Multibody Kinematics. A Topological Formulation Based on Structural-Group Coordinates

Mariano Saura*, Pablo Segado*, Bernardino Muñoz*, Daniel Dopico†, Javier Cuadrado†

* Department of Mechanical Engineering
Universidad Politecnica de Cartagena
Doctor Fleming, s/n, 30202 Cartagena, Spain
saura.sanchez@upct.es; psc1@alu.upct.es

Mechanical Engineering Laboratory
University of la Coruña
Mendizabal, s/n, 15403 Ferrol, Spain
ddopico@udc.es; javicuad@cdf.udc.es

ABSTRACT

Traditional kinematic formulations that are often used to model and analyse multibody systems have advantages, ones with respect to others, in terms of computational efficiency, which make them more suitable for real-time applications and in terms of versatility, which facilitates the modelling and solving multibody systems. A topological formulation which allows an efficient solution and facilitates the automatic modelling of multibody systems would fill the gap that currently exists between these traditional formulations, taking advantage of their benefits and mitigating their drawbacks. In this paper, a topological formulation based on the decomposition of a multibody system into a set of kinematic chains whose kinematics can be solved independently, based on group-coordinates, is introduced. The main objective is to evaluate the efficiency of this formulation considering different levels of specificity in solving the kinematic chains that defines the kinematic structure of the multibody system. To this end, two scalable systems with up to 550 coordinates have been modelled and solved with a global formulation, as a reference, and with up to four solutions based on group-coordinates formulation. The main conclusions drawn by the analysis carried out in this work shows that this topological method offers a greater modularity, flexibility and efficiency than the global method, so it may be of interest both to develop automatic modelling procedures and improve the efficiency in computation time using this formulation.

Keywords: Kinematic structure, Computational kinematics, Group coordinates.

1 INTRODUCTION

Computational kinematic analysis plays a fundamental role in the study of mechanical systems. It is not only necessary in multibody dynamics formulations, frequently is employed as a first stage in the design of mechanical systems (dimensional and/or kinematic synthesis) and, sometimes, the interest in the multibody system (MBS) is purely kinematic (position analysis, range of movement, transmission angle, etc.). Two different families of formulations are normally used in the kinematic analysis of multibody systems: global and topological.

In the global approach, a simple body joint inspection is enough to identify the degrees of freedom that are constrained by each type of kinematic pair. A set of dependent coordinates (reference point, natural or mixed) is introduced to define the model, e.g. $[x_B\ y_B\ x_C\ y_C\ \theta_1]$ in Fig.1.a. Those dependent coordinates are then related through the corresponding constraint equations due to rigid-body and kinematic-pair conditions.

Topological approaches require a detailed study of the kinematic structure of the multibody system to perform a kinematic analysis. There exist different formulations that exploit the topology of the MBS. In the method based on closed-loop identification, the closed-loops have to be opened so as to yield a tree-like structure of the mechanism (Fig.1.b), and then, the kinematic relations among bodies due to the joints connecting them can be defined, along with the loop-closure equations which relate the system dependent ($\theta_2\ \theta_3$) and independent (θ_1) coordinates [1–3]. In the method based on structural group (SG) decomposition, the multibody system is split into SG (e.g. SG-I and SG-II, Fig.1.c) and the variables and constraint equations that each SG introduces into the

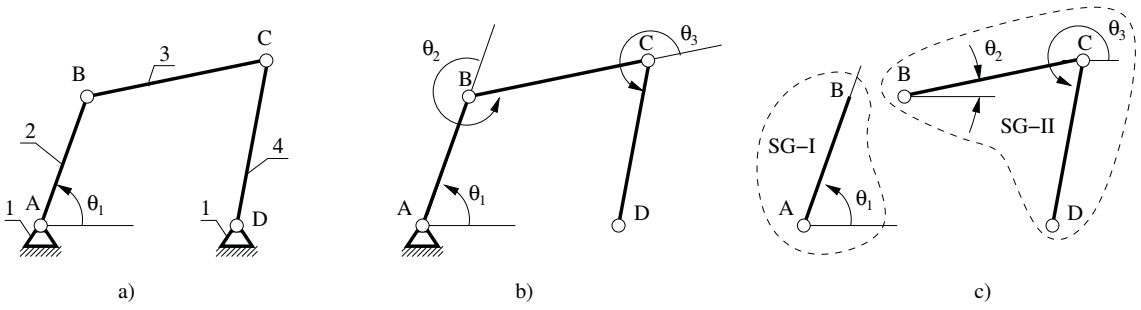


Figure 1: a) Four bar linkage with given input θ_1. b) Tree-like structure and coordinates. c) Split into structural groups: SG-I and SG-II and coordinates.

system (SG-I: $\theta_1\ x_B\ y_B$ and SG-II: $x_D\ y_D$) can be included in kinematic and dynamic formulations to obtain the response of the whole system.

For both global and topological formulations there exist a permanent interest in the scientific community in improving the efficiency of the solution and facilitating the automatic modelling of multibody systems. The global formulation is the best suited for automatic modelling of MBS [4–8] but at expenses of its efficiency. Topological formulations based on the decomposition of the system into independent closed-loops [1–3, 9, 10] are more efficient but they lack generality. The topological approach based on SG decomposition combines the advantages of both formulations and could fill the gap between the two.

Because of its interest, the main objective of this paper is to introduce and evaluate the efficiency of a topological formulation based on the kinematic structure of a MBS. To that end, section 2 resumes the basic concepts of the structural analysis and explains how to obtain the kinematic structure of a MBS. Section 3 introduces the algorithms that solve the kinematics of a MBS using its kinematic structure and section 4 defines two case studies which will be used to study the capabilities and efficiency of the proposed method. Section 5 shows and discusses the results derived from the case studies finally, the main conclusions and future developments of the present work are drawn in sections 6 and 7.

2 KINEMATIC STRUCTURE OF A MULTIBODY SYSTEM

The theory of Structural Analysis defines a Structural Group as any kinematic chain whose number of independent chain inputs n_c coincides with its mobility L_c ($n_c = L_c$). The kinematic chains which satisfy this condition and have neither excessive constrains, nor additional DOF due to special geometric considerations among their bodies, are defined as *normal* SG [11], *statically determined* SG [12], or *desmodromic* kinematic chains. Furthermore, kinematic chains that cannot split into SG of smaller number of bodies are denominated *simple* SG. From the *simple* SG condition and using the Grübler criterion to determine the mobility of a kinematic chain, a useful expression to check whether a given kinematic chain forms a SG or not is obtained (Eq.1). In Eq.1, S_c indicates the number of degrees of freedom allowed by the P kinematic pairs formed by the N_m mobile bodies.

$$S_c - n_c = 3 \cdot (P - N_m) \tag{1}$$

2.1 Obtaining the MBS Kinematic Structure

The kinematic structure of a multibody system defines which SG it is composed of and the specific order in which their SG kinematics have to be solved. Both graph-analytical and computational methods ([11] and [13] respectively) can be employed to obtain the kinematic structure of the MBS; due to its simplicity the former one is introduced and applied to a four-bar linkage (Fig.2.a).

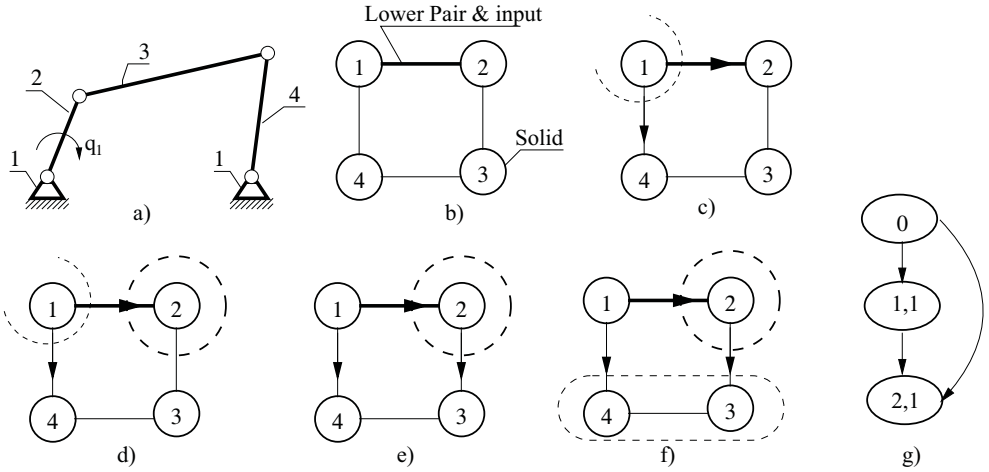


Figure 2: Four-bar linkage. a) Kinematic graph. b) Structural graph. c) to f) Steps to perform structural analysis through its structural graph. g) Structural diagram.

In the graph-analytical method, the topology of the MBS is represented by its *structural graph* (Fig.2.b): vertices correspond to bodies and edges to kinematic pairs. The number of edges connecting two vertices equals the degrees of freedom (DOF) or relative movements allowed between them. Finally, a number of these edges, equal to the n_c independent movements defined between the bodies of the kinematic pair, become bold lines referred to as root edges.

The kinematic structure of the MBS is obtained in a very simple manner as depicted in Fig.2.c-f following four basic steps. **First step:** frame 1 isolation and DOF assignment (Fig.2.c). The DOF allowed by each pair in which the frame participates are assigned to the bodies (2 and 4) that forms a kinematic pair with the frame (directed edge). The later bodies become *candidates* to be a SG. **Second step:** Search for a SG from shorter to larger length. Each one of the candidates is checked to satisfy Eq.1. Here, body 2 is selected (Fig.2.d). The number P of kinematic pairs in which the bodies of the kinematic chain participate are accounted for as the sum of the internal pairs and, from the external pairs, only those with a directed edge (a DOF has been assigned). Thus, for this solid we find that condition in Eq.1 is satisfied and this body is SG, as: $P = 1$, $S_c = 1$, $n_c = 1$, $N_m = 1$. **Third step:** Re-assign DOF. If a kinematic chain forms a SG, the DOF of its external pairs are assigned to the corresponding external bodies. In the example, body 2 is a SG and assigns the DOF $(2-3)$ to the body 3, which now is a new *candidate* (Fig.2.e). There are no more assignments. **Fourth step:** Turn to Step 2. Bodies 3 and 4 are *candidates*. Starting from one candidate, e.g. body 3, the parameters of this kinematic chain are: $S_c = 1$; $n_c = 0$; $N_m = 1$; $P = 1$. After substituting in Eq.1, body 3 shows not to be SG. Body 4 has the same parameters than body 3 so it is not a SG either. As it is not possible to form SG with a single body, larger chains have to be considered. Starting from a candidate, e.g. body 3, the chain is expanded by selecting another body that forms a kinematic pair with the candidate. The chain $3-4$, whose parameters are: $S_c = 3$; $n_c = 0$; $N_m = 2$; $P = 3$ satisfies equation Eq.1 and therefore is a SG (Fig.2.f).

2.2 Structural diagram

The kinematic structure of a mechanism is graphically represented by its *structural diagram* (Fig.2.g). It is composed by as many circles as SG have been obtained plus one, corresponding to the frame, which is identified with the number 0. The two parameters inside each circle (N_m, n_c) corresponds to the number of movable bodies and input movements of the SG. An arrow joins two circles if any of their bodies forms a kinematic pair, and is directed in the same way that the DOF which have been assigned during the structural analysis, showing the order in which the SG have been obtained and dictating the sequence in which their kinematics have to be solved.

3 COMPUTATIONAL KINEMATICS BASED ON STRUCTURAL GROUPS

In this section we introduce a method for the kinematic analysis of MBS that takes into consideration its kinematic structure. Two algorithms are described: the sequence defined in a main program to solve the whole MBS, and a generic subroutine needed to solve the kinematics of any SG.

3.1 Overall procedure

The general sequence in the kinematic analysis of a MBS can be followed in Algorithm 1. After a data file that models the MBS has been executed, the main program includes three loops. At each time step (first loop) the simulation time is increased and the values of the independent coordinates of the whole system are defined. Then, for each SG in the kinematic structure (second loop) the SG is identified and, depending on its kind (third loop), the appropriated subroutine is called to solve its kinematics. In Algorithm 2 the aspect of subroutine 3RSG called from the main program is shown; this subroutine solves the kinematics of 3RSG as explained in the next section.

Algorithm 1: Kinem: Topol. SG solution

```
%Read data MBS;
MBDatos;
%MBS Kin. analysis%%
for t = t0 : timeStep : tf do
    /* set indep.values */
    z = z + Δz
    for ng = 2 : length(MGroups) do
        /* solve each SG */
        switch MGroups(ng).kind do
            case MGroups(ng).kind == 1RSG
             | CALL Solve_1RSG(*ARGS)
            case MGroups(ng).kind == 3RSG
             | CALL Solve_3RSG(*ARGS)
        endsw
    end
end
```

Algorithm 2: Kinem. Structural Group solution

```
Solve_3RSG(*ARGS)      /* funct.CALL */
```

%% **I. Position problem** %
evaluate $\mathbf{\Phi} \rightarrow$ mFi
$error = norm(\mathbf{\Phi})$
while $error > tolerance$ **do**
  evaluate $\mathbf{\Phi_q} \rightarrow$ Jacob
  extract $\mathbf{\Phi}_\mathbf{q}^d$
  solve $\mathbf{q}_k^d = \mathbf{q}_{k-1}^d - \left(\mathbf{\Phi}_\mathbf{q}^d\right)_{k-1}^{-1} \cdot \mathbf{\Phi}_{k-1}$
  evaluate $\mathbf{\Phi} \rightarrow$ mFi
  $error = norm(\mathbf{\Phi})$
end

%% **II. Velocity problem** %%%
evaluate $\mathbf{\Phi_q} \rightarrow$ Jacob
extract $\mathbf{\Phi}_\mathbf{q}^d$; extract $\mathbf{\Phi}_\mathbf{q}^i$
solve $\dot{\mathbf{q}}^d = -\left(\mathbf{\Phi}_\mathbf{q}^d\right)^{-1} \mathbf{\Phi}_\mathbf{q}^i \dot{\mathbf{q}}^i$

%% **III. Acceleration problem** %%%%
evaluate $\dot{\Phi}_\mathbf{q}\dot{\mathbf{q}} \rightarrow$ Fiqpqp
evaluate $-\left[\mathbf{\Phi}_\mathbf{q}^i \ddot{\mathbf{q}}^i + \dot{\Phi}_\mathbf{q}\dot{\mathbf{q}}\right]$
solve $\ddot{\mathbf{q}}^d = -\left(\mathbf{\Phi}_\mathbf{q}^d\right)^{-1} \left[\mathbf{\Phi}_\mathbf{q}^i \ddot{\mathbf{q}}^i + \dot{\Phi}_\mathbf{q}\dot{\mathbf{q}}\right]$

3.2 Kinematic analysis of a SG

From the main program, the kinematics of each SG is solved by calling the corresponding subroutine. In order to solve each SG, the appropriate set of group coordinates $\mathbf{q}_G$ are selected and the corresponding constraint equations $\mathbf{\Phi}$ are defined. The specific subroutine can be programmed according to the following steps.

Identify the group coordinates and parameters: A local coordinate system attached to each body is defined and the appropriate set of coordinates (of any kind) that defines the kinematic chain is selected. We introduce two subsets of group coordinates: dependent $\boldsymbol{\varphi}$ and independent $\mathbf{h}$. The later might differ from the independent coordinates of the whole system (referred to as $\mathbf{z}$ in many recursive formulations). Other parameters which will depend on the specific SG to solve have to be identified from the geometry of the problem and the results of the computational

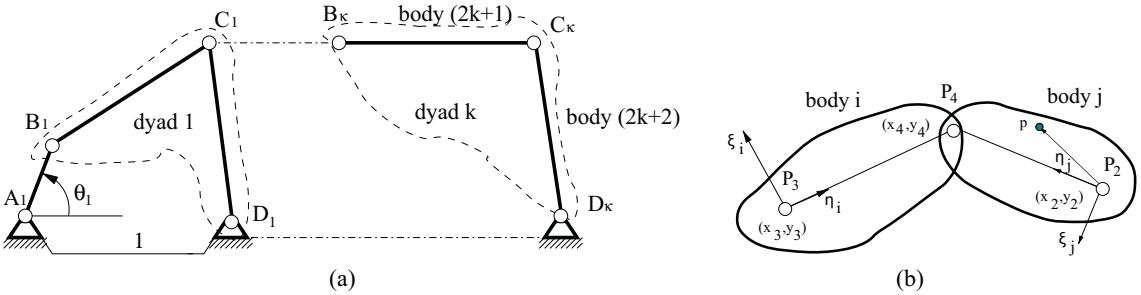


Figure 3: a) Scalable four-bar linkage is formed by a crank and one dyad. A number (k) of dyads, as many as required, can be added to the former one in order to control the number of coordinates of the model. b) Local coordinate systems and group coordinates in a 3R structural group.

structural analysis i.e. reference points needed to identify the $\mathbf{h}$ coordinates, dimensions of the bodies, and so on. As an example, a 3R Assur SG is shown in Figure 3.b. The local systems $(\{\eta_i\,\xi_i\},\{\eta_j\,\xi_j\})$ are attached to the bodies $(i,\,j)$ and the dependent $\boldsymbol{\varphi} = [x_4, y_4]$, and independent group coordinates $\mathbf{h} = [x_3, y_3, x_2, y_2]$ are defined.

Solve the position problem for the SG: To solve the position problem of any SG, the corresponding constraint equations, in accordance to the selected type of coordinates, have to be defined (Eq.2). For the given set of constraint equations, the terms of the Jacobian matrix $\boldsymbol{\Phi}_{\boldsymbol{\varphi}}$ can be analytically or numerically obtained and the Newton-Raphson iterative method can be applied to obtain the values of the dependent group coordinates at each k iteration step (Eq.2). Depending on the SG geometry, an explicit solution of the position problem might be possible and should be taken into account to reduce the computation time.

$$\boldsymbol{\Phi} = 0 \quad \rightarrow \quad \boldsymbol{\varphi}_k = \boldsymbol{\varphi}_{k-1} - \left(\boldsymbol{\Phi}_{\boldsymbol{\varphi}}\right)^{-1}_{k-1} \cdot \boldsymbol{\Phi}_{k-1} \tag{2}$$

Solve the velocity problem: As the values of the independent group velocities $\dot{\mathbf{h}}$ are known, the velocity problem can be formulated by deriving the constraint equations with respect to time, and solved for the dependent ones (Eq.3). Not only the Jacobian matrix $\boldsymbol{\Phi}_{\mathbf{h}}$, but the whole expression $-\left(\boldsymbol{\Phi}_{\boldsymbol{\varphi}}\right)^{-1}\boldsymbol{\Phi}_{\mathbf{h}}$ can be analytically obtained in 2D and 3D structural groups with a reduced number of constraint equations.

$$\dot{\boldsymbol{\Phi}}(\mathbf{q},t) = 0 \quad \rightarrow \quad \dot{\boldsymbol{\varphi}} = -\left(\boldsymbol{\Phi}_{\boldsymbol{\varphi}}\right)^{-1}\left[\boldsymbol{\Phi}_{\mathbf{h}}\dot{\mathbf{h}}\right] \tag{3}$$

Solve the acceleration problem: The acceleration problem for the dependent group coordinates can be solved by deriving the velocity constraint equations with respect to time (Eq.4). Again, if the matrices involved show a reduced dimension, most of the calculations can be analytically performed and included into each SG subroutine so that a solver is not needed.

$$\boldsymbol{\Phi}_{\boldsymbol{\varphi}}\ddot{\boldsymbol{\varphi}} + \dot{\boldsymbol{\Phi}}_{\mathbf{q}_G}\dot{\mathbf{q}_G} = 0 \quad \rightarrow \quad \ddot{\boldsymbol{\varphi}} = -\left(\boldsymbol{\Phi}_{\boldsymbol{\varphi}}\right)^{-1}\left[\boldsymbol{\Phi}_{\mathbf{h}}\ddot{\mathbf{h}} + \dot{\boldsymbol{\Phi}}_{\mathbf{q}_G}\dot{\mathbf{q}}_G\right] \tag{4}$$

Solve the kinematics of other POIs: Apart from the dependent coordinates, the results of other points of interest (POIs) might be necessary (i.e. centre of mass, or reference points for other SG). The position, velocity and acceleration of a POI that belongs to any body ($p \in j$, Figure 3.b) is easily obtained by making use of the well known equations of rigid body kinematics.

4 CASE STUDIES

To study the advantages or disadvantages that the topological formulation introduces with respect to a global one, two scalable systems are used: the four-bar linkage and the truck suspension.

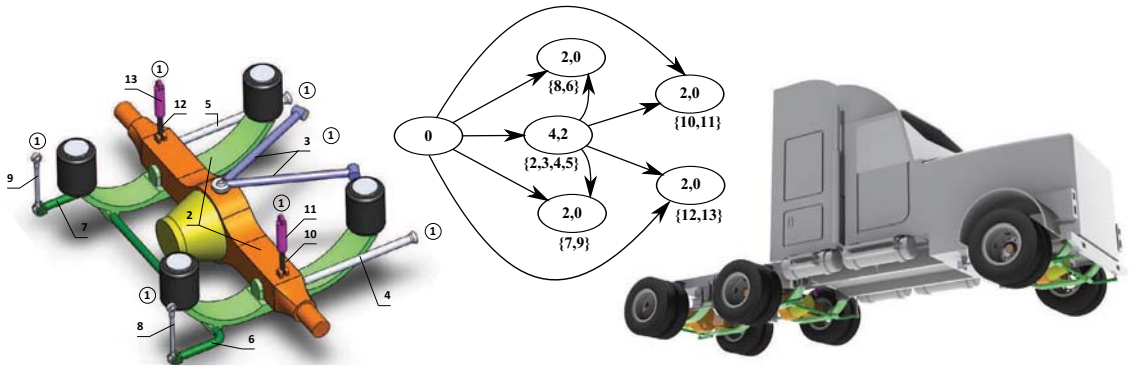


Figure 4: Bodies that forms the suspension system in one of the axes of a truck (Left). Structural diagram with SG distribution (Centre). A three axes truck (Right).

4.1 Scalable four-bar linkage

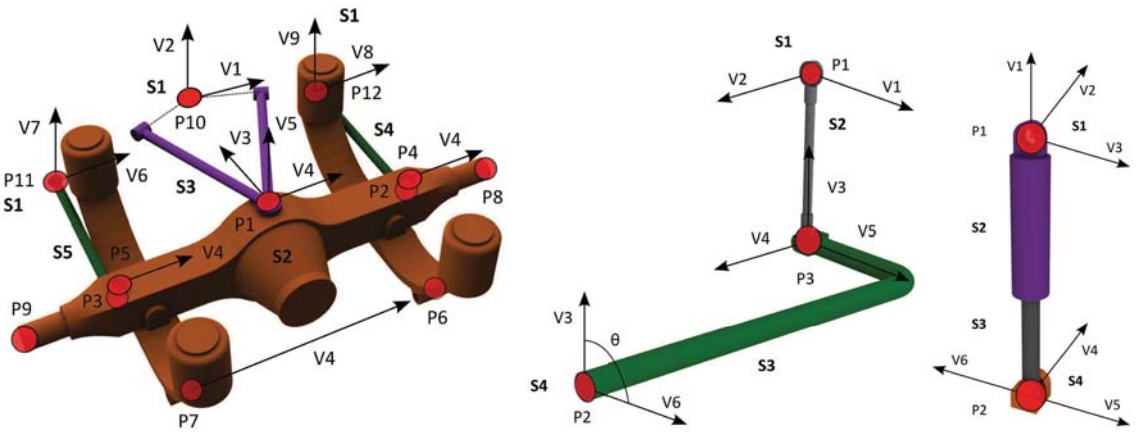
A planar four-bar linkage (Fig.3.a) consists of a crank $\overline{A_1B_1}$, a rod $\overline{B_1C_1}$ and a rocker $\overline{C_1D_1}$. To make the four-bar linkage scalable, an increasing number k of dyads have to be added. Each dyad introduces two bodies $(2k+1)$ and $(2k+2)$ joined with an internal rotation joint C_k and attached to the previous dyad and the frame with two external rotation joints B_k and D_k. This method allows us to control the number of constraint equations and coordinates in the model. To carry the kinematic analysis out, the input movement is defined as the crank rotation at a constant velocity $\dot{\theta}_1 = 1$ rad/s. All lengths are set equal to 6, except $\overline{A_1B_1} = 2$. In order to let the results be comparable against the second case study, the crank will complete 4.77 turns so that the simulation time lasts for 30 seconds.

4.2 Truck suspension

The second MBS corresponds to the suspension system of a truck axes; it is a scalable MBS as an increasing number of axes can be included in the model. Each one of these axes is a two DOF MBS composed of thirteen bodies and different kind of kinematic joints. Figure 4 shows the different bodies that form each one of the axes (left), its structural diagram (center), and a truck with three axes (right).

The structural diagram shows how many SGs forms a multibody system and the order in which their kinematics has to be solved. As it can be seen in the figure, one axes is formed by five groups: SG-I:$\{2,3,4,5\}$, SG-II:$\{6,8\}$ which is similar to SG-III: $\{10,11\}$, and SG-IV:$\{7,9\}$ which is similar to SG-V:$\{12,13\}$ (Fig. 5):

- **SG-I.** This is a two DOF structural group with four bodies $S2-S5$ (Fig.5.a). Joints: cardan ($S2-S4$ and $S2-S5$), spherical ($S1-S4$, $S1-S5$ and $S2-S3$), and revolute ($S1-S3$). The independent coordinates are defined as the vertical displacement of both ends of the axes (points P_8 y P_9). This SG is modelled with 8 points and 9 vectors. Other points of interest P_4-P_7 are needed to solve the kinematics of the remaining part of the SG.
- **SG-II.** This is a SG with null mobility and two bodies $S2$ y $S3$ (Fig.5.b). Joints: cardan ($S2-S3$), spherical ($S1-S2$) and revolute ($S3-S4$), being $S4$ body $S2$ in SG-I. This SG is modelled with 3 points, 5 vectors and an additional coordinate θ which defines the rotation of $S3$ relative to $S4$.
- **SG-III.** This is a SG with null mobility and two bodies, $S2$ y $S3$ (Fig.5.c.). Joints: spherical ($S1-S2$), prismatic ($S2-S3$) and cardan ($S3-S4$), being $S4$ body $S2$ in SG-I. This SG is modelled with 2 points, 5 vectors and an additional coordinate s which defines the displacement of $S2$ relative to $S3$.



(a) SG $\{2,3,4,5\}$ (b) SG $\{6,8\}$ (c) SG $\{10,11\}$

Figure 5: Kinematic scheme of the truck suspension system.

4.3 Methods

A total of six analyses have been carried out in this work to study the main differences between a global formulation and a topological one based on structural groups. These analyses are labelled using the following acronyms: FBL (four-bar linkage), TS (truck suspension), GB (global formulation), TP (topological formulation), and a reference to two sparse solvers: MA27 and MA28.

The purpose of the Global formulation in FBL-GB_MA27 and TS-GB_MA27 is to serve as reference analysis of the four-bar linkage and the truck suspension system, respectively. As many global formulations do, each time the system of equations has to be solved for position, velocity and acceleration analysis, the elements of each vector and matrix involved in those analyses are obtained by calling specific-purpose subroutines; then the whole MBS is modelled and solved. For example, to obtain the rigid body constraint of a body defined by two points in natural coordinates, in a planar MBS, an specific subroutine can be called from the main program. By calling this subroutine, with the corresponding arguments, all rigid-body constraints, in the constraints vector, can be automatically evaluated (i.e. Algorithm 3 and 4).

Algorithm 3: Global: deriving Fi

```
FUNCT. evalRestrics
```

%% **I. Rigid Body restrics** %

for $i = 1:1:numRigBdy$ **do**

```
    bdy=body(i);ptB=pointB(bdy)
    ptC=pointC(bdy)
    CALL restrRBdy2P(Fi,ptB,ptC,bdy)
    FiGlo(id(i),1)=Fi
```

end

Algorithm 4: Evaluate Rigid Body Restriccs

```
FUNCT. restrRBdy2P(Fi,P1,P2,body)
L = length(body)
```

$\texttt{Fi} = (P2(1)-P1(1))^2+(P2(2)-P1(2))^2-L^2$

FBL-TP analysis: As the structural groups in the four-bar linkage have a reduced number of coordinates q_G, the vector of constraint equations $\mathbf{\Phi}$, the Jacobian matrix $\mathbf{\Phi}_{q_G}$, and the vector $\dot{\mathbf{\Phi}}_{\mathbf{q}_G}\dot{\mathbf{q}}_G$, have their elements defined with symbolic expressions. Moreover, the inverse of the Jacobian matrix, $\mathbf{\Phi}_{q_G}^{-1}$, has also been defined so that a solver is not needed for the kinematic analysis.

TS-TP2_MA27 analysis: for the truck suspension system, the same vectors and matrices as in FBL-TP analysis, with exception of the inverse matrix, has their elements expressed in symbolic form. Solver MA27 (for symmetric semi-definite positive systems) is used to solve the system of equations, so that the products $\mathbf{\Phi}_{q_G}^T \times \mathbf{\Phi}_{q_G}$ and $\mathbf{\Phi}_{q_G}^T \times \mathbf{\Phi}$ have also been defined symbolically. The analysis TS-TP3_MA28 only differs from TS-TP2_MA27 in the selected solver. Finally, analy-

sis TS-TP1_MA27 differs from TS-TP2_MA27 in that the elements of the vectors and matrices involved in the kinematic analysis are not defined with symbolic expressions, but numerically evaluated as previously shown for the Global formulation. In that sense, the only difference between analyses TS-GB_MA27 and TS-TP2_MA27 is that the former solves the whole system of equations of the complete MBS, and the later solves the MBS by calling each of the SG in the kinematic structure sequentially. This analysis is important as it demonstrates that even though symbolic expressions for vectors and matrices in a SG had not been obtained, it is possible to solve this SG as global formulations would do, while others SG can be solved using a symbolic form.

All the analyses are programmed in FORTRAN, compiled with MS Visual Studio in RelWithDebInfo mode and run on a Intel Core i5-2400 CPU 3.10 GHz, RAM 16 GB, and Windows7 SP1 64 bits. All the simulations run from $t_{ini} = 0\ s$ to $t_{final} = 30\ s$ and all the MBS have been modelled with mixed (natural and relative) coordinates.

5 RESULTS AND DISCUSSION

In order to the efficiency of the different solutions be comparable, the position, velocity and acceleration of selected variables from each MBS are compared. As an example, Figure 6.a shows the stroke evolution of the hydraulic element ($SG\{10,11\}$) in the first axes of the truck suspension system with respect to the vertical displacement of point P_8 while point P_9 remains fixed. Recall that vertical displacement of points P_8 and P_9 defines the two DOF of each axes in the MBS. Two analyses are represented, TS-GB_MA27 and TS-TP2_MA27, and the results are identical.

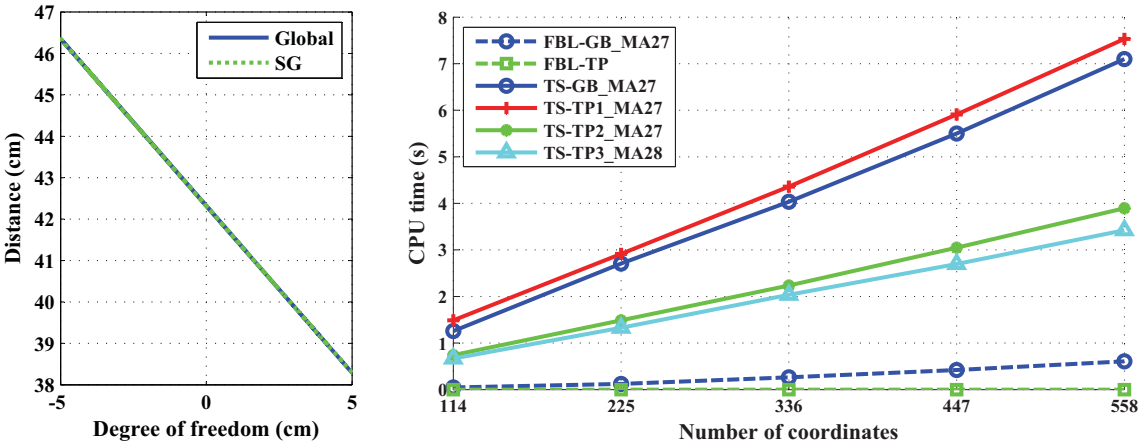
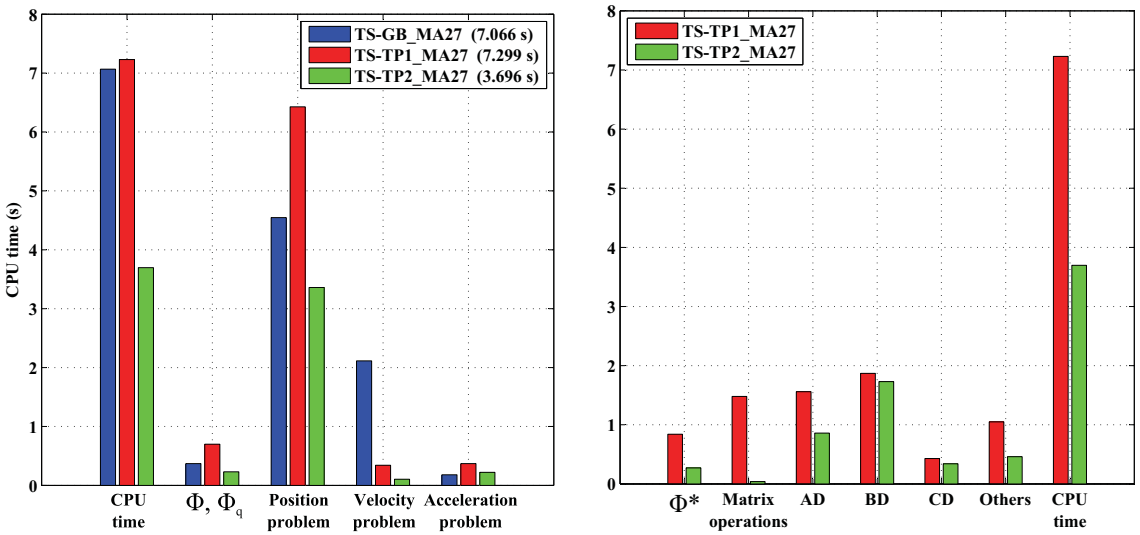


Figure 6: (Left) To validate the formulations some results are compared. (Right) CPU time versus number of model coordinates for the kinematic analysis of two case studies: Scalable four-bar linkage (FBL) and truck suspension (TS).

NUM. COORD.	FOUR-BAR LINK. GB_MA27	FOUR-BAR LINK. TP	TRUCK SUSPENSION GB_MA27	TRUCK SUSPENSION TP1_MA27	TRUCK SUSPENSION TP2_MA27	TRUCK SUSPENSION TP3_MA28
114	0.045	$\ll 0.015$	1.258	1.488	0.741	0.668
225	0.123	$\ll 0.015$	2.707	2.915	1.489	1.327
336	0.264	$\ll 0.015$	4.035	4.359	2.235	2.035
447	0.421	$\ll 0.015$	5.499	5.909	3.049	2.676
558	0.607	$\ll 0.015$	7.066	7.299	3.696	3.422

Table 1: CPU time vs number of coordinates for the kinematic analysis of the two case studies under different formulations.



(a) CPU time for differents formulations.

(b) CPU usage details for TP formulations.

Figure 7: CPU time distribution among specific phases of the kinematic analysis of the truck system.

5.1 Calculation time for different approaches

In Figure 6.b and Table 1, the calculation time (CPU time) versus the number of coordinates that defines a MBS is shown for the six kinematic analyses that have been carried out. In all cases, the CPU time evolves linearly as the number of coordinates increases. Dashed lines correspond to the four-bar linkage (FBL): the global sparse solution is represented as FBL-GB_MA27 and the topological solution as FBL-TP. In the later, the CPU time is close to zero even in models up to 550 coordinates (the number of coordinates has to be increased up to 1000 for the system to show a CPU time above 0.015 seconds). The global formulation shows a small CPU time for this mechanism for high number of coordinates.

For the truck suspension system (TS) there are four continuous lines in the graph representing: the global sparse analysis (TS-GB_MA27) and the three topological ones (TS-TP1_MA27, TS-TP2_MA27, TS-TP3_MA28), all of them defined in the previous section. This graph shows that when the vectors and matrices involved in the kinematic analysis are automatically evaluated (TS-GB_MA27 and TS-TP1_MA27) the CPU time is 1.8 times slower than the other topological solutions (TS-TP2_MA27 and TS-TP3_MA28) in which those elements are defined with symbolic expressions. Among the two later topological solutions, solver MA28 seems to be more efficient than MA27 as the number of coordinates increases although the differences are small.

It is interesting to observe that in the global formulation the CPU time is considerable lower in the four-bar linkage than in the suspension system, even though both MBS are defined by the same number of coordinates and make use of the same formulation and solver. When a profile tool is executed to evaluate how the CPU time is distributed among the different subroutines, it can be seen that the main differences among the two solutions are due to the higher sparsity of the Jacobian matrices in the four-bar linkage solution.

5.2 CPU time distribution for different approaches

In order to study more in depth the efficiency of these methods, the Visual profile tool has been used to watch how the CPU time is distributed among all the operations involved in the analysis of the truck suspension system (Fig.7.a): the global formulation TS-GB_MA27 and the topological solutions TS-TP1_MA27 and TS-TP2_MA27.

Together with the total CPU time, this figure shows four main calculations during the solution process: evaluate restrictions $\mathbf{\Phi}$ and obtain the Jacobian matrices $\mathbf{\Phi}_q$, and solve for the position, velocity and acceleration problem. The position problem consumes most of the calculation time due to the high dimension matrix operations in an Newton-Raphson iterative algorithm with a reduced tolerance ($1 \cdot 10^{-8}$). In the case of TS-TP2_MA27, this represents most of its total CPU time as all the vectors and matrices needed in the analysis (except the inverse of the Jacobian matrix) are defined in the corresponding subroutines.

In all the topological methods the acceleration problem (Eq.4) takes more time than the velocity one (Eq.3), because more vectors have to be evaluated. However, in the global solution the velocity problem consumes more time due to a new factorization of the Jacobian matrix after the position problem is solved, which can be used directly in the acceleration problem.

The time consumption in two topological solutions, TS-TP1_MA27 and TS-TP2_MA27, is shown (Fig.7.b) for each of the following six calculation processes: restrictions, Jacobian and independent terms of the linear systems (identified as $\mathbf{\Phi}^*$), matrix operations, the three routines of solver MA27, denoted by **AD** (pivoting), **BD** (decomposition) and **CD** (solve), and other internal processes. Pivoting and decomposition are the most time-consuming processes, because of the need to access those routines during the Newton-Raphson algorithm in the position problem. The access to the **AD** routine in the symbolic solution spends a fewer amount of time than its counterpart, because the pivots obtained in the first iteration can be stored and used in the following ones (the same reason holds for the global solution TS-GB_MA27). But the major difference among these two topological solutions is the time spent in matrix operations which is not necessary in TS-TP2_MA27, as they have been performed in advance.

6 CONCLUSIONS

The kinematic structure of a multibody system (MBS), which can be obtained with both graph-analytical and computational methods, decomposes a MBS into kinematic chains called Structural Groups (SG), in a specific order, whose kinematic analysis can be carried out using specific-purpose subroutines. Then, solving the kinematics of each SG in the order stated by its kinematic structure, the kinematic analysis of the whole MBS can be achieved at each time step.

In this work, a topological formulation based on the kinematic structure of a MBS is presented and four different approaches that solve the kinematics of their SGs have been studied. In order to evaluate the advantages of this formulation compared to a global one, two scalable MBS with up to 550 coordinates have been modelled and solved: a planar four-bar linkage with an increasing number of bodies and the truck suspension system with an increasing number of axes.

The scalable four-bar linkage is formed by adding to a rotating crank as many SG of a certain type (called 3R-SG) as desired, to achieve an increasing number of coordinates and constraint equations. As these 3R-SG can be modelled with a set of only six natural coordinates, and only two of them are unknowns, the specific-purpose subroutine that solves the kinematics of these 3R-SG is very efficient; all the matrices, vectors and even the inverse of the Jacobian matrix of the constraint equations of this SG are expressed in symbolic form. This is the first and the most efficient approach to a topological formulation based on the kinematic structure of the MBS, and can be applied to any kind of SG defined with a reduced number of coordinates.

The kinematic structure of the truck suspension system is formed by SG of different complexity with a minimum of fifteen unknowns each, so that the inverse of the Jacobian matrix can not be expressed in symbolic form and a solver is needed; this is, however, the most common situation in spatial MBS. To deal with this situation, two different methods based on the use of specific-purpose subroutines that solve the kinematics of each SG are proposed.

In the first method, all the vectors, matrices and the products among them, that are involved in the kinematic analysis (except the inverse of the Jacobian matrix $\mathbf{\Phi}_q^{-1}$) have been defined in symbolic

form. As the topology of each SG is well known, most of these calculations can be performed in advance. Two solutions have been implemented in this symbolic form, TS-TP2_MA27 and TS-TP3_MA28, being the solver (MA27 or MA28, respectively) the only difference.

The second method, and fourth topological approach (TS-TP1_MA27) introduced in this work, uses specific-purpose subroutines to solve the kinematics of each SG, but its level of specificity is considerably reduced with respect to the other three solutions. In this case, the vectors and matrices involved in the kinematic analysis of the SG are numerically evaluated at each time step (not given in a symbolic form), making use of a set of more generic subroutines that would serve to solve any kind of SG. This method is directed towards an automatic modelling and solving of MBS.

The efficiency of the four solutions based on the kinematic structure of the MBS is compared to that of a global formulation, FBL-GB_MA27, that solves the scalable four-bar linkage, and TS-GB_MA27, which solves the truck suspension system. Moreover, a profile tool has been used for a detailed study of the time consumed in all the calculations during the kinematic analysis of the MBS. The results that have been obtained allow us to list some advantages that the topological methods based on the kinematic structure possess with respect to the global methods:

Flexibility: The topological approach allows the use of any kind of coordinates **q** (point reference, relative, natural or mixed coordinates) to model a structural group and solve its kinematics.

Modularity: The topological method introduced in this work is a modular approach. The kinematic analysis of each SG can be programmed, optimized and compiled in an independent subroutine which might be included in a extensive library of Structural Groups. This modularity offers several advantages: facilitates the modelling and solving of any MBS, and the analysis is fast and reliable.

Generality versus Efficiency: Depending on the number of coordinates needed to model each SG, different methods can be used to solve its kinematics. These methods go from FBL-TP with a maximum efficiency and the higher level of specificity, using symbolic expressions to evaluate the dependent coordinates, to a more generic and less efficient method (TS-TP2_MA27) which can be used for the solution of any kind of SG. The former is devoted to a fast and efficient solution (only available in SG with a reduced number of coordinates) while the later tends to the automatic modelling of the MBS. Intermediate solutions, TS-TP2_MA27 and TS-TP1_MA28, reduce the CPU time by half with respect to TS-TP2_MA27 and the global approach TS-GB_MA27, independently of the number of coordinates in the MBS. The use of symbolic software seems to be the solution to take profit of the potential advantages of the generic method and obtain specific symbolic solutions.

In addition, as the kinematic analysis of each SG is performed independently from the others, the efficiency of any SG subroutine can be improved by selecting the most appropriated solver depending on the structure of the Jacobian matrix of the system of constraint equations. For example, MA27 if $\mathbf{\Phi}_q$ is semi-positive definite, or MA28 if it is not. The LU factorization of the Jacobian matrix of each SG can be stored and used at any time step, which improves efficiency.

7 FUTURE DEVELOPMENTS

The topological formulation based on the kinematic structure of the MBS has shown many advantages with respect to global formulation. This fact encourages us to improve the possibilities it brings to the computational kinematic and dynamic analysis of MBS, as well as to solve its drawbacks.

As this is a modular approach, the automatic modelling and solving of a MBS can be improved in two ways: making use of symbolic software to obtain an efficient and optimized solution to the kinematics of any SG, and making use of the kinematic structure of the MBS, obtained with computational methods, to automatically define the analysis sequence.

Apart from improving the level of automation, the efficiency of this method could be also improved by including parallel processing and optimizing the use of the best-suited solver to each SG: conventional linear solvers (LAPACK) or other sparse solvers (PARDISO, WSMP or MUMPS) are only some examples.

Finally, a well known drawback related to all the topological formulations is that the efficiency of the solution, and even the capability to find a solution itself, depends on the kinematic structure of a MBS, which is not unique, and might change during the analysis. Methods to find the more efficient kinematic structure and to allow the solution procedure to change to a different kinematic structure, at any time step, must be developed to make this formulation both efficient and general.

REFERENCES

[1] S. Kim and E. J. Haug, "A recursive formulation for flexible multibody dynamics, part II: Closed loop systems," *Comput. Methods Appl. Mech. Eng.*, Vol. 74, pp. 251–269, 1989.

[2] J. Samin and P. Fisette, *Symbolic modeling of multibody systems*. Kluwer Academic Publishers, 2003.

[3] A. Kecskeméthy, T. Krupp, and M. Hiller, "Symbolic processing of multiloop mechanism dynamics using closed-form kinematics solution," *Multibody System Dynamics*, Vol. 1, No. 1, pp. 23–45, 1997.

[4] A. A. Shabana, *Dynamics of multibody systems*, 2nd ed. Cambridge Univ Press, 1998.

[5] M. Gerardin and A. Cardona, *Flexible multibody dynamics: A finite element approach*. John Wiley & Sons, 2001.

[6] E. J. Haug, *Computer-aided kinematics and dynamics of mechanical systems*. Allyn and Bacon, Boston, MA, 1989.

[7] P. E. Nikravesh, *Computer aided analysis of mechanical systems*. Prentice Hall, Englewood Cliffs, NY, 1988.

[8] J. G. Jalon, *Kinematic and dynamic simulation of multibody systems*. Springer-Verlag, NY, 1993.

[9] M. Leger and J. McPhee, "Selection of modeling coordinates for forward dynamics multibody simulations," *Multibody System Dynamics*, Vol. 18, No. 18, pp. 277–297, 2007.

[10] J. Cuadrado, D. Dopico, M. Gonzalez, and M. A. Naya, "A combined penalty and recursive real-time formulation for multibody dynamics," *Journal of Mechanical Design*, Vol. 126, No. 4, pp. 602–608, 2004.

[11] M. Z. Kolovsky, A. N. Evgrafov, Y. A. Semenov, and A. V. Slousch, *Advanced theory of mechanism and machines*. Springer, 2000.

[12] A.Sljoka, O.Shai, and W.Whiteley, "Cheking mobility and decompositions of linkages via peeble game algorithm," in *Proceedings of the ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Washington, DC, USA, 2011, pp. 1–10.

[13] M. Saura, A. Celdran, D. Dopico, and J. Cuadrado, "Computational structural analysis of planar multibody systems with lower and higher kinematic pairs," in *Proceedings of the 2nd Joint International Conference on Multibody System Dynamics*, Stuttgart, Germany, 2012.