasks performed by an object of the *Element* class are the indication of the number and direction of the active nodal d.o.f., the computation of the element vectors (e.g., internal force) and matrices (e.g., stiffness matrix), and the computation of the element responses (e.g., stresses). The OOP hierarchy of the *Element* class is partially illustrated in Figure 3.



Figure 3: Element class hierarchy, Martha [2].

The Shape class holds the geometric and field interpolation aspects of the element (dimension, shape, number of nodes, etc.), while the *Analysis Model* class handles the aspects related to the differential equation that governs the problem to be solved.

The Integration Point object holds the parametric coordinates and the corresponding weight used for the numerical integration, while the *Constitutive Model* is an abstract base class that provides a common interface to the different constitutive relations implemented in the program.

Finally, the *Load Element* class was created to allow the generic consideration of natural boundary conditions and body forces.

2.1.3 Software Implementation

In order to include the connection finite element (Del Savio [1]) in the FEMOOP [2] program structure the only required steps were: creation of a new class on the Element class hierarchy (Figure 4) and add a new procedure on the file responsible for reading the data inputs. It should be pointed out all the required information needed for the new element was included on its class without altering any other part of the program code. The connection finite element implemented was based on a 2D non-linear co-rotational beam element.



Figure 4: New element class hierarchy.

2.2 FTOOL [3]

The FTOOL [3] (Two-dimensional Frame Analysis Tool Program) was conceived in 1991, from a research project of the Computer Graphics Technology Group (Tecgraf/PUC-Rio) and is based on a function library HED (Half-Edge Data Structure), Cavalcanti [4], for the internal data representation.

Over the last few years FTOOL [3] have demonstrated to be a valuable tool for structural engineering teaching being used on Structural Analyses, Concrete and Steel Design courses on various civil engineering programs of Brazilian and foreign universities. FTOOL [3] is a structural analysis system possessing, in a single platform, all the necessary tools for an efficient model pre and post processing and a fast solving strategy.

2.3 FTOOL/SRC [1]

The newly implemented FTOOL [3], version named FTOOL/SRC (Two-Dimensional Frame Non-Linear Analysis Tool Program Incorporating Semi-Rigid Connections), is focused on solving linear and non-linear semi-rigid steel portal frames inheriting all the FTOOL [3] functionalities.

The present implementation was centred on: the substitution of the previous data structure (HED, Cavalcanti [4]) for a new object-oriented data structure; enhancing the graphical interface made with IUP [5] and CD and removing the solver algorithm from the internal structure of the FTOOL [3] program. Due to the inherent object-oriented philosophy advantages the FTOOL [3] modules were created using the C++ language centred on maximizing the program code reuse, and conceiving generic procedures.

2.3.1 Data Structure

FTOOL [3] is based on a data structure centralized on a topological representation of a planar subdivision. It incorporates an efficient adjacent information data search and also being capable of acting as a data structure organizer. The topologic data structure HED (Half-Edge Data Structure), used in the FTOOL [3] program is fully described in Cavalcanti [4].

Figure 5 illustrates the HED data structure communication. The grey and white rectangles represent, respectively, the main registry names and the pointer variables relating the entities. Figure 6 describes the FTOOL [3] (version 2.11) data structure communication responsible for registering the entity attributes in the HED data structure.

The main reason for changing the FTOOL [3], data structure, the HED, was the challenge to create a new simpler data structure, implemented according to the object-oriented philosophy, possessing the same functionalities offered by the HED for beam elements but being capable of dealing with 3D structures, since the HED is limited to 2D elements.



Figure 5: HED Data structure communication.



Figure 6: FTOOL [3] and HED entity attributes data structure communication.

The object oriented philosophy main advantages are: easy code maintenance and program developments, reduction of the probability of error occurrence and the minimum impact caused by new code implementations on the existing code. Additionally the code reuse also leads to a widely adoption of generic procedures.

The FTOOL/SRC new data structure communication is depicted in Figure 7. The grey and white rectangles represent, respectively, the class names and the interrelating pointer variables. The class relations, grey-to-grey rectangles indicate

an inherence relation. For instance the Node class inherits the NodalSupport, NodalDisp, NodalSpring and Point classes.



Figure 7: FTOOL/SRC Data structure communication.

The fundamental characteristics of the main implemented classes will be described on this paper. Further details are present in Del Savio [1].

The *DoubleGenericLinkedList* and *Elem* classes possess methods and variables to manipulate a double connected generic list and are responsible for storage and control of all program data. Every class that uses this double connected generic list should declare a pointer to the *Elem* class optimising the search for data stored on the list. This pointer is the generic element memory address where the data is stored. This strategy eliminates time-consuming data searches, element by element, by directly accessing it.

The FTOOL/SRC program contains several lists: materials, section properties, etc. These lists presented common functionalities that were generalized in the *BaseList*, basic class. This class defines the generic behaviour of the subsequent classes implemented in the program including: MaterialParameters, SectionProperties, among others. This strategy enables newly created classes from the *BaseList* class only to be concerned with their specific and particular methods.

The *Member* class reunite all the required variables and methods for the beam element manipulation storing the nodes adjacent information. The *Node* class possesses all the required variables and methods for the manipulation of a node object. This class contains a list of bars adjacent to a node allowing an efficient adjacent information data search. The *Model* class stores a pointer for the main

FTOOL/SRC classes to enable access to any program class. When the program starts this is the first object to be defined.

2.3.2 Graphical Interface

The new FTOOL/SRC system incorporated new graphical user interface elements in FTOOL [3] program using IUP [5] elements and functions and LED [5] interface elements.

Figure 8 illustrates some of the new graphical interface features implemented for the semi-rigid connections pre-processing: the spring element representing the structural joint as well as its associated initial stiffness; the creations of a connection list (connection elements); the moment versus rotation curve describing the joint structural behaviour, by its coordinates, Figure 9, and finally the creation of buttons for defining the connections on a beam element.



Figure 8: FTOOL/SRC: Semi-rigid connections pre-processing menu.



Figure 9: Connection moment versus rotation curve input menu.

The next step was the implementation of the various types of structural analysis executed on the FTOOL/SRC, Figure 10. The user can choose: linear or non-linear analysis, co-rotational 2D beam elements and co rotational or plane connection elements.

The solving algorithmic includes Newton-Raphson, modified Newton-Raphson (Load Control), Displacement Control, Generalized Displacement Control, Energy Control, Deformation Control, Arc Length and Orthogonal Residue controls. Non-

linear post processing needs required the creation of a C++ program to visualize load versus displacement curves, Figure 11.

Units & Number Formatting	e Selection 11	Diagram	mbination: Single Cas T + + + + 14 (či Star: 300 m - hu
		<u> </u>	Rotation Release
Renofic Renofic Bane 300 (* Plane Costation Frane / Linear Correction Plane Frane / Correction (* Plane Costation Frane / Linear Correction)	C Linear Elasti Elements C Beam 30 / Hybrid Beam 30 C Plane Frame / Connection	C Nor Plane Corotation Fin Plane Corotation Fin	Linear Elastic
garban Nenton Radioum Diplement Generalized Displacement Strain Ubdeed Drisgong Ubdeed Drisgong Ubdeed Drisgong Ubdeed Training Tubget general Ubdeed Training Ubdeed Training Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubdee Ubd	Algorithms Algorithms Payloc Rightoon Diplacement Generalized Displacement Stain Initially Dirthogonal Updated Othogonal Clindrical Modified Spherical	Parameters Maximum steps: Step lactor: Tolerance: Maximum interations: Pirk Step: Orsplacement Update G. Renewed	0.1000 1.00+00 10
Program Defaults Dk Cancel	Program Defaults		Cancel

Figure 10: FTOOL/SRC: Analysis configuration menu.

2.3.3 FEMOOP [2] solver use

FTOOL [3] uses an internal solver, FRAMOOP, that is a simplified version of the FEMOOP [2] (in C language, 1991) adequate for linear analysis. In the new strategy, the FTOOL/SRC program was responsible for the pre and post-processing phases, while the FEMOOP [2] program performs the structural analysis. This strategy helps future implementations and independent code maintenance in both programs.



Figure 11: Load factor versus node deflection curves.

3 Case Studies

This section objective is to validate, by means of comparisons to literature results, the implemented connection finite element, Del Savio [1]. First a steel portal frame study made by Steenhuis et al [6] will be presented. This will be followed by comparisons with non-linear analysis of semi-rigid portal frames performed by Keulen et al. [8].

3.1 Single storey semi-rigid frame linear analysis

Figure 12 depicts a steel portal frame investigated by Steenhuis et al. [6] to evaluate, on a non-sway structure, the effects of the semi-rigid joints over the frame internal force distribution and associated deformations. The portal frame span and height are six and four metres, respectively. An IPE360 steel profile was used on the beam and columns. The loading configuration consisted on a horizontal load of 25 KN and a uniformly distributed load of 40kN/m. The portal frame deflected shape and associated bending moment distribution is present in figure 12.

The portal frame was modelled using half of the joint tangent stiffness as the joint secant stiffness, as suggested in Eurocode 3, [9]. The adopted joint stiffness were 35kNm/mrad, 60kNm/mrad, 130kNm/mrad and a fully rigid joint.



Figure 12: Portal frame loads, deflected shape and bending moments, [6].

In this example a second set of results produced by Brito [7], for the same frame will also be presented. This investigation was centred on the proposal and validation of a simplified model for semi-rigid beam to column connections using the Ansys program. The authors used a linear joint stiffness and performed a non-linear elastic portal frame analysis.

The portal frame results are expressed in terms of a moment versus joint stiffness graph, figure 13 for the upper left corner (M_A), beam centre span (M_B) and upper right corner (M_C), respectively. Horizontal and vertical displacement versus stiffness curves for the upper left corner (d_h) and beam centre span (d_v), are also illustrated on Figure 14.



Figure 13: Bending moment comparisons.



Figure 14: Horizontal and vertical displacement comparisons.

Generally the bending moment and displacements obtained with the present formulation were similar to Steenhuis et al. [6] results and better than Brito [7], demonstrating the adequacy of the proposed connection element implementation.

3.2 Single storey semi-rigid frame non-linear analysis

Comparisons were made with Keulen et al. results [8], using full moment versus rotation curves ad a bi-linear approximation. This approximation uses a well-known simplified joint representation named half initial secant stiffness method. A steel portal frame spanning 7.2 metres and having 3.6 metres of height, figure 15, was investigated, [8]. The beam and column sections used, respectively, IPE360 and HEA260 steel profiles. The adopted joint and element moment versus rotation curves are presented in figure 16. The beam to column connections are bolted flush endplates, Figure 15.

The horizontal load α F represents wind load, but is also used to take into account imperfections affecting the second order effects. The α -factors are taken as 0.1, 0.15, 0.2, 0.3 and 0.5. The vertical load w is equal to 1/6 of the vertical load F.



Figure 15: Single storey frame and associated beam to column joint details [8].



Figure 16: Moment versus rotation curves [8].

The curves illustrated in Figure 17 represent the results of the three earlier mentioned analyses. The first two curves were performed by Keulen et al. [8] while the third used the proposed connection elements, Del Savio [1]. The Keulen et al. [8] analysis comprised:

- *Reference Analysis* was analysed using ANSYS version 5.5. To obtain the reference solution, a second-order elastic-plastic frame analysis is used. Plastic hinges in the beams are modelled using rotational spring elements at locations where plastic hinges are expected to occur. These rotational spring elements have a rigid-plastic characteristic neglecting the influence of normal and shear forces on the plastic moment capacity. Figure 16 illustrates the spring properties used for the base joints, eaves joints and the beam springs.

- *Half Initial Secant Stiffness Approach*, modelling the portal frame using a bilinear moment versus rotation curve considering half of the joint secant initial stiffness.

It can be observed in Figure 17, for the considered values of the α factor (0.1, 0.15, 0.2, 0.3 e 0.5), that the displacements obtained with the proposed formulation

are very similar to Keulen et al. [8] results demonstrating, one more time, the adequacy of the proposed connection element, Del Savio [1].



Figure 17: Load versus deformation diagram.

4 Final Remarks

Linear and non-linear analysis methods as well as a connection finite element for semi-rigid portal frames proposed by Del Savio [1] were implemented with success using the numerical solver present in the FEMOOP [2] program.

One of the main contributions of this newly created formulation was related to its data structure implemented in the C++ programming language, according to the object-oriented philosophy. Some other implementation key points deserve to be mentioned:

- The fully integration of the FTOOL/SRC program to other related systems.

- The program easy maintenance and expansion.

- The possibility of programming code reuses enabling the use of generic procedures.

Finally the implemented graphical interface, based on a simple and efficient user interface for the mentioned structural analysis also deserved to be mentioned.

In order to validate the proposed formulation, some structural models present in literature were utilized. These comparisons lead to the following considerations:

In Steenhuis et al. [6] and Brito [7], semi-rigid portal frame structures the current formulation lead to very similar results when compared to Steenhuis et al. [6] and better results than Brito [7].

The modification of the connection element stiffness matrix, Del Savio [1], when used in a non-linear structural analysis produced results close to the *Reference Analysis* [8], (second order elasto-plastic analysis), and better than the *Half Initial Secant Stiffness Approach* [8].

References

- [1] A.A. Del Savio, "Modelagem Computacional de Estruturas de Aço com Ligações Semi-Rígidas", MSc. Dissertation, Civil Engineering Department – PUC-Rio, Brazil, (in Portuguese), 2004.
- [2] L.F. Martha, E. Parente Jr., "An Object-Oriented Framework for Finite Element Programming", Proceedings of the Fifth World Congress on Computational Mechanics, IACM, Vienna, Austria, Jul. 2002.
- [3] L.F. Martha, "FTOOL: A Structural Analysis Educational Interactive Tool", Proceedings of the Workshop in Multimedia Computer Techniques in Engineering Education, Technical University of Graz, Austria, 51-65, 1999.
- [4] P.R. Cavalcanti, P.C.P. Carvalho, L.F. Martha, "Criação e manutenção de subdivisões planares", IV Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens, São Paulo, SP, 13-24, 1991.
- [5] C.H. Levy, L.H. Figueiredo, M. Gattass, C. Lucena, D. Cowan, "IUP/LED: A Portable User Interface Development Tool", Software: Practice & Experience, 1995.
- [6] M. Steenhuis, N. Gresnigt, K. Weynand, "Pre-Design of Semi-Rigid Joints in Steel Frames", Proceedings of the Second State of the Art Workshop on Semi-Rigid Behaviour of Civil Engineering Structural Connections, Prague, Czech Republic, 131-140, 1994.
- [7] O.F. Brito, "Desenvolvimento de Sistemas Estruturais Semi-Rígidos em Aço e Mistos para Edificações Residenciais Multi-Familiares", MSc. Dissertation, Civil Eng. Depart. – PUC-Rio, Brazil, (in Portuguese), 2001.
- [8] D.C. van Keulen, D.A. Nethercot, H.H. Snijder, M.C.M. Bakker, "Frame analysis incorporating semi-rigid joint action: Applicability of the half initial secant stiffness approach", Journal of Constructional Steel Research, v. 59, 1083-1100, 2003.
- [9] Eurocode 3, ENV 1993-1-1:1992/A2, Annex J, *Design of Steel Structures Joints in Building Frames*, CEN, European Committee for Standardisation, Document CEN/TC 250/SC 3, Brussels, 1998.