Software Requirements Specification for Truss Tool: A Tool for Truss Analysis

Maryam Valian

February 20, 2023

Contents

Revision History

Date	Version	Notes
2023-01-30	1.0	Initial version of the SRS
2023-01-10	1.1	Modification according to the feedback
2023-02-20	1.2	Updating according to SRSfdbk.pdf

1 Reference Material

This section records information for easy reference.

1.1 Table of Units

Throughout this document, SI (Système International d'Unités) is employed as the unit system. In addition to the basic units, several derived units are used as described below. For each unit, the symbol is given followed by a unit description and the SI name.

symbol	**unit**	**SI**
m	length	metre
N	force	newton
deg	angle	degree

1.2 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units. The choice of symbols was made to be consistent with the structural statics literature and with existing documentation for the truss analysis problem. The symbols are listed in alphabetical order.

symbol	**unit**	**description**
F_{i}	N	External force of joint i
F_{xi}	N	Force component in the x direction of joint i
F_{yi}	N	Force component in the y direction of joint i
M_{i}	Nm	Moment component of joint i
S_{p}	-	Pin support
S_{r}	-	Roller support
θ	deg	Angle between two members

1.3 Abbreviations and Acronyms

symbol	description
A	Assumption
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
T	Theoretical Model

1.4 Mathematical Notation

In this document, we do not use any specific mathematical notation.

2 Introduction

A truss is a structure that consists of members organized into connected triangles to enable the distribution of loads and forces. Trusses are most commonly used for wide spans like bridges, and roofs. Figure 1 shows a bridge built with planar trusses in 1969 called the Burlington lift bridge. Truss Analysis shows whether the external forces are well-distributed among the members or not.

We introduce Truss Tool that helps engineers analyze a given planar truss. The following section provides an overview of the Truss Tool's software requirements specification (SRS). In this section, first, we explain the purpose of the document. Then we describe the scope of the requirements, the intended reader's characteristics and the document's organization.

Figure 1: Burlington lift bridge: an example of the planar truss

2.1 Purpose of Document

The primary purpose of this document is to outline the software requirements of the truss analysis tool. To provide a good understanding of the system, different aspects of the system such as goals, assumptions, theoretical models, and definitions will be explained. The following SRS document will remain abstract exploring what is being solved rather than how it will be solved.

The following document will describe the system context and constraints, the specific problem definition and solution characteristics, requirements and likely and unlikely changes for the development of the tool.

2.2 Scope of Requirements

The scope of the requirements includes the analysis of the two-dimensional trusses where all members and nodes lie within a two-dimensional plane. For more details, you can also see the assumptions section (Section 4.2.1).

2.3 Characteristics of Intended Reader

Reviewers of this documentation should have a basic understanding of software development and structure statics and high school physics and high school Mathematics. The users of the Truss Tool must have a higher level of expertise, as explained in Section: User Characteristics (Section 3.2).

2.4 Organization of Document

The organization of the document follows the template for an SRS for scientific computing software proposed by Smith and Lai [3]. The template will present the system's goals, theories, definitions, and assumptions. Readers interested in top-down reading can begin by reading the system's goal statements (Section 4.1.3). Subsequently, the theoretical models will elaborate on the goal statements. Lastly, readers can finish with a more understanding of the system by reading instance models of the system.

3 General System Description

This section provides general information about the system. It identifies the interfaces between the system and its environment, describes the user characteristics and lists the system constraints.

3.1 System Context

Figure 2 shows the system context. The circles represent a user that interacts with the software. The rectangle represents the software system for the truss tool. The arrows display the input data from the user and the output data that is useful for the user.

The interaction between the product and the user is through a user interface. The responsibilities of the user and the system are as follows:

- User Responsibilities:
  - Provide truss geometry, supports and external forces.
  - Ensure the input data describes a valid truss.
- Truss Tool Responsibilities:

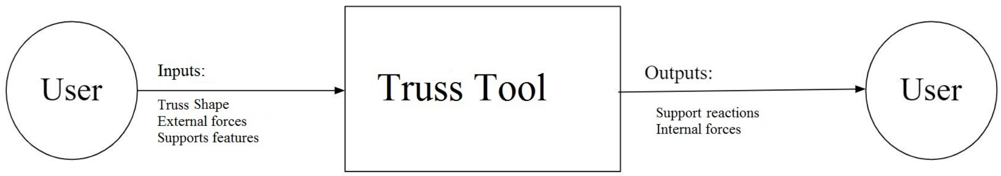

Figure 2: System Context

- Detect data type mismatch, such as a string of characters instead of a floating point number.
- Calculate external forces and support's reaction.

3.2 User Characteristics

The end user of Truss Tool should be an architecture/civil/mechanic engineer or should have an understanding of undergraduate Level 1 structural analysis.

3.3 System Constraints

There are no constraints on the development of the Truss Tool.

4 Specific System Description

This section first presents the problem description, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the assumptions, theories, definitions and finally the instance models.

4.1 Problem Description

Truss Tool is intended to solve a given truss with given external forces. By solving a truss, we mean that we are interested to calculate all internal forces among the members and the reactions of the supports. As a result, Truss Tool will help engineers to make a decision on whether the design of the given truss is proper or not.

4.1.1 Terminology and Definitions

This subsection provides a list of terms that are used in the subsequent sections and their meaning, with the purpose of reducing ambiguity and making it easier to correctly understand the requirements:

- Planar truss: A planar truss is one where all members and nodes lie within a two-dimensional plane.
- Joint (nodes): A place where two or more members of the truss are connected.
- Force equilibrium: A body is in force equilibrium if the sum of all the forces acting on the body is zero.
- Moment equilibrium: A body is in moment equilibrium if the sum of all the moments of all the forces acting on the body is zero.
- Moment: The turning effect of a force is called the moment. The moment is the result of the force multiplied by the perpendicular distance from the line of action of the force to the pivot or point where the object will turn.
- Compression: The internal force acts to shorten the member.
- Tension: The internal force acts to long the member.
- Pinned support: A pinned support can resist both vertical and horizontal forces but not a moment.
- Roller support: Roller support and reactions resist perpendicular forces and they cannot resist parallel or horizontal forces and moments. It means the roller support will move freely along the surface without resisting horizontal force.
- Zero force members: Members which do not have any force in them.

4.1.2 Physical System Description

The physical system of the Truss Tool, as shown in Figure 3, includes the following elements:

- PS1: The joints (j_1,j_2,..,j_n).
- PS2: The members (m_1,m_2,..,m_k).
- PS3: The supports (S_1,S_2).

4.1.3 Goal Statements

Given the truss features and external forces, the goal statements are:

- GS1: Calculate the reactions of the supports.
- GS2: Calculate the internal forces for each member.

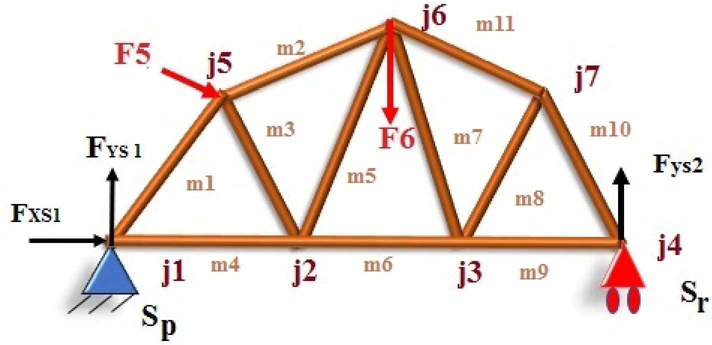

Figure 3: The physical system of Truss Tool

4.2 Solution Characteristics Specification

The instance models that govern Truss Tool are presented in Subsection 4.2.8. The information to understand the meaning of the instance models and their derivation is also presented so that the instance models can be verified.

4.2.1 Assumptions

This section simplifies the original problem and helps in developing the theoretical model by filling in the missing information for the physical system. The numbers given in the square brackets refer to the theoretical model [T], the general definition [GD], data definition [DD], instance model [IM], or likely change [LC], in which the respective assumption is used.

- A1: All members and nodes lie within a two-dimensional plane.
- A2: Members are inter-connected only at their ends.
- A3: Members must be straight.
- A4: All joints are frictionless hinges.
- A5: All forces must only be applied at joints
- A6: All reactions must only be applied at joints
- A7: Self-weight of the member will be neglected
- A8: Members are subjected to axial forces only.
- A9: The number of supports are at most two.

4.2.2 Theoretical Models

This section focuses on the general equations and laws that Truss Tool is based on.

Number	T1
Label	**General Equilibrium equations**
SI Units	All Forces are measured in N Moments are measured in N m
Equation	$\sum F_x = 0$, $\sum F_y = 0$, $\sum M = 0$
Description	The equilibrium equation describes the static equilibrium of all forces of the system and the moment for the system so that $\sum M = 0$ and $\sum F = 0$. F is any force in the system (N). M is a moment that is the turning effect of a force. Moments act about a point in a clockwise or anticlockwise direction(N m) According to A1 the summation of forces either horizontal $\sum F_x = 0$ or vertical $\sum F_y = 0$ should be equal to zero.
Source	[2]
Ref. By	IM2,IM3

4.2.3 General Definitions

Not applicable.

4.2.4 Data Definitions

This section collects and defines all the data needed to build the instance models. The dimension of each quantity is also given.

Number	DD1
Label	**Length of a straight Line**
Symbol	$L(m) : Member \rightarrow \mathbb{R}$
SI Units	m
Equation	$L = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$
Description	For every two points such as X_1,X_2 with coordination (x_1, y_1) and (x_2, y_2) the length of line between two point is L.
Sources	https://www.cuemath.com/distance-formula/
Ref. By	IM1

Number	DD2
Label	**Finding angle by Law of cosine**
Symbol	θ
SI Units	Degree
Equation	$\theta = \arccos(\frac{a^2+b^2+c^2}{2ab})$
Description	The Law of Cosine helps us to find any angle for a given triangle with a known length of sides. Where θ is the angle between sides a and b. and c is the length of the opposite side.
Sources	https://en.wikipedia.org/wiki/Law_of_cosines
Ref. By	IM1

Number	DD3
Label	**Decomposition of the force**
Symbol	$D(F, \theta) \rightarrow (F_x, F_y)$
SI Units	(N, N)
Equation	$F_x = F\cos\theta$, $F_y = F\sin\theta$
Description	Each force say F with angle θ with horizontal line can be decomposed to F_x and F_y
Sources	[1]
Ref. By	IM1

4.2.5 Data Types

This section collects and defines all the data types needed to document the models. For Truss Tool, all data types are real numbers or Boolean.

4.2.6 Instance Models

This section transforms the problem defined in Section 4.1 into one which is expressed in mathematical terms. It uses concrete symbols defined in Section 4.2.4 to replace the abstract symbols in the models identified in Sections 4.2.2.

The goals GS1 and GS2 are solved by IM1, IM2, IM3

4.2.7 Data Types

This section collects and defines all the data types needed to document the models. For Truss Tool, all data types are real or integer numbers.

4.2.8 Instance Models

For better understanding the inputs and outputs of the instant models consider a simple truss with $n = 3$ joints as shown in figure 4. The other trusses with higher number of joints can modeled similarly.

The input for IM1 is geometric location of each joint $J = \{(0,0),(1,1),(2,0)\}$. Also we have index for joint, for example $J_1 = (0,0)$, $J_2 = (1,1)$, $J_3 = (2,0)$. The other input is Members $M = \{(1,2),(1,3),(2,3)\}$. for example $M_1 = (1,2)$ which means that J_1 is

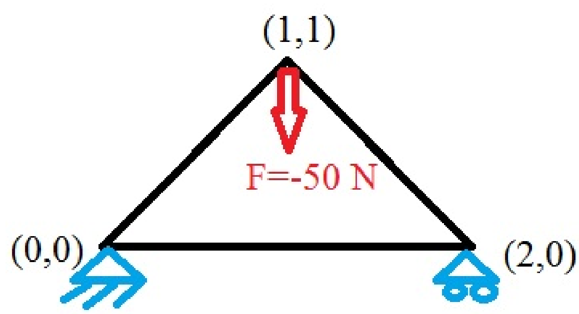

Figure 4: Example of simple truss for better understanding.

connected to J_2. Now by applying DD1 we can calculate $L_1 = \sqrt{(1-0)+(1-0)} = 1.41$. Same for $M_2 = (1,3)$, the distance between $j_1 = (0,0)$ and $J_3 = (2,0)$ is calculated as 2. The out put of the IM1 is $L = \{1.41, 2, 1.41\}$By DD2 and L we can calculate θ for each joint. $\theta_1 = \arccos(\frac{(1.41)^2-(1.41)^2-(2)^2}{-2*2*1.41} = 45$. The array of angles will be calculated as $\theta = \{45, 90, 45\}$. For IM2 other inputs are modeled as external forces on each joint $F = \{(0,0),(0,-50),(0,0)\}$ which means there is an external force only on J_2. Also $S_p = 1$ and $S_r = 3$ which means pinned support is on $J_1 = (0,0)$ and roller support is on $J_3 = (2,0)$. The output will be calculated as $R_{px} = 0, R_{py} = +25, R_{ry} = +25$.

Number	IM1
Label	**Calculate truss design features**
Input	J_i = Tuples of (X, Y) location of joints M_j = Tuples of end-joints for the members where each end-joint has a specific integer index.
Output	L_j = Tuples of the length of all members such that $0 \leq L_j \leq 20$ $\theta_{p,q}$ = Tuples of all angles between two members M_p and M_q so that $0 < \theta < 180$
Description	For each member, Truss Tool should calculate the length Lj from DD1. For each two members M_p and M_q, Truss Tool should calculate $\theta_{p,q}$ from DD2.
Sources	[1]
Ref. By	IM3

Number	IM2
Label	**Find support's reactions**
Input	S_p = Position of the pinned support as joint index, S_r Position of the roller support as joint index F_i = Tuples of a joint index for the position of an external force, and the amount of force in x direction and y direction (N)
Output	R_px, R_py for pinned support (S_p) and R_ry for roller support (S_r)
Description	By considering the whole truss as a free body, The reaction of the supports should be calculated from T1 The input is constrained so that we have at most only two supports at most S_p, S_r (A9) For inputting the position of an External force or support, the index of a join is needed. (A5,A6)
Sources	[1]
Ref. By	IM3,GS1

Number	IM3
Label	**Calculate internal forces for all members** IF_m
Input	F_i = External forces. External forces can be applied only at joints. We have index of each joint as integer number i. Amount of external force should be entered as F_x, F_y. Lj all Members lengths, $\theta_{p,q}$ all angles between members from IM1 R_px,R_py, R_ry reactions from IM2
Output	IF_j Internal force for each member by solving $(n-1)$ equation of equilibrium for each joint except the last joint. where j is an integer index for a member.(N)
Description	By decomposition of each internal force IF_i to IF_x and IF_y (A1) and applying equilibrium equations from T1 for each joint, the internal forces will be calculated jointly. For the last joint j_n, there will be no unknown internal force left. Hence we can use the last equation as the verification test of output correctness.
Sources	[1]
Ref. By	GS2

4.2.9 Input Data Constraints

Table 1 shows the data constraints on the input-output variables. The column for physical constraints gives the physical limitations on the range of values that the variable can take. The column for software constraints restricts the range of inputs to reasonable values. The software constraints will be helpful in the design stage for picking suitable algorithms. The constraints are conservative, allowing the model user to experiment with unusual situations. The column of typical values is intended to provide a feel for a common scenario. The uncertainty column estimates the confidence with which the physical quantities can be measured. This information would be part of the input if one were performing an uncertainty quantification exercise.
The specification parameters in Table 1 are listed in Table 2.

(*) The count of Joints in a given truss is an integer number. it must be greater or equal to 3 to be considered a triangle. For small trusses, the number of joints is around 8. The maximum number of joints n_{max} for the run time considerations will be considered 20.

Table 1: Input Variables

Var	Physical Constraints	Software Constraints	Typical Value	Uncertainty
n	$n > 3$	$n_{\min} \leq n \leq n_{\max}$	5	5%

Table 2: Specification Parameter Values

Var	Value
$n_{\min}$	3
$n_{\max}$	9

4.2.10 Properties of a Correct Solution

Table 3 shows the physical constraints on the output. Suppose all joints from index 1 to $n-1$ are solved. Then all IF_m are already calculated and the last joint will be considered as a physical constraint on the output.

Table 3: Output Variables

Var	Physical Constraints
$\sum F_n = 0$	(by A1,A5)

5 Requirements

This section provides the functional requirements and the business tasks that the software is expected to be complete, and the nonfunctional requirements, the qualities that the software is expected to exhibit.

5.1 Functional Requirements

- R1: Input the values from Table 4
- R2:Echoing inputs as part of output.
- R3: Calculate support reactions from IM2, and internal forces from IM1, IM2, IM3
- R4: Check summation of internal forces is zero in the last joint. IM3

Table 4: Required Inputs

Symbol	Description	Data Type

5.2 Nonfunctional Requirements

NFR1: **Reliability: The Time performance of the software should be able to tested through verification and validation plan (VnV Plan).**

NFR2: **Maintainability: The effort required to make any of the likely changes listed for Truss Tool should be less than 30% of the original development time.**

NFR3: **Portability: Truss Tool is runnable on Windows.**

6 Likely Changes

LC1: The software may be changed to solve both types of trusses: two-dimensional and three-dimensional [A1]

LC2: The software may be changed to consider friction of the joints [A4]

7 Unlikely Changes

UC1: The truss members are only connected at their joints [A2]

UC2: The truss members are straight [A3]

8 Traceability Matrices and Graphs

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Every time a component is changed, the items in the column of that component that is marked with an "X" may have to be modified as well. Table 5 shows the dependencies of theoretical models, general definitions, data definitions, and instance models with each other. Table 6 shows the dependencies of instance models, requirements, and data constraints on each other. Table 7 shows the dependencies of theoretical models, general definitions, data definitions, instance models, and likely changes in the assumptions.

9 Development Plan

Not applicable.

10 Values of Auxiliary Constants

	T1	DD1	DD2	DD3	IM1	IM2
T1						
DD1	X					
DD2						
DD3						
IM1			X	X		
IM2		X				
IM3		X			X	X

Table 5: Traceability Matrix Showing the Connections Between theoretical models, general definitions, data definitions and Instance Models

	IM1	IM2	IM3	Constraint 4.2.9
R1	X			X
R2	X			X
R3	X	X	X	
R4				X

Table 6: Traceability Matrix Showing the Connections Between Requirements and Instance Models

	A1	A2	A3	A4	A5	A6	A7	A8	A9
T1	X								
DD1			X						
DD2			X	X					
DD3	X	X	X						
IM1		X	X						
IM2	X	X	X	X	X	X			X
IM3	X			X			X	X	X

Table 7: Traceability Matrix Showing the Connections Between assumptions and other sections. For example A3 is essential to calculate distance byDD1

References

[1] Vera Galishnikova. *Geometrically Nonlinear Analysis of Plan trusses and Frames*. AFRICAN SUN MeDIA, 2009.

[2] Russell Charles Hibbeler and Kiang-Hwee Tan. *Structural analysis*. Pearson Prentice

Symbol	Description	Value	Units
F_{max}	the maximum value for external force	35000	N
F_{min}	the minimum value for external force	-35000	N
θ_{max}	the maximum value for angle	180	Degree
θ_{min}	the minimum value for angle	0	Degree
d_{max}	the maximum value for the count of joints	9	
d_{min}	the minimum value for the count of joints	3	
S_{max}	the maximum value for the count of supports	2	
S_{min}	the minimum value for the count of supports	1	

Table 8: Auxiliary Constants

Hall Upper Saddle River, 2006.

[3] W Spencer Smith and Lei Lai. A new requirements template for scientific computing. In *Proceedings of the First International Workshop on Situational Requirements Engineering Processes–Methods, Techniques and Tools to Support Situation-Specific Requirements Engineering Processes, SREP*, volume 5, pages 107–121. Citeseer, 2005.