ISSN 2176-5480

22nd International Congress of Mechanical Engineering (COBEM 2013)
November 3-7, 2013, Ribeirão Preto, SP, Brazil

DEVELOPMENT OF A COMPUTATIONAL TOOL FOR STATIC ANALYSIS WITH SCREW THEORY

Marcos Goulart Reis
Henrique Simas
Daniel Martins
Raul Guenther Laboraty of Robotics, Federal University of Santa Catarina, Campus Universitário, Trindade, Florianópolis/SC, Brasil.
academico@marcosgreis.com, hsimas@gmail.com, danielemc@gmail.com

Abstract. *This paper treats the development of a software module for static analysis for any mechanism. The module was developed as an addition to a framework for kinematic analysis created by Raul Guenther Laboratory of Robotics of Federal University of Santa Catarina. The expanded framework can determine the static analysis of a moving mechanism, being possible to include external forces such as gravitational forces of each link and environmental contact. The created module realizes the analysis based on screw theory, graph theory and Assur's virtual chains to determine the static model of a mechanism. In order to illustrate the use of the proposed module, a didactic example are developed to validate it and a static model of a spatial parallel robot with three degrees of freedom is solved.*

Keywords: *Screw Theory; Static Analysis; Graph Theory; Parallel Robot*

1. INTRODUCTION

The research around industrial robotics is focused on the following areas: kinematic, static, dynamic and control Sciavicco and Siciliano (2004). Some research has been developed in these areas. Among these, the statics for robots problem was addressed by many thesis (Cazangi, 2008; Erthal, 2010; Cruz, 2010; Weihmann, 2011; Rincon, 2012).

Recent work at Laboratory of Robotics have resulted a computational framework for modelling robotic system based on screw theory. This framework, named as KAST (Kinematic Analysis by Screw Theory), is also used for kinematic analysis of this systems (Rocha, 2012).

The use of computational frameworks for simulation is growing. There are many computational tools for robotic simulations that are free software, such as OpenRAVE (Open Robotics Automation Virtual Environment), Gazebo, v-rep (Virtual Robot Experimentation Platform) and KAST (Diankov, 2010, 2013; Open Source Robotics Foundation, 2013; Coppelia Robotics, 2013; Rocha, 2012).

The OpenRAVE provides and environment for testing, developing and deploying motion planning algorithms in real-world robotics applications. The main focus is on simulation and analysis of kinematic and geometric information (Diankov, 2010, 2013). The Gazebo is a multi-robot simulator for outdoor environments. It is capable of simulating a population of robots, sensors and objects, but does so in a three-dimensional world. It also has a graphical user interface to build and edit models and simulations (Open Source Robotics Foundation, 2013). The *v-rep* provides the features of the other 2 mentioned software, OpenRAVE and Gazebo, and further. It includes particle dynamics, an interface for local and remote programming and collision detection (Coppelia Robotics, 2013). The KAST is a software in its initial state of development, it was made to be modular and extensible. Until now, the KAST provides a module for modeling robotic systems and kinematic analysis of robots through screw theory, a module for path generation e an additional framework for motion planning in a context (Rocha, 2012).

None of the mentioned software allows to analyse the statics for mechanisms and robots.

This paper reports the development of a software module for statics analysis for any mechanism. This module is developed to expand the KAST. The module can provide functions to solve the statics for any mechanism or robot, the results given include the whole joint robot data in terms of positions, moments and forces.

This paper first presents the tools used to develop the module, that is, the screw theory and the statics modelling. The module is also validated with a didactic example and it is explored the capability of simulate spatial robots.

2. STATICS FOR ROBOTS THROUGH SCREW THEORY

The screw represents the state of movements or actions, that is, the kinematic and statics for rigid bodies in space. This theory is based on two theorems, the Poinsot theorem and the Mozzi-Chasles theorem (Ball, 1900).

The screws may be expressed using Plücker coordinates Davidson and Hunt (2004). The screw can be defined as shown in Eq. (1), where $\vec{s}$ is the screw axis direction unit vector, $\vec{S}_0$ is the position vector of a point that belongs to the screw, relative to the origin, and h is called step. The homogeneous Plücker coordinates are L, M, N, P^*, Q^* and R^*

(Davidson and Hunt, 2004).

$$\$ = \left(\begin{array}{c} \vec{S}_0 \times \vec{s} + h\vec{s} \\ \cdots\cdots\cdots\cdots \\ \vec{s} \end{array} \right) = \left(\begin{array}{c} P^* \\ Q^* \\ R^* \\ \cdots\cdots \\ L \\ M \\ N \end{array} \right) \quad (1)$$

For the statics analysis, the screw in ray order are used and called *wrenches*, the Eq. (2) shows its representation. The $\vec{M}_0$ represents the torque, $\vec{F}$ represents the forces and the superscript A of $\$^A$ shows that is a action screw, also called *wrench* (Davidson and Hunt, 2004).

$$\$^A = \left(\begin{array}{c} \vec{S}_0 \times \vec{s} + h\vec{s} \\ \cdots\cdots\cdots\cdots \\ \vec{s} \end{array} \right) = \left(\begin{array}{c} P^* \\ Q^* \\ R^* \\ \cdots\cdots \\ L \\ M \\ N \end{array} \right) = \left(\begin{array}{c} \vec{M}_0 \\ \cdots\cdots \\ \vec{F} \end{array} \right) \quad (2)$$

To analyse the statics, is needed to define the *action matrix* $[A_D]_{\lambda \times C}$. In this matrix, the amount of rows is given by the system order (λ) and the columns represents the unit action through each joint's wrench, it is shown in Eq. (3) (Cazangi, 2008).

$$[A_D]_{\lambda \times C} = \left[\begin{array}{cccc} \$_1^A & \$_2^A & \cdots & \$_F^A \end{array} \right] \quad (3)$$

Given the norm of each wrench of action matrix, we obtain the *unit action matrix* $[\hat{A}_D]_{\lambda \times C}$, represented in Eq. (4). The magnitudes of each wrench results the *magnitude action vector* $\{\vec{\Psi}_{C \times 1}\}$ shown in Eq. (5) (Cazangi, 2008).

$$[\hat{A}_D]_{\lambda \times C} = \left[\begin{array}{cccc} \hat{\$}_1^A & \hat{\$}_2^A & \cdots & \hat{\$}_F^A \end{array} \right] \quad (4)$$

$$[\vec{\Psi}_{C \times 1}] = \left[\begin{array}{cccc} \psi_1 & \psi_2 & \vdots & \psi_F \end{array} \right]^T \quad (5)$$

More information can be found in (Martins, 2002; Campos, 2004; Cazangi, 2008; Simas, 2008; Erthal, 2010; Cruz, 2010; Weihmann, 2011; Rincon, 2012; Rocha, 2012)

2.1 Kirchhoff's Laws

The Kirchhoff's Laws for electric circuits were adapted by Davies to be used on mechanical systems (Davies, 1981; Martins, 2002; Campos, 2004; Cazangi, 2008).

Adapting the Kirchhoff's current law, it was possible to establish the relationships between actions belonging to the same partition, that contributed to the statics analysis. This law states that the algebraic sum of currents in a network of conductors meeting at a point is zero. Analogously, Davies (2006) states that the algebraic sum of wrenches belonging to the same partition is zero, which is the *Cut Law* (Davies, 1981).

This implies that to any coupling networks in balance, any subset of couplings separated by partitions, the sum of each element of this action is null. Equation (6) shows the sum considering a cut into space and using screws. In the Eq. (7) is applied the matrix notation according to the Eqs. (3), (4) and (5).

$$\sum P^* = \sum Q^* = \sum R^* = \sum L = \sum M = \sum N = 0 \quad (6)$$

$$\sum \$^A = [A_D]_{\lambda \times C} = [\hat{A}_D]_{\lambda \times C} \{\vec{\Psi}\}_{C \times 1} = [\vec{0}]\lambda \times 1 \quad (7)$$

2.2 Statics Analysis

Cazangi (2008) presents the statics analysis for mechanisms and robots separated into 9 steps. This simplifies its development in a computational tool. These steps are briefly shown below.

1. Mechanism characterization

ISSN 2176-5480

   (a) Schematic representation with coordinated system O_{xyz}.
   (b) Coupling network representation.
   (c) Coupling graph representation.: G_C.
   (d) Determine the incidence matrix from graph G_C: $[I_C]_{n\times e}$.
   (e) Calculate the echelon form of incidence matrix $[I_C]_{n\times e}$ to obtain the cut-f matrix of G_C: $[Q_C]_{k\times e}$.

2. Coupling characterization
   (a) Geometric characterization ($\vec{s}^A, \vec{S}_0^A, h^A$) and action ($c_p, c_a$).
3. Topologic (Cuts)
   (a) Action graph representation: G_A.
   (b) Parallel expansion of the C unit constraint of cut-f matrix: $[Q_A]_{k\times C}$.
4. Geometric (Wrench)
   (a) Defining wrenches: $\$^A$
   (b) Action matrix determination: $[A_D]_{\lambda\times C}$
5. Equation system
   (a) Unit actions network matrix determination: $[\hat{A}_D]_{\lambda.k\times C}$.
   (b) Kirchhoff's Law: $[\hat{A}_D]_{\lambda.k\times C}\{\vec{\Psi}\}_{C\times 1} = [\vec{0}]\lambda.k \times 1$.
6. Under constraint (extra freedom)
   (a) Determination of the F_N dependent equations.
   (b) Eliminating the dependent equations.
7. Variable separation
   (a) Selecting the C_N primary variables on vector $\{\vec{Psi}\}_{C\times 1}$.
   (b) Separating the primary and secondary variables on the equation system.
8. Solution
   (a) Calculate the inverse matrix of $[A_{NS}]_{a\times a}$, give values to $\{\vec{\Psi}_P\}_{C_N\times 1}$ and obtain the solution $\{\vec{\Psi}_S\}_{a\times 1}$
9. Actions instant state
   (a) Apply the magnitudes $\{\vec{\Psi}\}$ to the wrench $\$^A$ of each coupling.

Examples and more information can be found in (Cazangi, 2008; Erthal, 2010; Cruz, 2010; Weihmann, 2011; Rincon, 2012).

3. FRAMEWORK FOR KINEMATIC AND STATICS ANALYSIS

The framework KAST, developed by Rocha (2012), is a computational tool for kinematic analysis using the Davies method. With this framework is possible to analyse different types of kinematic chains. The input kinematic models are exclusively based on screw theory. The development was motivated by the need to ease the creation of cooperative/vehicle-manipulator systems simulations.

The KAST was developed to be modular and extensible, therefore it was decided to create a module for statics for mechanisms and robots in addition to KAST. With the expanded framework, named KSAST (Kinematic and Static Analysis by Screw Theory), is possible to analyse the statics for any mechanism or robot through screw theory. The module allows to include external forces such as gravitational forces of each link and environmental contact.

The framework provides two ways to define kinematic chains, using a description file, XML (W3C, 2013) based, or directly through Python code. The first is generic and contribute to create a kinematic chains models repository. The second allows to change the kinematic chain during the simulation, enabling to simulate reconfigurable chains.

Figure 1 shows the UML diagram for the whole framework.

The diagram shown in Fig. 1 is divided in 3 groups: *Transformation*, *Loader* and *Components*. The *Transformation* group are classes that are responsible for change the coordinate system of specialized classes from **BaseScrew**. The *Loader* group are classes that are responsible for loading the XML based file into the memory of the system, converting the file attributes into classes of the *Components* group. The *Components* group brings together classes that represent a component of the system.

ISSN 2176-5480

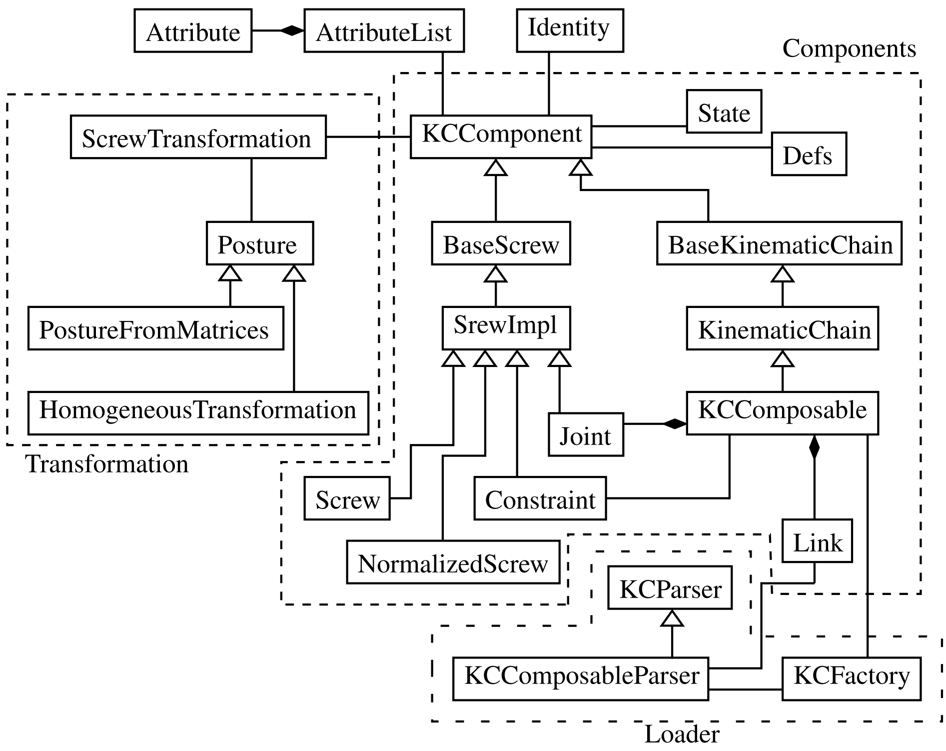

Figure 1. UML diagram

3.1 Using the framework

To use the framework, it is needed to provide information about the robot and its couplings. The better way to provide such data is using a XML based file. To ease the comprehension, the explanation about how to use the framework will be based on the steps shown at Subsection 2.2and will be applied to the structure shown in Fig. 2. This example is from the book Hibbeler (1999), page 261, solved problem 4.8.

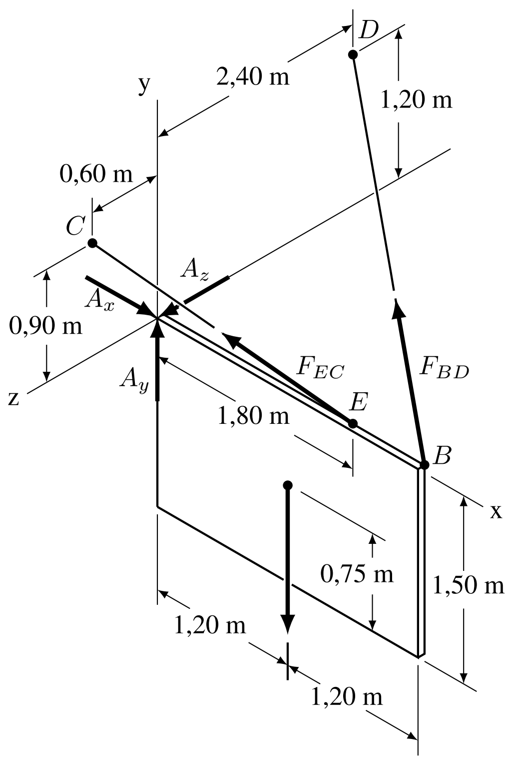

Figure 2. Schematic representation (Hibbeler, 1999)

ISSN 2176-5480

The example shows a sign fixed on the wall by a ball joint, cable BD and cable EC.

Some steps does not need input data from the user, these steps are performed automatically by the KSAST. The steps (1.a), (1.b), (1.c), (4.a) and (7.a) are the steps which need input. The step (2.a) is not used directly to create the XML file, but it is used by the step (4.a). They are explained and exemplified along this subsection.

Step (1.a) Within this step, the user must gather the information about the state variables needed to put them into the XML file. For the example, we have 6 variables for statics, being A_x, A_y, A_z, F_{BD}, F_{EC} and P. The XML file snippet is shown in List. 1.

```
<states>
  <state type="static">
    <var name="Ax">0.0</var>
    <var name="Ay">0.0</var>
    <var name="Az">0.0</var>
    <var name="Fbd">0.0</var>
    <var name="Fec">0.0</var>
    <var name="P">-1350.0</var>
  </state>
</states>
```

Listing 1: Structure's state variables

Step (1.b) With the coupling network shown in Fig. 3 is possible to determine the structure's links. In this case, we have 2 links, the wall and the sign. The XML snippet is shown is List. 2.

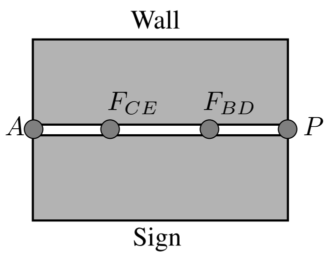

Figure 3. Structure's coupling network

```
<links>
  <Link id="0" name="wall" base="yes" />
  <Link id="1" name="sign" />
</links>
```

Listing 2: Structure's links

Step (1.c) The coupling graph, in Fig. 4 shows the relations between the links of the mechanisms. The information obtained in this step is used to fill the fields **linkfrom** and **linkto** of each constraint. The XML snippet for the whole constraint definition is shown in List. 3.

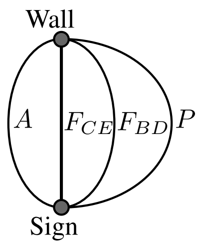

Figure 4. Structure's coupling graph

Step (2.a) In this step we gather coupling geometric characteristics for each constraint. The vector $\vec{s}$ of each constraint is

ISSN 2176-5480

shown in Eq.(8) and the position vector $\vec{S}_0$ is shown in Eq. (9).

$$\vec{s}_{a_x} = \begin{Bmatrix} 1 \\ 0 \\ 0 \end{Bmatrix}, \quad \vec{s}_{a_y} = \begin{Bmatrix} 0 \\ 1 \\ 0 \end{Bmatrix}, \quad \vec{s}_{a_z} = \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix},$$
$$\vec{s}_b = \begin{Bmatrix} -0,6666 \\ 0,3333 \\ -0,6666 \end{Bmatrix}, \quad \vec{s}_e = \begin{Bmatrix} -0,8571 \\ 0,4285 \\ 0,2857 \end{Bmatrix}, \quad \vec{s}_P = \begin{Bmatrix} 0 \\ -1 \\ 0 \end{Bmatrix} \tag{8}$$

$$\vec{S}_{0a} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix}, \quad \vec{S}_{0b} = \begin{Bmatrix} -2,4 \\ 1,2 \\ 2,4 \end{Bmatrix}, \quad \vec{S}_{0e} = \begin{Bmatrix} -1,8 \\ 0,9 \\ 0,6 \end{Bmatrix}, \quad \vec{S}_{0P} = \begin{Bmatrix} 1,2 \\ 0,75 \\ 0,0 \end{Bmatrix} \tag{9}$$

Step (4.a) With the couplings geometric characteristics shown in Step (2.a), the wrench is defined as displayed in the Eq. (2). In this step, we will consider the wrenches $\$^A_{F_{EC}}$ and $\$^A_P$. The result for the example wrenches used in this step are shown in the Eq. (10).

$$\$^A_{F_{EC}} = \begin{pmatrix} 0 \\ -0,5143 \\ 0,7714 \\ \cdots\cdots\cdots \\ -0,8571 \\ 0,4286 \\ 0,2857 \end{pmatrix}, \quad \$^A_P = \begin{pmatrix} 0 \\ 0 \\ -1,2 \\ \cdots\cdots\cdots \\ 0 \\ -1,0 \\ 0 \end{pmatrix} \tag{10}$$

The wrench components, in the XML file, are represented by the **component** field of each **Constraint**. The List. 3 shows a full example of defining the wrenches $\$^A_{F_{EC}}$ and $\$^A_P$.

```xml
<constraints>
  <Constraint id="104" name="el104"
                 type="translational" var="Fec">
    <components>0.0 -0.5143 0.7714
                 -0.8571 0.4286 0.2857</components>
    <linkfrom>parede</linkfrom>
    <linkto>cartaz</linkto>
  </Constraint>
  <Constraint id="105" name="el105"
                 type="translational" var="P">
    <components>0.0 0.0 -1.2 0.0 -1.0 0.0</components>
    <linkfrom>parede</linkfrom>
    <linkto>cartaz</linkto>
  </Constraint>
</constraints>
```

Listing 3: Constaint example

Step (7.a) To select the primary variables of vector $\{\vec{\Psi}\}_{C\times 1}$ is shown in the List. 4. The vector inside **partitioning** has size equal to the amount of variables defined on Step (1.a). Each digit indicates if it is a primary variable (with value 1) or a secondary variable (with value 0). In this example, only the force P is primary, therefore, only the last digit is 1.

```xml
<partitioning type="static">0 0 0 0 0 1</partitioning>
```

Listing 4: Partitioning example

Step (8.a) The solution values are shown in Tab. 1. This table also show a comparison between the KSAST results and the results presented in Hibbeler (1999). The difference in the values are because the framework does not round the obtained values nor considers only significant digits.

The comparison in Tab. 1 contributes to validate the framework for spatial statics analysis.

ISSN 2176-5480

Table 1. Statics result comparison

	KSAST	(Hibbeler, 1999)
A_x	1687,5 N	1690 N
A_y	506,25 N	504 N
A_z	-112,5 N	-114 N
F_{BD}	506,25 N	506 N
F_{EC}	1575 N	1580 N

4. STATICS MODEL THROUGH SCREW THEORY FOR THE SPATIAL DELTA ROBOT

In this section, the statics model for a Delta robot is developed using screw theory. In the Section 5 this model is applied in the framework KSAST and results are presented. Figure 5 shows the Delta robot modelled in this paper. This robot is a spacial parallel robot with 3 degrees of freedom.

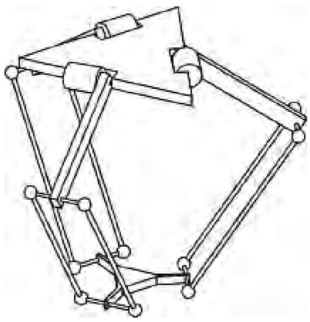

Figure 5. The Delta robot (Liu *et al.*, 2005)

The 3 legs of this robot are symmetrical, therefore, we can model the leg according the angle φ relative to the x axis. The Tab. 2 shows the values of φ for each leg i.

Table 2. Angle φ for each leg

i	φ
0	0°
1	120°
2	240°

Step (1.a) Figure 6 shows the schematic representation of leg i at the initial position, the origin O_{xyz}. In this figure is also possible to observe the screw axis relative to each joint's motion.

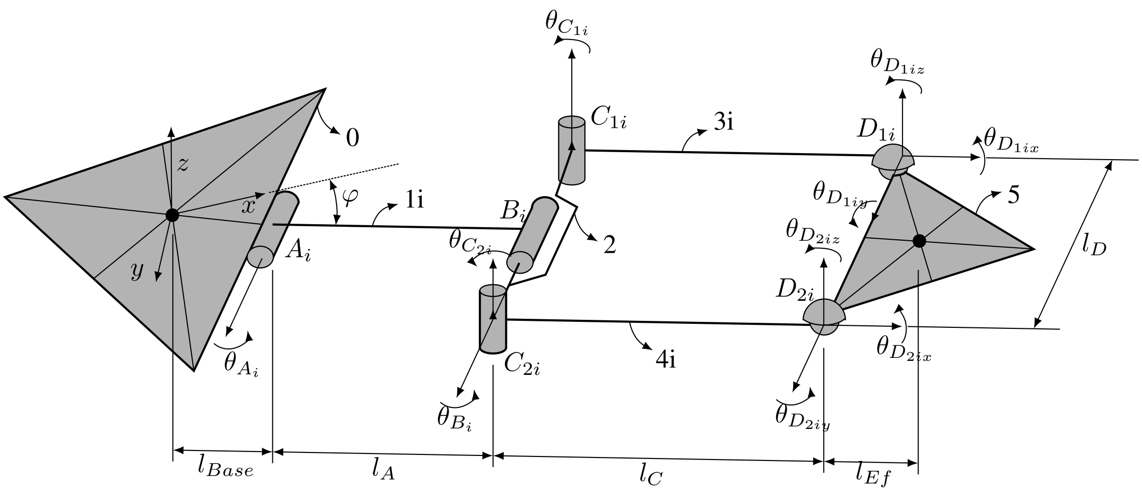

Figure 6. Schematic representation of Delta's leg i at the assumed reference position

ISSN 2176-5480

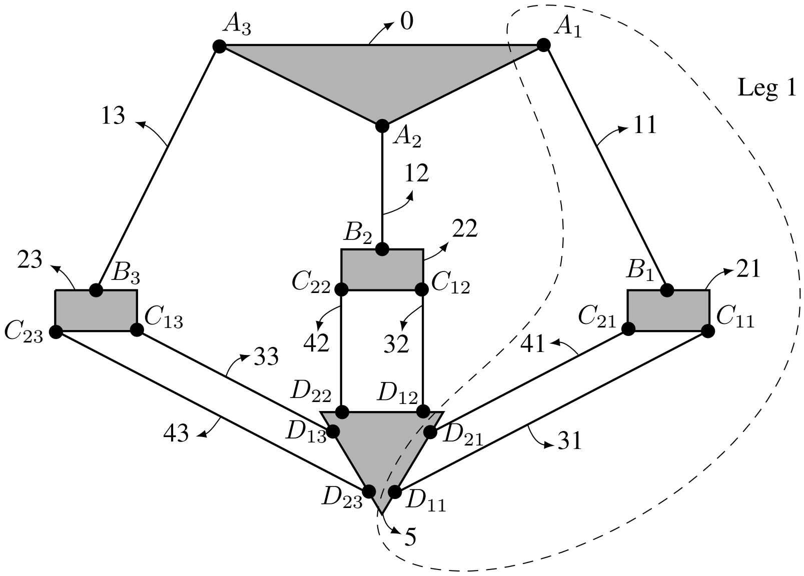

Figure 7. Delta's coupling network

Step (1.b) In Fig. 7 is shown the robot coupling network, with a single leg highlighted.

Step (1.c) The robot coupling graph is shown at Fig. 8.

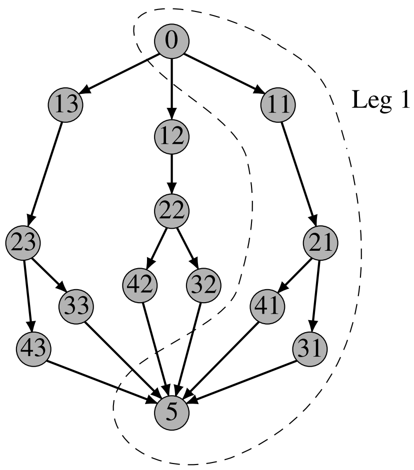

Figure 8. Delta's coupling graph

Step (2.a) In this step the coupling geometric characteristics are obtained. The i-th leg has $e = 6$ couplings, being that the A_i, B_i, C_{1i} and C_{2i} are rotative pairs, that is, they have $c_p = 5$ constraints. The couplings D_{1i} and D_{2i} are spherical pairs and have $c_p = 3$ constraints. The direction of constraints A_i e B_i are along x-axis, y-axis and z-axis, and also around x-axis and z-axis, while the couplings C_{1i} and C_{2i} have constraints along x-axis, y-axis and z-axis, and also around x-axis and y-axis. The spherical pairs D_{1i} and D_{2i} have the only the force constraints F_x, F_y e F_z. Besides these, is needed to consider $e_a = 3$ more active couplings, being one for each leg in coupling A_i, around the y-axis.

The Eq. (11), Eq. (12), Eq. (13) and Eq. (14) presents the vector $\vec{s}$ for A_i, B_i, C_{ji} and D_{ji} constraint, respectively.

$$\vec{s}_{A_{iF_x}} = \vec{s}_{A_{iT_x}} = \begin{Bmatrix} \cos(\varphi) \\ \sin(\varphi) \\ 0 \end{Bmatrix} \quad , \quad \vec{s}_{A_{iF_y}} = \vec{s}_{A_{iT_y}} = \begin{Bmatrix} \cos(\varphi+\pi/2) \\ \sin(\varphi+\pi/2) \\ 0 \end{Bmatrix} \quad , \quad \vec{s}_{A_{iF_z}} = \vec{s}_{A_{iT_z}} = \begin{Bmatrix} 0 \\ 0 \\ 1 \end{Bmatrix} \tag{11}$$

ISSN 2176-5480

$$\vec{s}_{B_iF_x} = \vec{s}_{B_iT_x} = \left\{ \begin{array}{c} \cos(\varphi) \\ \sin(\varphi) \\ 0 \end{array} \right\} \quad , \quad \vec{s}_{B_iF_y} = \left\{ \begin{array}{c} \cos(\varphi + \pi/2) \\ \sin(\varphi + \pi/2) \\ 0 \end{array} \right\} \quad , \quad \vec{s}_{B_iF_z} = \vec{s}_{B_iT_z} = \left\{ \begin{array}{c} 0 \\ 0 \\ 1 \end{array} \right\} \tag{12}$$

$$\begin{array}{l} \vec{s}_{C_{ji}F_x} = \left\{ \begin{array}{c} 1 \\ 0 \\ 0 \end{array} \right\} \quad , \quad \vec{s}_{C_{ji}T_x} = \left\{ \begin{array}{c} \cos(\theta_1 + \theta_2) * \cos(\varphi) \\ \cos(\theta_1 + \theta_2) * \sin(\varphi) \\ \sin(\theta_1 + \theta_2) \end{array} \right\} \quad , \quad \vec{s}_{C_{ji}F_y} = \left\{ \begin{array}{c} 0 \\ 1 \\ 0 \end{array} \right\} \\ \vec{s}_{C_{ji}T_y} = \left\{ \begin{array}{c} \cos(\varphi + \pi/2) \\ \sin(\varphi + \pi/2) \\ 0 \end{array} \right\} \quad , \quad \vec{s}_{C_{ji}F_z} = \left\{ \begin{array}{c} 0 \\ 0 \\ 1 \end{array} \right\} \end{array} \tag{13}$$

$$\vec{s}_{D_{ji}F_x} = \left\{ \begin{array}{c} 1 \\ 0 \\ 0 \end{array} \right\} \quad , \quad \vec{s}_{D_{ji}F_y} = \left\{ \begin{array}{c} 0 \\ 1 \\ 0 \end{array} \right\} \quad , \quad \vec{s}_{D_{ji}F_z} = \left\{ \begin{array}{c} 0 \\ 0 \\ 1 \end{array} \right\} \tag{14}$$

The position vector $\vec{S}_0$ for each constraint are defined in Eq. (15).

$$\begin{array}{l} \vec{S}_{0_{A_i}} = \left\{ \begin{array}{c} 0 \\ 0 \\ 0 \end{array} \right\} \quad , \quad \vec{S}_{0_{B_i}} = \left\{ \begin{array}{c} b_x \\ b_y \\ b_z \end{array} \right\} \quad , \quad \vec{S}_{0_{C_{1i}}} = \left\{ \begin{array}{c} c_{1x} \\ c_{1y} \\ c_{1z} \end{array} \right\} \quad , \quad \vec{S}_{0_{C_{2i}}} = \left\{ \begin{array}{c} c_{2x} \\ c_{2y} \\ c_{2z} \end{array} \right\} \\ \vec{S}_{0_{D_{1i}}} = \left\{ \begin{array}{c} d_{1x} \\ d_{1y} \\ d_{1z} \end{array} \right\} \quad , \quad \vec{S}_{0_{D_{2i}}} = \left\{ \begin{array}{c} d_{2x} \\ d_{2y} \\ d_{2z} \end{array} \right\} \end{array} \tag{15}$$

Step (4.a) With the geometric characteristic of the coupling, the wrenches are defined according the Eq. (2).

Step (7.a) The primary variables for vector $\{\vec{\Psi}\}_{C\times 1}$ for this model are the constraints related to the linear constraints F_x, F_y and F_z of the virtual chain, its through them that the actions are imposed at the end effector.

The robot modelled in this section is applied in the framework in the Section 5. Some graphical results are shown and analysed.

5. APPLYING THE STATICS MODEL IN THE FRAMEWORK

To show the use of the framework with a complex mechanism, the statics model of the Delta was applied on the KSAST. For this application, the end effector path was a circular path on xy-plane with a radius R, centered at the point $P = (0, 0, -0.9\text{m})$. A constant force $F = (F_x, F_y, F_z)$ was applied at the center of end effector. The model was used with different values for the radius R and the force F.

In every figure in this section, the x axis represents the x position of the end effector, the y axis represents the y position and the z axis represents the calculated torque of the first joint, named A in Section 4. All the units are expressed in the S.I. (metric) system.

Figure 9 shows the torque of the joint A of each leg with a $R = 0.2\text{m}$ and the force $F = (0, 0, -10\text{N})$. In this figure is possible to observe that the torques applied in each leg are symmetrical. In the Section 4.was shown the symmetry of the analysed robot, therefore this result was expected for a symmetrical force.

ISSN 2176-5480

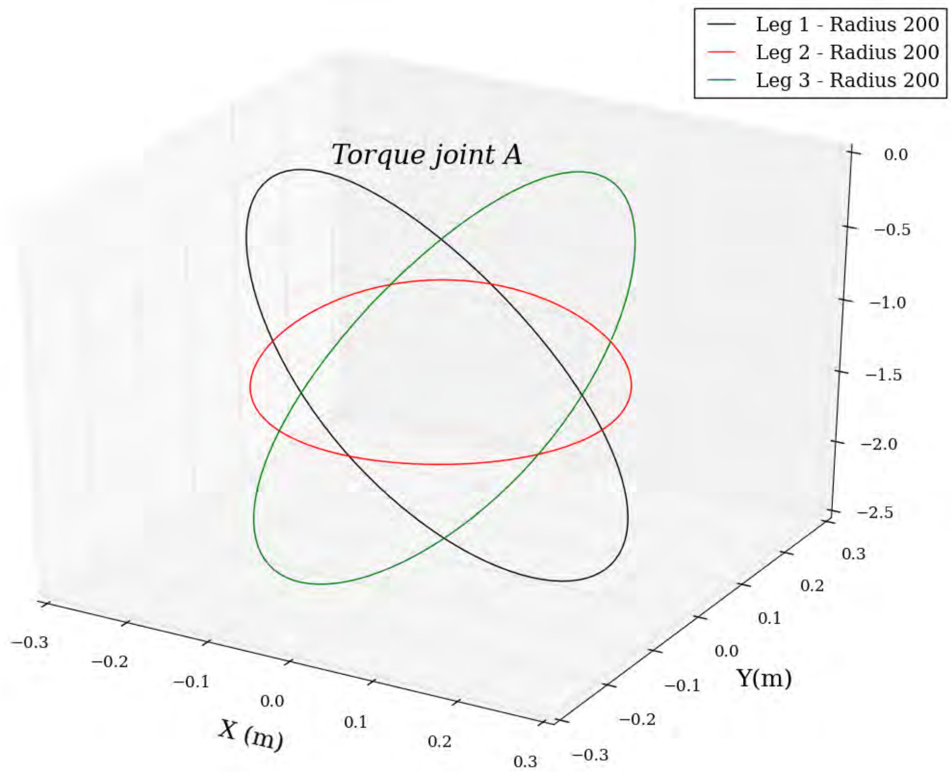

Figure 9. Positions and torques with radius $R = 0.2$m and $F = (0, 0, -10\text{N})$

Figure 10 shows only the joint A of the first leg, but for different values for the radius R.

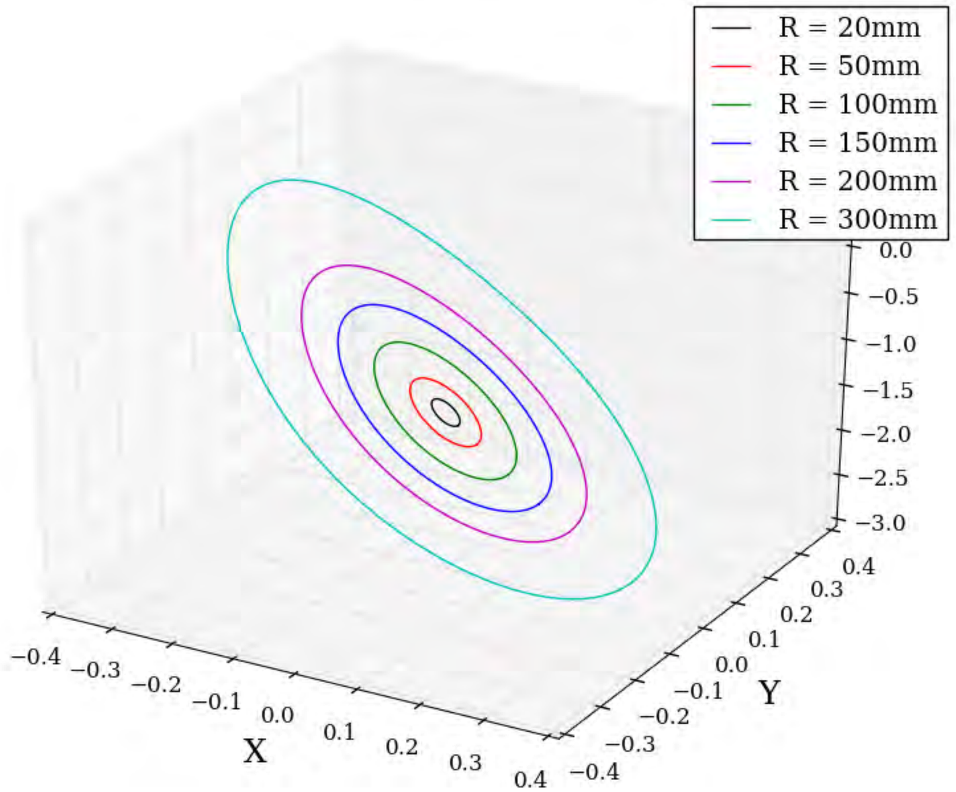

Figure 10. Positions and torques with different values for the radius R and $F = (0, 0, -10\text{N})$

Figure 11 shows the torque of the joint A of each leg with a $R = 0.2$m and the force $F = (1\text{N}, 0, -10\text{N})$. This figure shows that the leg 2 and leg 3 have symmetrical torques, while the leg 1 shows a different torque. That is because the F_x is aligned with the x axis, this alignment let the leg 2 and leg 3 with an angle of 120^o and -120^o with the applied force.

ISSN 2176-5480

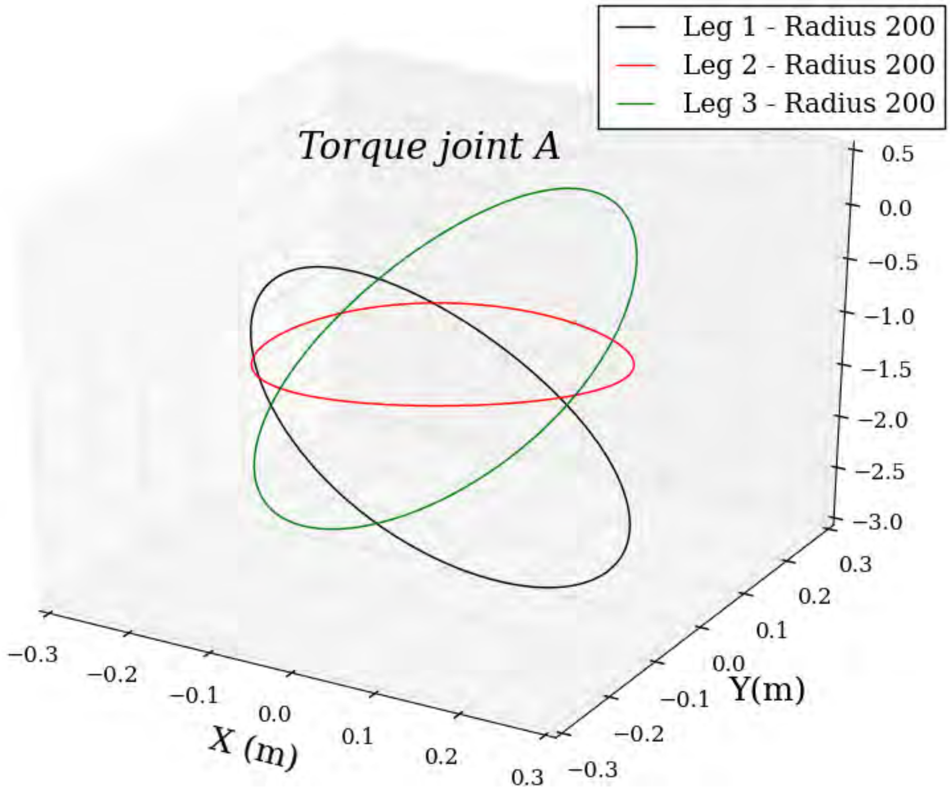

Figure 11. Positions and torques with radius $R = 0.2$m and $F = (1\text{N}, 0, -10\text{N})$

6. CONCLUSIONS

This paper presented another feature of a computational tool in its initial stage of development. With this new module it is possible to analyse the statics for any mechanism, which none of the tools surveyed have this feature.

In Subsection 3.1, a didactic example was solved to validate the developed module. To show that it is possible to model complex robots with ease, the Delta robot model was applied in the framework. The results obtained with the simulation in the framework includes the whole joint robot data in terms of positions, moments and forces.

The module contributes in projects and design of robots.

7. ACKNOWLEDGEMENTS

This optional section must be placed before the list of references.

8. REFERENCES

Ball, R.S., 1900. *A Treatise on the Theory of Screws*. Cambridge University Press, Cambridge.

Campos, A.A., 2004. *Cinemática diferencial de manipuladores empregando cadeias virtuais*. Tese de doutorado, Universidade Federal de Santa Catarina, Florianópolis.

Cazangi, H.R., 2008. *Aplicacão do Método de Davies para Análise Cinemática e Estática de Mecanismos de Múltiplos Graus de Liberdade*. Mestrado em engenharia mecânica, Universidade Federal de Santa Catarina.

Coppelia Robotics, 2013. "Coppelia robotics v-rep: Create. compose. simulate. any robot". URL `http://goo.gl/5sp8p`.

Cruz, F.B.C., 2010. *Modelagem, controle e emprego de robôs em processos de usinagem*. Tese de doutorado, Universidade Federal de Santa Catarina, Florianpólis.

Davidson, J. and Hunt, K., 2004. *Robots and SCREW Theory: Applications of Kinematics and Statics to Robotics*. Oxford University Press, Incorporated.

Davies, T., 2006. "Freedom and constraint in coupling networks". *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, Vol. 220, pp. 989–1010.

Davies, T.H., 1981. "Kirchhoff's circulation law applied to multi-loop kinematic chains". *Mechanism and Machine Theory*, Vol. 16, pp. 171–183.

Diankov, R., 2010. *Automated Construction of Robotics Manipulation Programs*. Phd thesis, Carnegie Mellon University, Pittsburgh.

ISSN 2176-5480

Diankov, R., 2013. "Openrave | home". URL http://openrave.org/.
Erthal, J.L., 2010. *Modelo cinestático para análise de rolagem em veículos*. Tese de doutorado, Universidade Federal de Santa Catarina, Florianpólis.
Hibbeler, R., 1999. *Estática: mecânica para engenharia*. Pearson Education do Brasil, São Paulo.
Liu, X.J., Wang, J. and Pritschow, G., 2005. "A new family of spatial 3-dof fully-parallel manipulators with high rotational capability". *Mechanism and Machine Theory*, Vol. 40, No. 4, pp. 475 – 494. ISSN 0094-114X.
Martins, D., 2002. *Análise cinemática hierárquica de robôs manipuladores*. Doutorado, Universidade Federal de Santa Catarina, Florianópolis.
Open Source Robotics Foundation, 2013. "Overview - gazebo". URL http://goo.gl/qIDjl.
Rincon, L.M., 2012. *Otimização da capacidade de carga de um manipulador paralelo 3RRR simétrico em trajetórias com contato*. Mestrado em engenharia mecânica, Universidade Federal de Santa Catarina.
Rocha, C.R., 2012. *Planejamento de movimento de sistemas robóticos de intervenção subaquática baseado na teoria dos helicoides*. Doutorado, Universidade Federal de Santa Catarina, Florianópolis.
Sciavicco, L. and Siciliano, B., 2004. *Modelling and Control of Robot Manipulators*. Advanced Textbooks in Control and Signal Processing. Springer, London.
Simas, H., 2008. *Planejamento de trajetórias e evitamento de colisão em tarefas de manipuladores redundantes operando em ambiantes confinados*. Doutorado, Universidade Federal de Santa Catarina, Florianópolis.
W3C, 2013. "Extensible markup language(xml)". URL http://www.scipy.org.
Weihmann, L., 2011. *Modelagem e otimização de forças e torques aplicados por robôs com redundância cinemática e de atuação em contato com o meio*. Doutorado, Universidade Federal de Santa Catarina, Florianópolis.

9. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.