Formal Kinematic Analysis of Epicyclic Bevel Gear Trains

Kubra Aksoy(✉), Adnan Rashid, and Sofiène Tahar

Department of Electrical and Computer Engineering, Concordia University, Montreal, QC, Canada
{k_aksoy,rashid,tahar}@ece.concordia.ca

Abstract. Epicyclic Bevel Gear Trains (EBGTs) play a vital role in providing highly efficient solutions for power transmissions between shafts in various engineering applications, such as wind turbines and jet airplane engines. The kinematic analysis of EBGTs involves identifying fundamental cycles, and utilizing screw theory to understand the velocities and relative motion of the system's components. In this paper, we propose to use higher-order-logic theorem proving for the formal kinematic analysis of EBGTs. In particular, we formalize a directed graph representation of EBGT systems, consisting of links and joints (pairs). Next, we formalize a corresponding cycle matrix form of the EBGT in order to analyze fundamental cycles in the graph. Moreover, we formalize the linear and angular velocities of the joint components in the systems using various notions of screw theory, such as screw and twist. We use the above to formally verify the kinematic equations providing a sound relationship between the relative angular joint velocities. In order to illustrate the utility of our proposed formalization, we formally analyze the Bendix wrist, which is a well-known industrial geared mechanism, by providing a verified solution of its kinematic equations.

Keywords: Epicyclic Gear Trains · Kinematic Analysis · Cycle Matrix · Screw Theory · Higher-Order Logic · Theorem Proving · Isabelle/HOL

1 Introduction

Epicyclic Bevel Gear Trains (EBGTs) [22], known as transmission mechanisms, are composed of gear pairs that have intersecting axes, and at least one gear axis is in circular motion with respect to the ground/fixed link of the mechanism. EBGTs are integrated to a variety of mechanisms for the purpose of transmitting rotational motion and/or adapting the speed of various components of mechanisms. EBGTs are used in a wide range of engineering and mechanical applications, such as automotive, avionics, aerospace and renewable energy, thanks to their advantages like altered speed ratios and higher efficiency of the power transmission. For instance, EBGTs are of key importance in the design of robotic wrists for power transmission, which assist in reducing the weight and inertia of

K. Ogata et al. (Eds.): ICFEM 2024, LNCS 15394, pp. 162–180, 2024.
https://doi.org/10.1007/978-981-96-0617-7_10

a robotic manipulator and thus enhancing the efficiency of the mechanism [18]. Due to their aforementioned abilities and usage in safety-critical applications, e.g., wind turbines and jet airplane engines, their kinematic analysis becomes quite significant since the occurrence of any unexpected behavior may result in financial loss and even fatalities.

To perform a kinematic analysis of an EBGT mechanism, we need to develop a set of kinematic equations capturing the relative angular velocities[1] of various components of the system, such as joints and links, which are further analyzed by finding out their solutions providing a relationship between the output and input velocities. There are several techniques used for the kinematic analysis including tabular methods [14], train value methods [4] and graph theory methods [20,21]. For instance, one of the commonly used methods in textbooks is the *tabular method* based on Willis' inversion method [24]. However, this method is applicable to non-parallel axis gear trains mechanisms only. Similarly, the *train value method* solely focuses on the overall angular velocity ratios, and thus cannot deal with the velocities of intermediate links. On the other hand, the *graph theory* with the fundamental cycle concept [16] is suitable for systems with plenty of gears and multitude Degree of Freedoms (DOF). Here, the analysis starts with a synthesis of the kinematic structure of the system using fundamental cycles, from which a cycle matrix, as an algebraic representation, is used to capture the relationship between the edges and the cycles. Next, by utilizing the *screw theory* [5], the velocity of each joint is represented by screws and twists based on the fixed frame of each component. Then, the orthogonality conditions are established using the cycle matrix and screw/twist matrices in order to derive the kinematic equations. Finally, these equations for the relative angular velocities of joints are analyzed to conclude the analysis.

Conventionally, EBGTs are analyzed using paper-and-pencil proof and computer-based simulation techniques. However, the former is prone to human-error, while the latter frequently relies on unverified numerical algorithms in the core of the associated tools that can introduce approximation errors, leading to potential inaccuracies in the results. In consideration of these limitations, conventional methods do not provide the necessary level of accuracy and precision required for a comprehensive analysis of systems. On the other hand, formal methods, such as higher-order-logic (HOL) theorem proving, provides a more rigorous approach by constructing computer-based mathematical modelling of systems and verifying its properties through logical reasoning. HOL theorem proving is hence well-suited for the formal modeling and analysis of EBGT mechanisms since it provides soundness and a high level of expressiveness.

In this paper, we propose to use the Isabelle/HOL [13] for performing the kinematic analysis of EBGTs. In particular, we formalize cycle matrix based directed graph models of EBGT mechanisms using higher-order logic theorem proving. In general, an EBGT can be modeled using either an undirected graph with adjacency matrix or a directed graph with cycle matrix. Unlike undirected

[1] Relative angular velocity, or rotational velocity, is the difference between the rotational speeds of two links.

graphs with adjacency matrices, a directed graph based analysis does not require additional rules to identify graph cycles, which are obtained using a cycle matrix that is further used to generate the kinematic equations of the system [19]. Furthermore, directed graphs can model symmetric and asymmetric relationships between components, making them more generic. Therefore, in this paper, we use a directed-graph based modeling approach to analyze EGBT mechanisms. Next, we formalize the screw for describing spatial geometry of local frames assigned to joints, which provides rotational and translational motion of joints. Similarly, we formalize the twist vector to model angular and linear velocity of joints. In the next step, we utilize the above formalization alongside the orthogonality condition to formalize a set of kinematic equations capturing the relative angular velocities. To demonstrate the utility of our proposed formalization, we formally analyze a Bendix wrist mechanism [19] by verifying its kinematics using Isabelle/HOL. To the best of our knowledge, there exists no formal analysis of EBGT systems that uses graph-based cycle matrices and screw theory.

The remainder of the paper is organized as follows: Sect. 2 discusses an overview of the related work regarding the formal kinematic analysis of engineering and physical systems, topology matrices and screw theory. We present a foundational formalization of EBGTs analysis in Sect. 3. As an application, in Sect. 4, we develop the formal kinematic analysis of the Bendix wrist mechanism in Isabelle/HOL. Finally, Sect. 5 concludes the paper with pointers to some future directions.

2 Related Work

Higher-order-logic theorem proving has been used for the formal kinematic analysis of engineering and physical systems. For instance, Farooq et al. [10] used the HOL Light theorem prover to formally analyze the kinematics of two-link planar manipulators. Similarly, Rashid et al. [15] formally analyzed the dynamics of robotic cell injection systems up to 4-DOF using HOL Light. Chen et al. [7] also used HOL Light to formalize a camera pose estimation algorithm based on Rodrigues formula for robotic systems. Moreover, Wang et al. [23] formally verified the inverse kinematics of a three-fingered dexterous hand by analyzing the Paden-Kahan-sub-problem based on the screw theory in HOL Light.

On the other hand, Affeldt et al. [1] developed some geometrical foundations in 3-Dimensions (3D), including rotation matrices, screw motion as well as Denavit-Hartenverg (D-H) convention for forward kinematics of robot manipulators using the Coq theorem prover. Similarly, Wu et al. [25] formalized a Jacobian matrix to perform the forward kinematic analysis of a 3-DOF planar robot manipulator in the HOL4 theorem prover. Later, Shi et al. [17] extended this work by formally verifying the kinematic Jacobian for serial manipulators using the screw-based methods. In particular, the authors formalized twists to represent relative motion of a rigid body using exponential mapping. Recently, Xie et al. [26] used Coq to formalize coordinate transformation for robots, especially for spiral motion of rigid bodies, using Rodrigues formula and homogeneous matrices.

The aforementioned methods, including the screw-based approach, are focused on formally analyzing serial robotic manipulators, and do not consider any topological aspects of the systems, which are required for a comprehensive analysis of EGBTs.

Regarding topological matrices, there exist only a few formalizations in higher-order-logic theorem provers. For example, Heras et al. [11] formalized incidence matrices for undirected graph-based representations using Coq and used it to formally analyze 2D digital image processing systems. Recently, Edmonds et al. [9] formalized incidence matrices in Isabelle/HOL to represent a design in order to verify the Fisher's inequality. However, these contributions are able to only analyze systems represented by *undirected graphs*. Moreover, they do not focus on the kinematic analysis of EBGTs mechanisms, which is the scope of the current paper.

3 Formalization of EBGTs Analysis Foundations

3.1 Formalization of Cycle Matrices

A directed graph is defined by an ordered pair $DG = (N, E)$, where N represents a set of nodes and E is a set of edges, where each of them is a pair of distinct nodes. For the case of EBGT mechanisms, in a mechanical digraph (directed graph) representation, links are represented by nodes and connectors between the links, called joints/pairs are represented by directed edges. The mechanical digraph of the mechanisms with cycles can be algebraically represented as a matrix, called *cycle matrix*. This matrix captures the relationship between the cycles and joints of a mechanism represented by a (mechanical) directed graph. A cycle matrix is mathematically defined as follows [8]:

Definition 1. *Cycle Matrix of a Directed Graph*

Consider a set of edges $\{e_1, \cdots, e_n\}$ and cycles $\{L_1, \cdots, L_l\}$. $\mathcal{C}$ is an $l \times n$ cycle matrix of the directed graph, such that

$$C_{i,j} = \begin{cases} 1 & \text{if } e_j \in L_i, \textit{and the direction of } e_j \textit{ and } L_i \textit{ are the same} \\ -1 & \text{if } e_j \in L_i, \textit{and the direction of } e_j \textit{ and } L_i \textit{ are opposite} \\ 0 & \text{if } e_j \notin L_i \end{cases}$$

Here a cycle of a graph is a closed path, defined as a finite sequence of distinct edges where the start and the end nodes of the path are the same. It is worth to note that in this paper, we consider mechanical digraphs EGBTs without self loops, and having distinct cycles in a cycle-basis[2]. This concept, known as *fundamental cycles*, enables the analysis of the system on any cycle basis, which is sufficient to understand the kinematics of the entire mechanism. Moreover, the cycle matrix of the mechanism in this concept holds an important property

[2] A cycle basis in a directed graph is defined as the set of independent cycles.

that the number of gear pairs is the same as the number of cycles in a cycle basis.

In order to formalize a cycle matrix in Isabelle/HOL, we first formally model a mechanical digraph utilizing locale modules [6] in Isabelle/HOL. A locale module provides a series of context elements, needed to structure abstract algebraic concepts. These elements, namely parameters and assumptions, are declared using the keywords `fixes` and `assumes`, respectively. Locales can be expanded by adding new parameters, definitions and assumption into existing ones, which makes them flexible and reusable.

We now introduce a locale `mech_digraph` consisting of a list of nodes ($\mathcal{N}$s) and a list of edges ($\mathcal{E}$s), as well as the component relationships as well-formed assumptions[3]. It is worth mentioning that we chose the parameter types as `real` for the purpose of labeling the nodes and thus edges.

```
locale mech_digraph =
fixes nodes_list :: nodes (Ns) and edges_list :: edges (Es)
assumes mechdg_wf: e ∈ set Es ⟹ fst e ∈ set Ns ∧ snd e ∈ set Ns ∧ fst e ≠ snd e
assumes distincts: distinct Ns distinct Es
```

where the function `set` accepts a list of nodes and edges and returns a set. Similarly, the functions `fst` and `snd` accept a pair, and extract its first and second elements, respectively. The function `distinct` takes a list and ensures that elements of the list are disjoint. Furthermore, the assumption `mechdg_wf` ensures that the digraph has no self-loop. Next, we formalize the system with cycles by adding cycle parameter and well-formed assumptions on the locale `nempty_mechdg` by ensuring a valid digraph with a non-empty list of nodes and edges.

```
locale cycle_system = nempty_mechdg +
 fixes cycle_basis :: edges list (Ls)
 assumes wf_1: ls ∈ set Ls ⟹ set ls ⊆ symcl E ∧ length ls ≤ length Es
 assumes wf_2: ls ∈ set Ls ⟹ cycle ls ∧ cycle (reverse ls)
 and distinct: distinct Ls
```

where `symcl` accepts a Cartesian set and guarantees that this set contains elements with their reversed version. The assumption `wf_1` provides a validity of every cycle by ensuring that every element of the `cycle_basis` $\mathcal{L}$s is a subset of the symmetric relation of edges. It also makes sure that the size of the cycle cannot be larger than the number of edges in a graph. Similarly, `wf_2` guarantees that every element of the cycle list is a cycle and its reverse is also a cycle. The assumption `distinct` ensures the non-repetition of cycles in the cycle basis. Additionally, we formally define a `nempty_cycle_system` locale on top of the `cycle_system`, which ensures that the cycle system has at least one cycle in the cycle basis.

[3] We abbreviate `real` as `node`, `real list` as `nodes`, `real×real` as `edge`, and `(real×real) list` as `edges` for a better readability.

```
locale nempty_cycle_system = cycle_system +
      assumes cycle_basis_nempty: ℒs ≠ []
```

We now describe the relationship between the directions of a cycle and its corresponding edges. The direction of edges in a cycle is the same as that of a cycle, called positively oriented cycle, whereas the cycle is said to be negatively oriented with the edge if the edge is in the reversed cycle. For instance, a positively oriented cycle is formalized as follows:

```
definition in_pos_cycle x es ≡ (x,es) ∈ C_p
```

where $\mathtt{C_p}$ is a set of pairs consisting of the edge and the cycle, formalized as:

```
definition C_p ≡ {(x,es). x ∈ set ℰs ∧ es ∈ set ℒs ∧ x ∈ set es}
```

Similarly, the formalization of the relation between the edge and the cycle where the edge is negatively oriented with the cycle is given as:

```
definition in_neg_cycle x es ≡ (x,es) ∈ C_n
```

where $\mathtt{C_n}$ is a set of pairs consisting of the edge and the cycle

```
definition C_n ≡ {(x,es). x ∈ set ℰs ∧ es ∈ set ℒs ∧ x ∉ set es
                                                  ∧ x ∈ set (reverse es)}
```

Here, the function `reverse` takes a pair list and reverses its order by swapping the elements of each pair in the list. Therefore, we obtain the concept of negatively oriented cycles using the function `reverse`. Note that we explicitly indicate the direction of the cycle while assuming each edge is positively directed. Next, the cycle matrix is formalized in Isabelle/HOL as follows:

```
definition cycle_matrix :: edges list ⇒ edges ⇒ real mat
    where cycle_matrix Ls Es ≡ mat (length Ls) (length Es)
                        (λ(k,j). if (Es!j) ∈ set (Ls!k) then 1 else
                           if (Es!j) ∈ set (reverse (Ls!k)) then -1 else 0)
```

3.2 Formalization of Screw Theory Notions

The screw theory provides a unified framework for describing the spatial displacement (screw motion) of a rigid body, encompassing both rotational and translational components. Similarly, the linear and angular velocities of a rigid body can be expressed within a single concept using a screw, called twist. A screw [5] is mathematically defined as a dual vector consisting of two three-dimensional vectors, where the first vector represents the direction vector of a line (screw axis) and the second describes the moment vector specifying the translation along the screw axis. When a screw has zero pitch[4], resulting in pure

[4] A pitch is a scalar quantity that describes the ratio between the translational and rotational part of the screw.

rotation without translation, its six elements become mathematically equivalent to the Plücker coordinates [12]. In the analysis of EBGTs, a screw is analogous to Plücker coordinates that can define the spatial geometry of the axis z_k of the locale frames attached to the pair k [20]. The screw $\mathbf{u}^0_{c,k}$ is mathematically expressed as follows:

$$\mathbf{u}^0_{c,k} = \begin{pmatrix} \mathbf{u}^0_k \\ \mathbf{I}^0_{c,k} \times \mathbf{u}^0_k \end{pmatrix} \tag{1}$$

where the subscripts k and c refer to all pairs[5] and gear pairs, respectively. Similarly, the superscript 0 denotes that the components of the vectors in a screw are measured with respect to the base frame. The first vector in the screw, $\mathbf{u}^0_k$, describes the orientation of the unit vectors of a frame pair k with respect to the base frame 0. The first vector $\mathbf{u}^0_k$ can mathematically be derived using a *direction cosines matrix* as follows [20]:

$$\mathbf{u}^0_k = \mathbf{\Theta}_{0,k} \cdot \mathbf{u} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\varphi_{z_0,z_k} & sin\varphi_{z_0,z_k} \\ 0 & -sin\varphi_{z_0,z_k} & cos\varphi_{z_0,z_k} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{2}$$

where $\mathbf{u}$ is the unit vector along the z-axis. Moreover, $\mathbf{\Theta}_{0,k}$ is a direction cosines matrix that describes the instantaneous orientation of the z-axis in the base frame with respect to the z-axes of all pairs frames, and thus provides their rotation about the respective axes. In addition, φ_{z_0,z_k} presents the angle between the z-axes of the base and the pair frames and is equally represented by $\widehat{(z_0, z_k)}$. By simplifying Eq. (2), we obtain the first vectors of screw for EBGTs as follows:

$$\mathbf{u}^0_k = \begin{pmatrix} 0 \\ \sin(\widehat{z_0, z_k}) \\ \cos(\widehat{z_0, z_k}) \end{pmatrix} \tag{3}$$

Similarly, the second vector in Eq. (1), $\mathbf{I}^0_{c,k} \times \mathbf{u}^0_k$, called the moment of the unit vector $\mathbf{u}^0_k$, is defined as the cross product of two vectors such that

$$\mathbf{I}^0_{c,k} \times \mathbf{u}^0_k = \mathbf{r}^0_{c,k} = \begin{pmatrix} x_k - x_c \\ y_k - y_c \\ z_k - z_c \end{pmatrix} \times \mathbf{u}^0_k \tag{4}$$

Here, $\mathbf{I}^0_{c,k}$ is a distance vector between the points on the frames k and c. Moreover, the EBGT mechanism exhibits pure rotation around the x-axis and there is no translation for the components of the system along the x-axis. This implies that the vector moment of $\mathbf{u}^0_k$ is parallel to the x-axis. The generic form of the moment vector can be obtained by using Eq. (3) in Eq. (4) as follows [19]:

$$\mathbf{r}^0_{c,k} = \begin{pmatrix} (z_c - z_k) \cdot \sin(\widehat{z_0, z_k}) + (y_k - y_c) \cdot \cos(\widehat{z_0, z_k}) \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} P_{c,k} \\ 0 \\ 0 \end{pmatrix} \tag{5}$$

[5] All pairs of an EBGT mechanism consist of turning and gear pairs.

We use $P_{c,k}$ as an abbreviation of the first element of the moment vector. Moreover, it can also be expressed as the function of the pitch diameters of the mechanisms, which is further used to represent a coefficient matrix in terms of the speed ratio of gear pairs as described in Sect. 4 [19]. In general, a pitch diameter of a gear indicates the diameter of the pitch circle, which is an imaginary circle that measures the distance between a point on a tooth and its corresponding point on the adjacent tooth. It is used for the characterization of speed ratio of a gear train, which refers to the relative speed of rotation between two rotating components of a gear train. The speed ratio, also known as tooth or gear ratio, is defined as the pitch diameter of the tail component of a gear pair divided by the pitch diameter of its head component. It can be defined as $i_g = d_{g_{tail}}/d_{g_{head}}$ where $g = [g_{tail}, g_{head}]$.

Now, we formalize a screw vector in Isabelle/HOL as:

```
definition screw :: real vec ⇒ real vec ⇒ real vec
    where screw u r ≡ u @v (r ×j u)
```

where $@_v$ is the operator for appending vectors, and the operator $\times_j$ is used for the cross-product. Note that, `r` refers to the distance vector in 3D space, which will be explicitly modeled in Sect. 4. Next, we formalize the rotation system providing orientation of the unit vectors (Eq. (2)) as follows:

```
definition mat_rot_sys :: real mat ⇒ real vec ⇒ real vec ⇒ bool
        where mat_rot_sys Θ u uk ≡ (uk = Θ *v u)
```

where $\boldsymbol{\Theta}$ denotes the direction cosines matrix and $*_v$ is an operator modeling multiplication between a matrix and a vector. Utilizing this formalization, we verify the general form of angular velocities of each pair (Eq. (3)) as follows:

```
lemma fstvec_form:
  assumes dim: u ∈ unitvecs and unitz: w = vec_of_list [0, 0, 1]
      and rotx_sys: rotx (vec_first (screw u r) 3) w α
   shows u = vec_of_list [0, sin α, cos α]
```

Here, assumption `dim` ensures that `u` is 3-dimensional unit vector, while `unitz` asserts that `w` is a unit vector of the z-axis. Similarly, `rotx_sys` ensures that the orientation of `u` is about the x-axis with an angle α. The function `vec_first` accepts a vector and the number 3 and becomes a new vector partitioned according to its first 3 elements (Eq. (1)). Furthermore, the function `vec_of_list` takes a list and returns it to a vector. Next, we verify the moment vectors for EBGT systems (Eq. (5)) in Isabelle/HOL as follows:

```
lemma sndvec_form:
  assumes dimI: r ∈ points3D and xdist0: r$0 = 0
      and dim: u ∈ unitvecs
      and unitz: w = vec_of_list [0, 0, 1]
      and rotx_sys: rotx (vec_first (screw u r) 3) w α
 shows vec_last (screw u r) 3 = vec_of_list [-(r$2)*sin θ+(r$1)*cos θ, 0, 0]
```

Here, `points3D` is a set of 3 dimensional vectors and `xdist0` ensures that there is no translation on the x-axis. `vec_last` is a function that takes a vector v and a number n, and returns a partitioned vector based on the last n elements of the vector v. The verification of the above lemma is based on the `fstvec_form` lemma, alongside some reasoning on vectors, lists and sets. Next, we verify the orthogonality relationship between these vectors as follows:

```
lemma rel_fs_orth:
  assumes dim: u ∈ unitvecs and dimI: r ∈ points3D
  shows vec_first (screw u r) 3) • vec_last (screw u r) 3) = 0
```

The verification of the above lemma is based on the already verified lemmas, such as `fstvec_form` and `sndvec_form`, along with some simplifications on dot product and dimension of the screw vector. Next, the relative angular and linear velocity are defined in a compact form (block/dual vector), called twist $\mathbf{s}^0_{c,k}$ as follows:

$$\mathbf{s}^0_{c,k} = \mathbf{u}^0_{c,k} \cdot \dot{q}_k = \begin{pmatrix} \dot{\mathbf{q}}^0_k \\ \overline{\mathbf{I}}^0_{c,k} \times \dot{\mathbf{q}}^0_k \end{pmatrix} \tag{6}$$

where $\dot{q}_k$ is a scalar velocity variable of a pair (e.g., turning or gear pair) in the mechanism, called twist intensity. The twist intensities assigned to each pair describe the scalar magnitude of the motion between one component of the pair and the other. Similarly, the 3-dimensional vector $\dot{\mathbf{q}}^0_k$ represents the pairs' relative velocities that are also equal the difference between angular velocity of the pair's component (links). Now, we formalize a twist (Eq. (6)) in Isabelle/HOL as:

```
definition twist:: real ⇒ real vec ⇒ real vec ⇒ real vec
  where twist q u r ≡ q ·v (screw u r)
```

where $\cdot_v$ is an operator providing multiplication of a scalar and a vector. Similar to the screw, the orthogonality relation between the first and second vector in the twist shall be satisfied, which we verify in Isabelle/HOL as follows:

```
lemma twist_fs_orth:
  assumes dim: u ∈ unitvecs and dimI: r ∈ points3D
  shows vec_first (twist q u r) 3) • vec_last (twist q u r) 3) = 0
```

This lemma is verified using the relationship between screw and twist as well as simplifications on cross and dot product.

3.3 Orthogonality Condition for Kinematic Equations

In order to develop the kinematic equations of an EBGT system, we define the following two orthogonality conditions for angular and linear velocities of pairs, respectively.

$$[C \circ \widehat{\mathbf{u}}^0_k] \cdot \dot{\mathbf{q}}^0_k = \mathbf{0_c} \tag{7}$$

$$[C \circ \widehat{\mathbf{r}}^{0}_{c,k}] \cdot \dot{\mathbf{q}}^{0}_{k} = \mathbf{0_c} \tag{8}$$

Here, we use the symbol $\widehat{\ }$ to represent a matrix whose elements are vectors. Similarly, $\circ$ represents the Hadamard product, which is used for element-wise multiplication between matrices of the same sizes. $\widehat{\mathbf{u}}^{0}_{k}$ is a matrix whose elements are unit vectors assigned to each pair of the system, and the matrix has the same dimension as a fundamental cycle matrix C. Moreover, $\widehat{\mathbf{r}}^{0}_{c,k}$ denotes a $c \times k$ matrix whose entries are moment vectors, and $\mathbf{0_c}$ is a c-dimensional vector with zero vector entries. Note that we drop the superscript 0 from the elements of aforementioned matrices for better readability. Since Eqs. (7) and (8) describe the relative velocities and moments of unit vectors, the following conditions are satisfied for every cycle in the cycle basis:

- The sum of twist intensities $\dot{\mathbf{q}}_{\mathbf{k}} = 0$,
- The sum of twists' moments with respect to gear pairs is 0.

Utilizing the above equations, the kinematic equations for the relative velocities of joints can be developed for a given system. For instance, the first condition above provides to obtain twists of gear pairs in terms of twist of turning pairs. Similarly, the solutions of the kinematic equations can be derived in a closed-form by solving the equations for the total number of DOF in terms of speed ratios as given in Sect. 4.

The formalization of the orthogonality conditions requires the notion of the Hadamard product, which is formalized in Isabelle/HOL as follows:

```
definition hadamard_prod :: 'a :: semiring_0 mat ⇒ 'a vmat ⇒ 'a vmat
  where hadamard_prod A B = (let ra = dim_row A; ca = dim_col A in
   if ra = dim_row B ∧ ca = dim_col B
    then (mat (ra) (ca) (λ(i, j). A $$ (i,j) ·v B $$ (i,j)))
     else undefined)
```

where `dim_row` and `dim_col` are functions that accept a matrix and return the number of rows and columns in the matrix, respectively. Moreover, `'a vmat` and `'a vvec` are the abbreviations for the type synonyms of `'a vec mat` and `'a vec vec`, respectively. Next, we formalize a new operator over reals that provides the multiplication between a matrix (where its components are vectors) and a vector with scalar entries and returns a vector with elements are vectors.

```
definition mult_vmat_vec :: real vmat ⇒ nat ⇒ real vec ⇒ real vvec
  where mult_vmat_vec A n v ≡ vec (dim_row A) (λi. (vec n (λk.
                                  Σ j < dim_col A. (v $ j) * A $$ (i,j) $ k)))
```

Here, n denotes the size of the vectors in the matrix A. Next, we formalize an m-dimensional vector whose elements are n-dimensional zero vectors, denoted by 0_{vv}, as:

```
definition zero_vvec :: nat ⇒ nat ⇒ 'a zero vvec ("0vv")
  where 0vv m n ≡ vec m (λi. 0v n)
```

where 0_v `n` describes a n-dimensional zero vector. Using preceding definitions, we now formalize the generic form of the orthogonality condition, which is used to develop Eqs. (7) and (8) for the Bendix wrist mechanism (Sect. 4), as follows:

definition `orth_cond_imp`
where `orth_cond_imp A n v ≡ mult_vmat_vec A n v = 0_vv (dim_row A) n`

4 Formal Analysis of the Bendix Wrist Mechanism

In this section, we formally analyze a Bendix wrist mechanism (BWM) [19] based on the formalization that we developed in Sect. 3. A BWM is a 3-DOF rotational mechanism which consists of 6 moving links and 6 turning pairs (revolute pairs) and 3 gear pairs as depicted in Fig. 1.

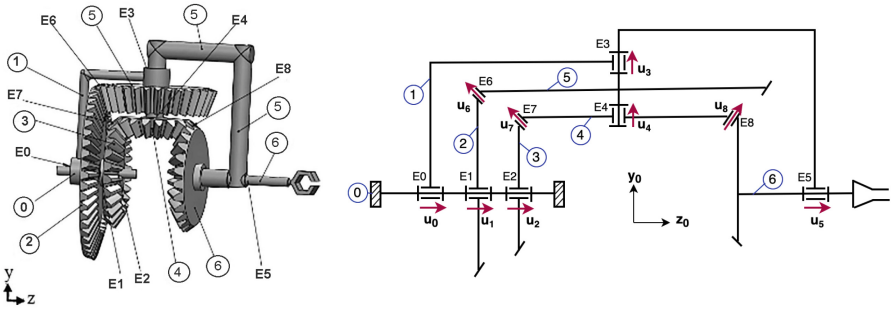


Fig. 1. Bendix Wrist Mechanism [20]

The numbering of the links starts with "0", which is assigned to the base (or forearm). The mobile links are labeled as $\{1, 2, 3, 4, 5, 6\}$, which are the geared wheels and carriers. The number of all pairs in the mechanism is equal to the number of turning pairs labeled $\{E0, E1, E2, E3, E4, E5\}$ and the number of gear pairs[6] labeled $\{E6, E7, E8\}$. Each pair in the mechanism is attached with the unit vectors to describe the direction of motion. For the above BWM, the unit vectors of turning pairs are considered as

$$\mathbf{u_0} = \mathbf{u_1} = \mathbf{u_2} = \mathbf{u_5} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \text{ and } \quad \mathbf{u_3} = \mathbf{u_4} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \tag{9}$$

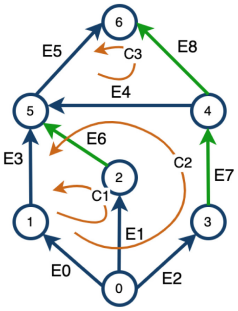
In Fig. 2, we present a directed graph of the system with nodes are labeled as mobile links, and edges labeled as pairs. The cycle basis of the system is

[6] The number of gear pairs is also the number of DOF.

{C1,C2,C3}. Each edge is represented by a pair of nodes, e.g., $E6 = (5, 6)$ and $E8 = (4, 6)$. Three independent cycles in the system are given as:

$$C1 = [E6, -E3, -E0, E1]$$
$$C2 = [E7, E4, -E3, -E0, E2]$$
$$C3 = [E8, -E5, -E4]$$

Here, the negative sign represents the opposite direction between a cycle and an edge in the cycle, e.g., $-E5$ is in cycle $C3$ but their orientations do not coincide. Furthermore, the number of independent cycles is established to be equal to the number of gear pairs in the mechanism.



(a) Directed Graph

	E0	E1	E2	E3	E4	E5	E6	E7	E8
C1	−1	1	0	−1	0	0	1	0	0
C2	−1	0	1	−1	1	0	0	1	0
C3	0	0	0	0	−1	−1	0	0	1

(b) Cycle Matrix

Fig. 2. Directed Graph of BWM and its Corresponding Cycle Matrix

We begin by formally parameterizing the graph components, i.e., nodes, edges, and cycles, which will make the rest of the formalizations easier.

```
definition cyc_sys_bwm where
  cyc_sys_bwm ≡ 𝒩s = [0,1,2,3,4,5,6] ∧
    ℰs = [(0,1),(0,2),(0,3),(1,5),(4,5),(5,6),(2,5),(3,4),(4,6)] ∧
    ℒs = [[(2,5),(5,1),(1,0),(0,2)],[(3,4),(4,5),(5,1),(1,0),(0,3)],
                                          [(4,6),(6,5),(5,4)]]
```

Here, nodes and edges are denoted as $\mathcal{N}$s and $\mathcal{E}$s, and the cycle basis is represented by $\mathcal{L}$s. Note that all formalizations related to cycles are done under the locale `nempty_cycle_system` (presented in Sect. 3.1) in order to facilitate the usage of already developed system properties, such as the dimensional and index. We then verify the cycle matrix of BWM (given in Fig. 2b) as follows:

```
lemma cycle_matrix_bwm:
 assumes cyc_sys_bwm
 shows cycle_matrix ℒs ℰs = mat_of_rows_list 9 [[-1,1,0,-1,0,0,1,0,0],
                                                [-1,0,1,-1,1,0,0,1,0],
                                                [0,0,0,0,-1,-1,0,0,1]]
```

where `mat_of_rows_list` accepts a number indicating the column size of the matrix and a list, which its elements are lists that express each row of the matrix. The verification of the above lemma involves the cycle matrix definition and its properties such as row/column and index properties, negative and positive cycle relationship and reasoning on sets and lists. Next, the first orthogonality condition (Eq. (7)) for BWM is mathematically expressed as:

$$\underbrace{\begin{bmatrix} -\mathbf{u}_0 & \mathbf{u}_1 & \mathbf{0} & -\mathbf{u}_3 & \mathbf{0} & \mathbf{0} & \mathbf{u}_6 & \mathbf{0} & \mathbf{0} \\ \mathbf{u}_0 & \mathbf{0} & \mathbf{u}_2 & -\mathbf{u}_3 & \mathbf{u}_4 & \mathbf{0} & \mathbf{0} & \mathbf{u}_7 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{u}_4 & -\mathbf{u}_5 & \mathbf{0} & \mathbf{0} & \mathbf{u}_8 \end{bmatrix}}_{C \circ \widehat{\mathbf{u}}_k^0} \cdot \dot{\mathbf{q}} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \tag{10}$$

where $\dot{\mathbf{q}} = (\dot{q}_0, \dot{q}_1, \dot{q}_2, \dot{q}_3, \dot{q}_4, \dot{q}_5, \dot{q}_6, \dot{q}_7, \dot{q}_8)^T$[7]. Moreover, the characters written in bold represent vectors. To model Eq. (10) for the case of the BWM, we first formalize the $\widehat{\mathbf{u}}_k^0$ matrix in Isabelle/HOL as:

```
definition unitvecs_mat :: real vec list ⇒ real vmat
  where unitvecs_mat U ≡
    (let units = [U!0, U!1, U!2, U!3, U!4, U!5, U!6, U!7, U!8]
       in mat_of_rows_list 9 [units, units, units]
```

Now, we formalize the first orthogonality condition (Eq. (7)) as:

```
definition fst_orth_cond :: real vec list ⇒ real vec ⇒ bool
  where fst_orth_cond U q ≡
              orth_cond_imp (hadamard_prod C (unitvecs_mat U)) 3 q
```

where `C` indicates the cycle matrix. The following lemma verifies the Hadamard product of `C` and `unitvecs_mat` utilizing the definitions of the Hadamard product, the matrices as well as their dimensions and index properties.

```
lemma had_mat_fst_bwm:
    assumes cyc_sys_bwm and units_bwm U u6 u7 u8
    shows hadamard_prod C (unitvecs_mat U) = had_mat_fst U
```

Here, the predicate `units_bwm` represents Eq. (9) along with the unknown vectors assigned to gear pairs, i.e., `u6`, `u7`, and `u8`, which are 3-dimensional unit vectors. Furthermore, we define `had_mat_fst` that takes unit vectors `U`, and returns the explicit form of the matrix $C \circ \widehat{\mathbf{u}}_k^0$, depicted in Eq. (10). The first orthogonality condition is used to derive the following set of equations in matrix form, which describes the relationship between the gear pairs and the turning pairs twists.

$$\begin{pmatrix} \dot{\mathbf{q}}_6 \\ \dot{\mathbf{q}}_7 \\ \dot{\mathbf{q}}_8 \end{pmatrix} = - \begin{bmatrix} -1 \cdot \mathbf{u}_0 & 1 \cdot \mathbf{u}_1 & \mathbf{0} & -1 \cdot \mathbf{u}_3 & \mathbf{0} & \mathbf{0} \\ -1 \cdot \mathbf{u}_0 & \mathbf{0} & 1 \cdot \mathbf{u}_2 & -1 \cdot \mathbf{u}_3 & 1 \cdot \mathbf{u}_4 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -1 \cdot \mathbf{u}_4 & -1 \cdot \mathbf{u}_5 \end{bmatrix} \cdot \begin{pmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \end{pmatrix} \tag{11}$$

[7] The superscript "0" is removed from the vectors for better readability.

Here, $\dot{\mathbf{q}}_6$, $\dot{\mathbf{q}}_7$, $\dot{\mathbf{q}}_8$ are the first vectors in the twists (relative angular velocity vectors) related to the gear pairs such that $\dot{\mathbf{q}}_6 = \dot{q}_6 \cdot \mathbf{u}_6$, $\dot{\mathbf{q}}_7 = \dot{q}_7 \cdot \mathbf{u}_7$ and $\dot{\mathbf{q}}_8 = \dot{q}_8 \cdot \mathbf{u}_8$. This relationship is verified in the following lemma.

```
lemma rel_gear_turning_pairs:
    assumes sys: cyc_sys_bwm and units: units_bwm U u6 u7 u8
    assumes twd: dim_vec q = 9
    shows fst_orth_cond U q ⟹ fst_results_eqs U q
```

where `fst_results_eqs` is formally modeling the set of equations for the relationship between the gear twists and the turning pair twists, described in a compact form in Eq. (11). The verification of the lemma utilizes the verified lemmas `had_mat_fst_bwm` and `fst_orth_eqs`[8] alongside the definition of `mult_vmat_vec` with reasoning on dimension and index.

Next, the second orthogonality condition (Eq. (8)) for BWM mechanisms can be mathematically expressed as

$$\underbrace{\begin{bmatrix} -\mathbf{r}_{0,0} & \mathbf{r}_{0,1} & \mathbf{0} & -\mathbf{r}_{0,3} & \mathbf{0} & \mathbf{0} & \mathbf{r}_{0,6} & \mathbf{0} & \mathbf{0} \\ -\mathbf{r}_{1,0} & \mathbf{0} & \mathbf{r}_{1,2} & -\mathbf{r}_{1,3} & \mathbf{r}_{1,4} & \mathbf{0} & \mathbf{0} & \mathbf{r}_{1,7} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{r}_{2,4} & -\mathbf{r}_{2,5} & \mathbf{0} & \mathbf{0} & \mathbf{r}_{2,8} \end{bmatrix}}_{C \circ \widehat{\mathbf{r}}^0_{c,k}} \cdot \begin{pmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \\ \dot{q}_6 \\ \dot{q}_7 \\ \dot{q}_8 \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \tag{12}$$

where $\mathbf{r}_{c,k}$[9] are entries of the matrix $\widehat{\mathbf{r}}^0_{c,k}$. To formally verify the second orthogonality condition, we need to formalize a few notions, such as the moment matrix and the distance vector. For instance, the moment matrix, $\widehat{\mathbf{r}}^0_{c,k}$, is formalized as:

```
definition moment_mat
      where moment_mat U G T ≡
          mat (length G) (length U) (λ(i,j). snd_vecs U G T i j)
```

Here, the function `moment_mat` accepts three real vector lists and returns them in the form of a matrix. Each element of the matrix is obtained through the function `snd_vecs` that maps each index of the moment (second) vector of the screw in the matrix. Similarly, the distance vector (Eq. (4)) is formalized as:

```
fun dist_vec where dist_vec T G i j = (T@G)!j - (G!i)
```

where `dist_vec` is a function that accepts two vector lists as 3-dimensional points in $\mathbb{R}$ and returns them into a difference vector. Next, we verify the relationship between the screw and the moment vector of the screw as follows:

[8] This lemma and more details about the proof can be found in our Isabelle/HOL script [2].

[9] For better readability, we remove the superscript "0" from the elements of $\widehat{\mathbf{r}}^0_{c,k}$.

```
lemma tw_sndvecs_bwm:
      assume units_bwm U u6 u7 u8
        shows ∧i j. i < length G ⟹ j < length U ⟹
          vec_last (screw (U!j) (dist_vec T G i j)) 3 = snd_vecs U T G i j
```

where `i` and `j` denote the indices of the lists `G` and `U`, respectively. Based on Eq. (4), the distance vector $\mathbf{l}_{c,k} = \mathbf{0}$ when the pair is the gear pair. For the BWM, this equality holds for the gear pairs in the system such that $\mathbf{r}_{0,6} = \mathbf{r}_{1,7} = \mathbf{r}_{2,8} = \mathbf{0}$. In the sequel, we verify these equations, which are further used to generate kinematic equations.

```
lemma tw_resultants_bwm:
      assume points_param T G and units_bwm U u6 u7 u8
        shows "snd_vecs U G T 0 6 = 0v 3" "snd_vecs U G T 1 7 = 0v 3"
              "snd_vecs U G T 2 8 = 0v 3"
```

Here, `points_param` ensures that the elements of the turning pair list `T` and the gear pair list `G` are 3-dimensional points represented as vectors, and the size of these lists are 6 and 3, respectively. The verification of above lemma is based on the lemma `tw_sndvecs_bwm`, definitions `snd_vecs`, `dist_vec`, and `screw`, as well as reasoning on cross product, lists and sets. Next, we formalize the second orthogonality condition (Eq. (8)) as follows:

```
definition snd_orth_cond
    where snd_orth_cond U G T q ≡
              orth_cond_imp (hadamard_prod C (moment_mat U G T)) 3 q
```

Since the moment matrix entries $r_{c,k}$ are functions of $P_{c,k}$ (see Eq. (5)), we can derive a set of equations using the second orthogonality condition as the following matrix form [19]:

$$\begin{bmatrix} -1\cdot P_{0,0} & 1\cdot P_{0,1} & 0 & -1\cdot P_{0,3} & 0 & 0 \\ -1\cdot P_{1,0} & 0 & 1\cdot P_{1,2} & -1\cdot P_{1,3} & 1\cdot P_{1,4} & 0 \\ 0 & 0 & 0 & 0 & -1\cdot P_{2,4} & -1\cdot P_{2,5} \end{bmatrix} \cdot \begin{pmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad (13)$$

Next, we verify the relation between Eqs. (12) and (13) as follows:

```
lemma snd_eqs_ver:
    assumes cyc_sys_bwm and units_bwm U u6 u7 u8 and points_param T G
    assumes dim_vec q = 9
     shows snd_orth_cond U G T q ⟹ snd_part_kin_eqs U G T q
```

Here, `snd_part_kin_eqs` represents the equations obtained from Eq. (13). The proof of the above lemma is similar to the verification of the lemma `rel_gear_turning_pairs`. Next, we represent the set of equations (Eq. (13)) in terms of

speed ratios by using the values of scalar $P_{c,k}$'s, which is the first entry of the moment vector of the screw, along each cycle. These values can be given as follows [19]:

$$\begin{aligned}
&\text{Cycle C1} \longrightarrow P_{0,0} = -0.5 \cdot d_2, P_{0,1} = -0.5 \cdot d_2, P_{0,3} = -0.5 \cdot d_5 \\
&\text{Cycle C2} \longrightarrow P_{1,0} = P_{1,2} = -0.5 \cdot d_3, P_{1,3} = P_{1,4} = -0.5 \cdot d_4 \\
&\text{Cycle C3} \longrightarrow P_{2,4} = 0.5 \cdot d_4, P_{2,5} = -0.5 \cdot d_6
\end{aligned} \tag{14}$$

where d_2, d_3, d_4, d_5 and d_6 represent the pitch diameters, which are labeled the same way as the moving links. Moreover, the speed ratios with respect to each gear (labeled E6, E7 and E8) can be defined using the pitch diameters as:

$$\begin{aligned}
&\text{Gear 0} \longrightarrow \text{E6} = (2,5) \longrightarrow \mathbf{i}_0 = d_2/d_5 \\
&\text{Gear 1} \longrightarrow \text{E7} = (3,4) \longrightarrow \mathbf{i}_1 = d_3/d_4 \\
&\text{Gear 2} \longrightarrow \text{E8} = (4,6) \longrightarrow \mathbf{i}_2 = d_4/d_6
\end{aligned}$$

Establishing the speed ratios in Eq. (13) using the Eq. (14) as well as some algebraic manipulations, Eq. (13) can be rewritten as:

$$\begin{bmatrix} -\mathbf{i}_0 & \mathbf{i}_0 & 0 & -1 & 0 & 0 \\ -\mathbf{i}_1 & 0 & \mathbf{i}_1 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{i}_2 & -1 \end{bmatrix} \cdot \begin{pmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \tag{15}$$

Next, we utilize Eq. (15) to find the solution for relative velocities, which is provided in the following vector form:

$$\begin{pmatrix} \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \end{pmatrix} = \begin{pmatrix} -\mathbf{i}_0 \cdot \dot{q}_0 + \mathbf{i}_0 \cdot \dot{q}_1 \\ (-\mathbf{i}_0 + \mathbf{i}_1) \cdot \dot{q}_0 + \mathbf{i}_0 \cdot \dot{q}_1 - \mathbf{i}_1 \cdot \dot{q}_2 \\ (\mathbf{i}_2 \cdot \mathbf{i}_1 - \mathbf{i}_2 \cdot \mathbf{i}_0) \cdot \dot{q}_0 + \mathbf{i}_2 \cdot (\mathbf{i}_0 \cdot \dot{q}_1 - \mathbf{i}_1 \cdot \dot{q}_2) \end{pmatrix}$$

The last step is to formally verify the correctness of the above solution of kinematic equations derived from the second orthogonality condition in Isabelle/HOL as:

```
theorem sol_vel_snd_kin:
    assumes pitch_diams d2 d3 d4 d5 d6 T G U q
    and d4 ≠ 0 d5 ≠ 0 d6 ≠ 0 and dim_vec q = 9
    shows rel_vel_sol U G T q ⟹ snd_part_kin_eqs U G T q
```

Here, `pitch_diams` refers to the $P_{c,k}$ coefficients (Eq. (14)). The proof process of the above theorem is similar to that of verifying the relative velocities of gear pairs using the first orthogonality condition.

To the best our knowledge, this is the first formal kinematic analysis of epicyclic bevel gear trains based on both topological matrices and screw theory

in higher-order-logic theorem proving. One of the primary distinctions of our proposed approach for formally analyzing EBGTs is its generic nature. All lemmas are verified for universally quantified variables and functions, allowing for the formal analysis of any EBGT system without the need for individual system modeling, as compared to computer-based simulation methods. For example, the Hadamard product is modeled for generic variables, and specialized for the BWM mechanism (cf. Sect. 4) to verify the lemma `had_mat_fst_bwm`. Similarly, all lemmas are verified for an arbitrary number of components of the EBGTs, which enables the analysis of large and complex systems. For example, we have formalized the cycle graph model for EBGTs systems with any number of links and pairs. Moreover, our work relied on the mathematical analysis provided in the literature, some of which were often ambiguous or lacking in rigorous details, posing significant challenges during the formalization process. However, the use of the Isabelle/HOL theorem prover ensured that every assumption or fact that may have been overlooked in a paper-and-pencil proof, is explicitly provided. The initial focus of our formalization efforts was on providing a generic directed graph model associated with cycle matrices, and combining it with screw theory. To broaden the applicability of the formalization to a variety of scenarios in the modeling and analysis of EBGT systems, we established the locale `cycle_system` based on lists. Moreover, using locale allows flexibility to expand the formalization for potential future analysis and applications. The Isabelle/HOL code for formalization and verification efforts presented in this paper are available at [2].

5 Conclusion

In this work, we proposed to use higher-order-logic theorem proving for the formal kinematic analysis of epicyclic bevel gear train (EBGT) mechanisms. These systems can be analyzed using directed graphs, associated topological matrices, and screw theory concepts. Therefore, we first formally modeled a directed graph including cycle aspects using the locale modules of Isabelle/HOL. We then formalized cycle matrices since these can fully characterize topological properties of the mechanisms, which ease the development and manipulation of kinematic equation in a compact form. We also formally modeled the screw and twist for the relative motion of joints of the mechanisms and used them alongside the cycle matrix to formalize the orthogonality conditions. Finally, we illustrated the effectiveness of our proposed formalization by formally analyzing a Bendix wrist mechanism (BWM), where we formally verified the correctness of the solution of the kinematic equations. As a future work, we plan to extend our analysis through the use of singularity analysis, which enables the detection of changes in the kinematics of the systems [3]. Another future direction would be to explore dynamical aspects of geared mechanisms in order to perform the formal dynamic analysis of more complex physical and engineering systems.

References

1. Affeldt, R., Cohen, C.: Formal foundations of 3D geometry to model robot manipulators. In: Certified Programs and Proofs, pp. 30–42. ACM (2017)
2. Aksoy, K.: Formal kinematic analysis of epicyclic bevel gear trains, Isabelle/HOL script (2024). https://hvg.ece.concordia.ca/code/Isabelle-hol/ebgt_bwm.zip
3. Amirinezhad, S.V., Donelan, P.: Kinematic singularities of a 3-DOF planar geared robot manipulator. Adv. Robot Kinematics **2016**, pp.441–449 (2018)
4. Bagei, C.: Efficient method for the synthesis of compound planetary differential gear trains for multiple speed ratio generation, and constant direction pointing chariots. In: Applied Mechanisms Conference, pp. 14–35 (1987)
5. Ball, R.S.: A Treatise on the Theory of Screws. Cambridge University Press, Cambridge (1998)
6. Ballarin, C.: Locales: a module system for mathematical theories. J. Autom. Reason. **52**(2), 123–153 (2014)
7. Chen, S., Wang, G., Li, X., Zhang, Q., Shi, Z., Guan, Y.: Formalization of camera pose estimation algorithm based on Rodrigues formula. Formal Aspects Comput. **32**, 417–437 (2020)
8. Deo, N.: Graph Theory with Applications to Engineering and Computer Science. Courier Dover Publications (2017)
9. Edmonds, C., Paulson, L.C.: Formalising Fisher's inequality: formal linear algebraic proof techniques in combinatorics. In: Interactive Theorem Proving, LIPIcs, vol. 237, pp. 11:1–11:19 (2022). https://doi.org/10.4230/LIPIcs.ITP.2022.11
10. Farooq, B., Hasan, O., Iqbal, S.: Formal kinematic analysis of the two-link planar manipulator. In: Groves, L., Sun, J. (eds.) ICFEM 2013. LNCS, vol. 8144, pp. 347–362. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41202-8_23
11. Heras, J., Poza, M., Dénès, M., Rideau, L.: Incidence simplicial matrices formalized in Coq/SSReflect. In: Davenport, J.H., Farmer, W.M., Urban, J., Rabe, F. (eds.) CICM 2011. LNCS (LNAI), vol. 6824, pp. 30–44. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22673-1_3
12. Huang, Z., Li, Q., Ding, H., Huang, Z., Li, Q., Ding, H.: Basics of screw theory. In: Theory of Parallel Mechanisms pp. 1–16. Springer, Dordrecht (2013). https://doi.org/10.1007/978-94-007-4201-7_1
13. Nipkow, T., Wenzel, M., Paulson, L.C.: Isabelle/HOL: A Proof Assistant for Higher-Order Logic. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-45949-9
14. Norton, R.L.: Design of Machinery. McGraw-Hill (2019)
15. Rashid, A., Hasan, O.: Formal verification of robotic cell injection systems up to 4-DOF using HOL Light. Formal Aspects Comput. **32**, 229–250 (2020)
16. Recski, A.: Matroid Theory and its Applications in Electric Network Theory and in Statics. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-662-22143-3
17. Shi, Z., Wu, A., Yang, X., Guan, Y., Li, Y., Song, X.: Formal analysis of the kinematic Jacobian in screw theory. Formal Aspects Comput. **30**(6), 739–757 (2018). https://doi.org/10.1007/s00165-018-0468-0
18. Staicu, S.: Planetary gear train for robotics. In: International Conference on Mechatronic, pp. 840–845. IEEE (2005)
19. Talpasanu, I.: A general method for kinematic analysis of robotic wrist mechanisms. J. Mech. Robot. **7**(3), 031021 (2015)

20. Talpasanu, I., Simionescu, P.: Kinematic analysis of epicyclic bevel gear trains with matroid method. J. Mech. Des. **134**(114501), 1–8 (2012)
21. Tsai, L.W.: The kinematics of spatial robotic bevel-gear trains. IEEE J. Robot. Autom. **4**(2), 150–156 (1988)
22. Vullo, V.: Gears: Volume 1: Geometric and Kinematic Design, vol. 10. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-36502-8
23. Wang, G., Chen, S., Guan, Y., Shi, Z., Li, X., Zhang, J.: Formalization of the inverse kinematics of three-fingered dexterous hand. J. Logical Algebraic Meth. Program. **133**, 100861 (2023)
24. Willis, R.: Principles of Mechanism: Designed for the Use of Students in the Universities, and for Engineering Students Generally. Cambridge University Press (2010)
25. Wu, A., Shi, Z., Yang, X., Guan, Y., Li, Y., Song, X.: Formalization and analysis of Jacobian matrix in screw theory and its application in kinematic singularity. In: Intelligent Robots and Systems, pp. 2835–2842. IEEE (2017)
26. Xie, G., Yang, H., Chen, G.: A framework for formal verification of robot kinematics. J. Logical Algebraic Meth. Program. **139**, 100972 (2024)