Numerical tooth contact analysis of gear transmissions through the discretization and adaptive refinement of the contact surfaces

Francisco Sanchez-Marin*, Jose L. Iserte, Victor Roda-Casanova

Department of Mechanical Engineering and Construction, Universitat Jaume I, Castellon, Spain

Abstract

The Tooth Contact Analysis (TCA) is an important resource for the design of gear drives. This widely used analysis provides the contact pattern, contact path and the function of transmission errors that are directly related to the performance of the gear set. In this work, a new geometric approach for the TCA is proposed. This approach is general, deterministic and independent from the type or alignment status of the gears. It is based on the discretization of the contact surfaces of the reference teeth pair and on a geometrically adaptive refinement to solve the contact problem and to compute the instantaneous contact area for each position of the gear set along the gearing cycle. The new algorithm demonstrated to be versatile, robust and efficient through different test cases, obtaining accurate results with a relatively low computational cost.

Keywords: gear transmissions, tooth contact analysis, contact pattern, transmission error

1. Introduction

The unloaded Tooth Contact Analysis (TCA) is a classic test applied to gear transmissions that consists in the virtual simulation of the gear meshing allowing the designer to predict the performance of the transmission. The main hypothesis of the TCA is that the tooth contact surfaces of the gears are rigid. Consequently, the TCA constitutes a pure geometric problem where no deformations are taken into account. On the other hand, the main results of the TCA are the contact pattern and the function of transmission errors. Being both related to each other, the contact pattern is mostly related to the contact pressure distribution, that is related to the durability of the gears, while the function of transmission errors is mostly related to the dynamic performance of the gear set. In summary, the TCA is an excellent tool to predict the resulting

*Corresponding author:
Email address: francisco.sanchez@uji.es (Francisco Sanchez-Marin)

performance of new designs of gears with both standard and modified tooth surfaces and to test the sensitivity of the gear meshing to the misalignment of the gears.

Regarding the contact pattern, the TCA can be described as the virtual version of the *unloaded gear tooth contact pattern test* in which one gear is painted with a marking compound and, then, the teeth are rolled through the mesh so the compound is transferred to the unpainted gear through the contact area, obtaining the contact pattern. In the real test, the result depends on the thickness of the painted layer and, correspondingly, the contact pattern computed with the TCA depends on a parameter called *virtual marking compound thickness* (δ) that represents the thickness of the real layer.

Under the consideration of the tooth contact surfaces as rigid surfaces, the instantaneous locus of contact of two mating teeth is always a point or a line. Then, the locus of points of each tooth surface around the contact point (or line) where the distance to the contact surface of the opposite tooth is lower than the virtual marking compound thickness is called *instantaneous contact area*. Taking this into account, the contact pattern on each gear can be defined as the envelope of the infinite instantaneous contact areas obtained along the gearing cycle.

The first works of development of the TCA were done by Litvin and Kai [1, 2], and Baxter [3], followed by the engineers of Gleason Works [4], Klingelnberg [5] and Oerlikon [6], where the first approximation to the problem was analytic. In these methods, the contact problem for a specific position of the gear set is solved using a surface tangency condition (coincidence of normal vectors), leading to a system of non-linear equations that, in the most general case, must be solved numerically. This allows the computation of the contact point and the transmission errors associated to that position of the gear set. After that, the instantaneous contact area (defined as an ellipse in these analytic approaches) is calculated from the relative curvature of the contacting surfaces. Using this strategy, a considerable number of researchers solved the TCA problem for a variety of gear types with different objectives, being the most common the study of the sensitiveness of the gear set to the misalignment of the gears and the use of the TCA to verify the resulting performance of the gear set under certain modifications of the tooth flanks. Thus, with the analytic approach, the TCA has been extensively applied to spur gears [7–10], helical gears [11–15], face gears [16–20], spiral bevel gears [21–25], hypoid gears [26–30] and other gear types, to cite a few works. The high number of papers reveals the importance of the TCA for the simulation and design of gear drives.

However, Lin [31] stated that the analytical methods based on the surface tangency condition may fail solving the contact problem when an edge-contact occurs during the gear tooth meshing process because that condition is not necessary fulfilled. Additionally, when there is a perfect line contact between the mating teeth (for example, in the case of perfectly aligned involute spur gears), the approach may not be robust because the contact point is not unique. And, finally, in this last case, the estimation of the instantaneous contact ellipse from the relative curvature of the surfaces may have numerical complications because the semi-major axis of the ellipse tends to infinity. In fact, the inherent difficulty of these two cases (edge contact and line contact) makes them the most difficult cases to solve for any TCA approach and, consequently, they constitute the best test cases to evaluate the robustness and numerical stability of new TCA algorithms.

To prevent the described problems, different researches proposed numerical approaches to solve the tooth contact analysis problem. Among them, Sheveleva [32] developed an algorithm based on the examination of the distance field between the mating tooth surfaces using a quad mesh in a tangent plane as reference. Kolivand [33] used the ease-off topography, surface of action and roll angle surface to predict the unloaded transmission error and contact pattern. Vijayakar [34] proposed an discrete optimization method for finding the points of minimum distance in both tooth surfaces to solve the contact problem. Lin [31] proposed the use of a numerical approach to determine the coordinates of the contacting points according to the surface position vector exclusively, avoiding the use of the normal vector. In a previous work, Bracci [35] proposed an interesting fast geometric approach for the estimation of the contact pattern without the need of taking into account the curvature of the surface. In that work, the instantaneous contact area is estimated employing a surface intersection procedure that mimics the marking compound removal during meshing. Following the line of these papers, a new approach for unloaded TCA is proposed in the present work. The objectives considered during the design of the algorithm were the following: (i) it had to be robust, general and applicable to any type of gear, including types with line contact and types with point contact; (ii) it had to work properly for any relative position of the tooth contact surfaces (from aligned to misaligned), including edge and non-edge contacts; (iii) the final accuracy had to be parameter dependent, to be able to be controlled by the client of the algorithm, and (iv) the computational cost had to be as low as possible.

The proposed method is a geometric approach based on the discretization of the tooth contact surfaces and the progressive adaptive refinement of the obtained meshes to solve the contact problem

and to compute the instantaneous contact area for each position of the gear set along the gearing cycle. This approach is based on a previous work of the authors [36], that was tested with spur gears and supposed a first approximation to this final approach. In that work, the algorithm was simple, and not very efficient. However, in the algorithm presented here, important refinement and update steps were introduced to improve the ratio accuracy vs computational cost. The final algorithm has been tested with several cases to demonstrate its robustness and efficiency.

2. Prerequisites

The proposed approach has been formulated assuming the existence of an analytical model of the tooth contact surfaces. Consequently, the approach requires the existence of some algorithms (or functions) to compute the geometry of the teeth and certain parameters along the gearing cycle. Most of these functions involve simple geometric calculations while others have been extensively studied and can be found in the literature. For these reasons and for the aim of brevity, only a concise description of these required resources is presented.

R1. **Parametric functions defining the tooth contact surfaces.** The contact surface of a gear tooth is considered as defined as a parametric function ($\vec{S} : \Omega \in \mathbb{R}^2 \rightarrow \mathbb{R}^3$) that provides the 3D spatial coordinates (x, y, z) of a surface point from its parametric coordinates (u, v). Since there are two possible contact surfaces in one tooth (left and right sides, figure 1a), the contact side must be specified as well. Furthermore, the formulation of the parametric function is different from one gear type to other and, in the most general case, its implementation involves solving the gearing equation [37] that is usually formulated as a non-linear system of equations. The contact surface of each tooth side, computed with this function, is obtained in a position called *local reference position* that is described in section 3.

R2. **A discretization algorithm for the tooth contact surfaces.** Given the parametric function of the contact surface of a gear tooth and given a desired number of points in each one of the two intrinsic (profile and longitudinal) directions, defined by the client of the algorithm, this algorithm provides a structured triangle mesh of the surface. It includes not only the evaluation of the parametric function of the surface (resource R1) but also the computation of the surface limits according to the parameters of the gear (module, teeth number, addendum, dedendum, etc.). Each point of the resulting mesh stores both spatial (x, y, z) and parametric

(u, v) coordinates. Since the computation of the surface limits depends on the type of gear, this discretization algorithm must be specifically implemented for each gear type.

R3. **An algorithm to compute the minimum distance from a point to a triangle mesh in the 3D space.** This algorithm involves the computation of the minimum distance from a point to a triangle in the 3D space considering the internal area, the edges and the vertexes of the triangle (this means that the resulting distance is not necessarily measured along the normal vector of the triangle). After making the computation extensive to the whole mesh, the algorithm returns not only the distance but also the index of the triangle of minimum distance in the mesh and the point of minimum distance in that triangle.

R4. **An algorithm to compute the angle of rotation of a point around an axis to contact a triangle mesh in the 3D space.** Given a point that rotates around a specified axis with a defined rotating direction, this algorithm computes the minimum angle of rotation for the point to contact a triangle mesh. This work involves the computation of the intersection between a circumference (generated by the rotation of the point around the axis) and a triangle in the 3D space, that can be solved easily using geometric formulae. The intersection problem may have different solutions (two points, one point or no intersection) that must be treated properly. After making the computation extensive to the whole mesh, the algorithm returns not only the angle but also the index of the contacted triangle and the contact point in that triangle.

3. Local and gear set reference positions

The parametric function of a gear tooth surface (resource R1) defines the surface from its parameters in the local coordinate system of the gear (figure 1a). In this position, called *local reference position* (LRP), the reference tooth is defined vertical and centered on the Y axis and any point of the contact surfaces is defined by the position vector $\mathbf{r}_L(u, v)$.

To perform the contact analysis, it is necessary to have both gears in the position of the gear set. The transformation of the geometry of the gears from the local reference position to the gear set position (figure 1b) depends on the type of transmission. Without loss of generality, in this work a gear set with cylindrical gears (parallel axes) is assumed. Then, to obtain the pinion in the gear set position it is rotated to a initial angular position that will serve as reference and, after

that, it can be misaligned if desired for the analysis. Consequently, the position vector of a point of a pinion contact surface in the coordinate system associated to the gear set is obtained by means of the following transformations:

$$\mathbf{r}_{GS}^{(p)} = T_{GS-L}^{(p)} \cdot \mathbf{r}_{L}^{(p)} = \left[T_{m}^{(p)} \cdot Rt_z \left(\psi_0^{(p)} \right) \right] \cdot \mathbf{r}_{L}^{(p)} \tag{1}$$

where the superscript (p) indicates the pinion; $\psi_0^{(p)}$ is a any desired angle for the reference tooth that will serve as angular origin for the pinion; Rt_z is the rotation matrix around the Z axis corresponding to the specified angle and T_m is the transformation matrix that includes the desired misalignments (if any) for the pinion.

Similarly, to have the gear in the gear set position, the following transformations are applied:

$$\mathbf{r}_{GS}^{(g)} = T_{GS-L}^{(g)} \cdot \mathbf{r}_{L}^{(g)} = \left[T_{m}^{(g)} \cdot Tr(0, c, 0) \cdot Rt_z \left(\psi_0^{(g)} \right) \right] \cdot \mathbf{r}_{L}^{(g)} \tag{2}$$

where the superscript (g) indicates the gear; $\psi_0^{(g)}$ is the initial angle for the reference tooth that will serve as angular origin for the gear; Rt_z is the rotation matrix around the Z axis corresponding to the specified angle; Tr is the translation matrix along the specified vector, c is the center distance and T_m is the transformation matrix that includes the possible desired misalignment of the gear.

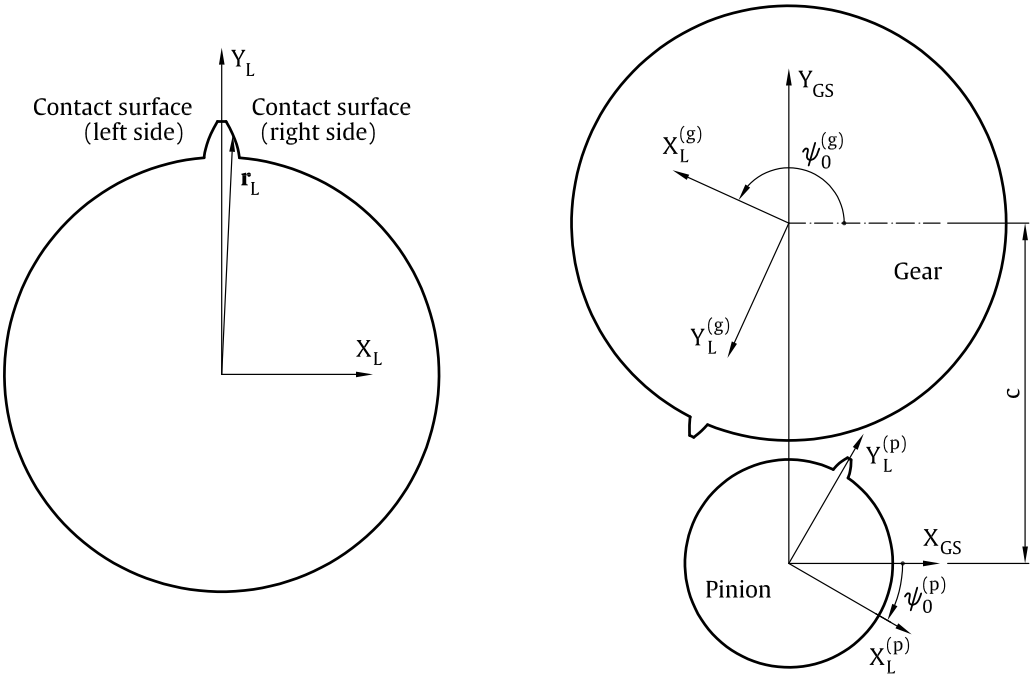
After applying the described transformations, the gears are in the initial position of the gear set that is called *gear set reference position* (GSRP).

The angles $\psi_0^{(p)}$ and $\psi_0^{(g)}$ can be assigned with any desired values that will be the angular origin of each gear, but values making the reference teeth of both gears to be close to each other ease the convergence of the algorithm. For this reason, values around of 0°for $\psi_0^{(p)}$ and 180°for $\psi_0^{(g)}$ are recommended.

4. Full definition of the gear set

The approach proposed in this work requires a fully defined gear set. The concept of *fully defined gear set* used here includes the following information:

- The geometric definition of the pinion, that includes all the parameters needed to define the geometry of the pinion in its LRP. It contains the information required by the resources R1 and R2 to get points and to get a mesh (discretization) from the tooth contact surface corresponding to the actual contact side of the pinion tooth.



Gear Local Reference Position (LRP)
(a)

Gear Set Reference Position (GSRP)
(b)

Figure 1: *Local* and *gear set* reference positions.

- The geometric definition of the gear, that includes all the parameters needed to define the geometry of the gear in its LRP. It contains the information required by the resources R1 and R2 to get points and to get a mesh from the tooth contact surface corresponding to the actual contact side of the gear tooth.

- The direction of rotation of the gear set, that is specified by the direction of rotation of the pinion. The tooth contact side (left or right, figure 1a) of each gear depends on the type of each gear (internal or external) and also on this direction of rotation.

- The transformation matrix $T_{GS-L}^{(p)}$ to transform the pinion from its LRP to the GSRP, as described above.

- The transformation matrix $T_{GS-L}^{(g)}$ to transform the gear from its LRP to the GSRP, as described above.

5. Tooth contact analysis algorithm (TCA algorithm)

The TCA algorithm presented here computes the contact of a reference teeth pair (one gear tooth contacting one pinion tooth) at different positions along the gearing cycle. Since the angular position of the gear set is specified by the angular position of the pinion, the algorithm varies the angular position of the pinion tooth along the range defined by the pinion step angle and, for each position, it solves the contact problem (algorithm CP) and computes the instantaneous contact area (algorithm ICA). The number of intermediate positions to compute along the step angle and the starting position of the pinion in this range are input parameters for the TCA algorithm.

After solving the contact problem for each position of the pinion, the transmission error is computed as the difference of the actual angular position and the theoretical angular position of the gear, taking the starting position as reference. Finally, the contact pattern is presented as the superposition of the instantaneous contact areas associated to the computed positions of the gear set.

6. Algorithm to solve the contact problem (algorithm CP)

Given a fully defined gear set at the GSRP and having rotated the pinion up to a specific position of analysis, this algorithm has the objective of computing the angle that is necessary to rotate the surface of the gear tooth to contact the surface of the pinion tooth.

The algorithm is a geometric approach designed to be independent from the transmission type and to be able to solve the contact problem for any relative position of the contacting surfaces. It works with discrete contact surfaces (triangle meshes) and the contact of the gear mesh with the pinion mesh is computed with the *least rotation angle* (LRA) strategy introduced by Simon [38] and extended by Lin [31]. According to this method, the necessary angle of rotation for the gear mesh to contact the pinion mesh can be obtained computing, for each node of the gear mesh, the angle of rotation to contact the pinion mesh by means of resource R4. Then, the node with the least rotation angle (LRA) is the point of the gear tooth that will contact first the pinion tooth. However, although this method is robust, it may introduce an error caused by the discretization. Figure 2a shows a two dimensional sample of this error. The problem can be solved in two ways: (i) refining both meshes up to a degree where the error is negligible or (ii) swapping the moving and static roles between pinion and gear meshes (figure 2b) to detect what node of which mesh

contacts first the triangles of the other mesh. In the algorithm presented here, the second solution has been implemented.

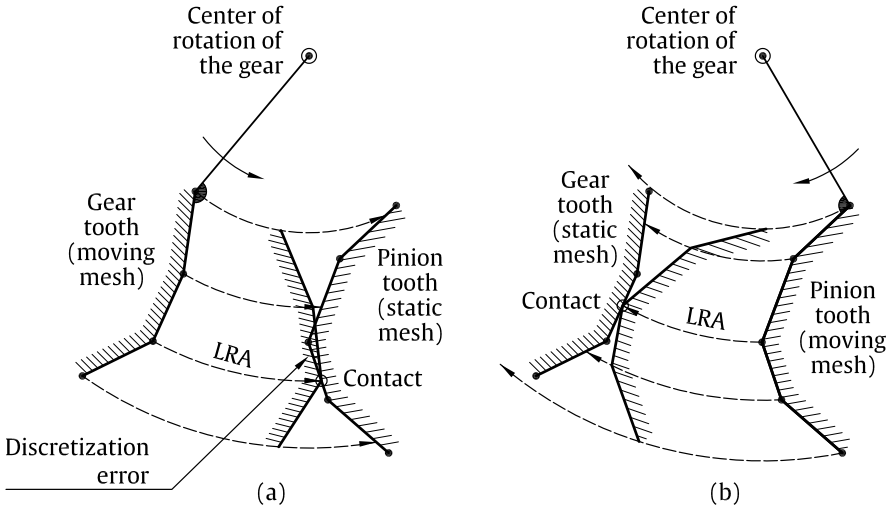


Figure 2: Discretization error (a) and solution (b) during contact detection.

The flowchart of the algorithm CP has 11 steps that are shown in figure 3. It requires a fully defined gear set and also the following input parameters:

- *The angular position of the pinion*(ψ_p). The contact analysis is made for a specific angular position of the gear set that is specified by the angular position of the pinion. This angular position is measured from the GSRP of the pinion.

- *The angle-of-rotation-to-the-contact tolerance for refinement* ($\Delta\theta$). This value delimits the area that it is going to be refined in each loop iteration. The meaning of this parameter is explained below, in the step 5 of the algorithm.

- *The triangle size tolerance for refinement*(S_{tol}). This value is used by the algorithm to detect the achievement of the refinement objective and to finish the loop. Thus, the refinement loop is continued until the size of all triangles refined in the last loop iteration is lower than this value. The size of a triangle is defined as the length of its longest edge.

The algorithm starts (**step 1**) obtaining a basic discretization of both pinion and gear contact surfaces in the form of triangle meshes by using resource R2. These meshes are structured, relatively coarse and they are obtained at the LRP of each gear. The nodes of each mesh are on the contact surface that generated the mesh and the limits of the mesh are a discretization of the limits of

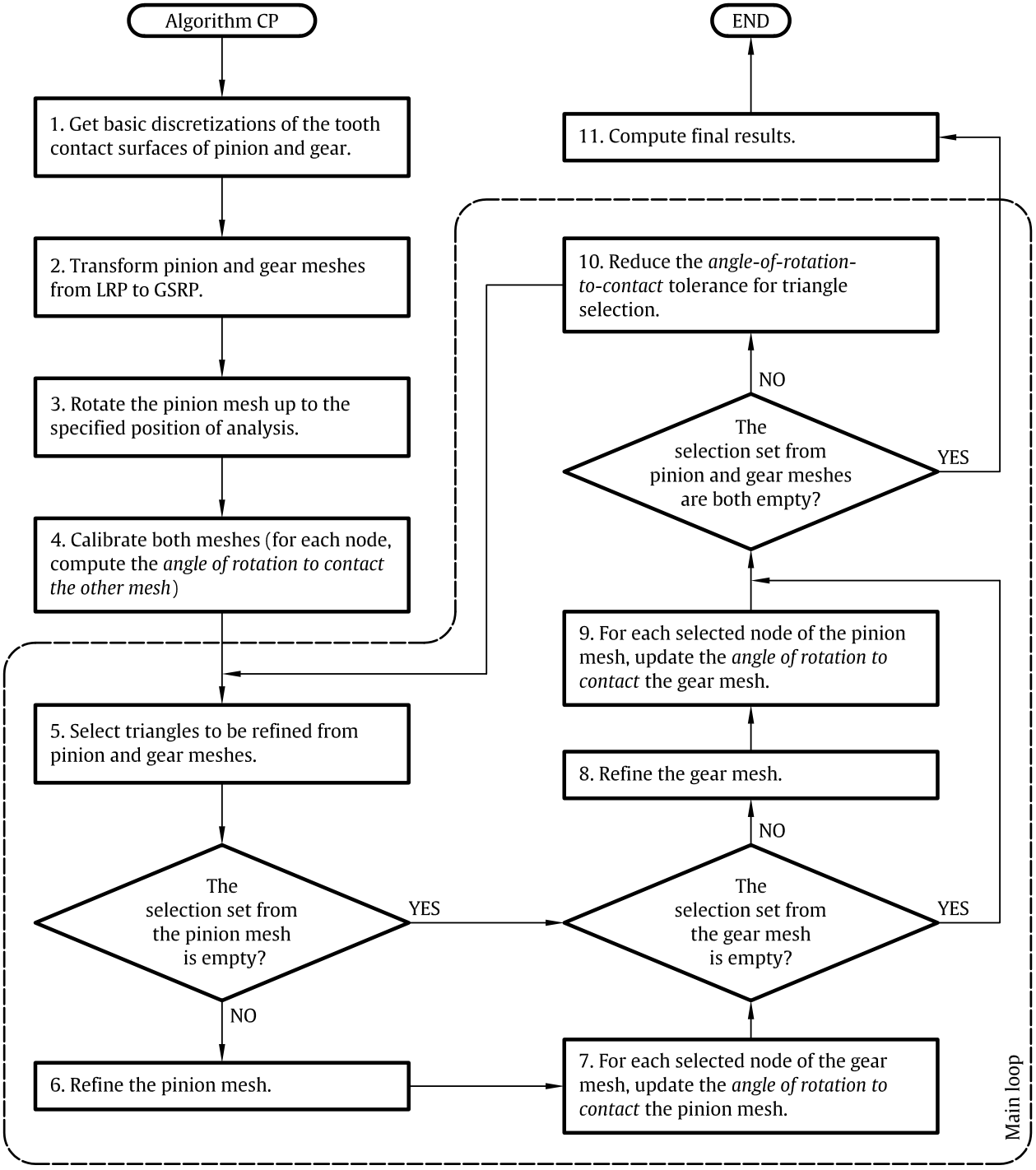


Figure 3: Algorithm to solve the contact problem (CP)

the contact surface. Each mesh node stores its 3D spatial coordinates (x, y, z) and also its surface parametric coordinates (u, v). The information of each mesh is stored using a half-edge data structure [39] that facilitates the traverse operations required during the refinement, increasing the

efficiency of the algorithm.

In the **step 2**, pinion and gear meshes are moved from their LRP to the GSRP by means of the transformation matrices $T^{(p)}_{GS-L}$ and $T^{(g)}_{GS-L}$, respectively. Then (**step 3**), the pinion is rotated around its own axis up to the position of analysis according to the specified value of the angular position of the pinion (ψ_p). In both gears, the successive geometric transformations applied from the LRP to the current position are stored in two respective transformation matrices ($T^{(p)}_{acc}$ and $T^{(g)}_{acc}$) for later use.

Both pinion and gear meshes are, then, calibrated (**step 4**). The calibration of the gear mesh implies that for each node i of that mesh, the angle of rotation to contact the pinion mesh (θ_i) and the index of the contacted triangle in the pinion mesh (t_i) are computed by using resource R3. During the process, the node of the gear mesh with minimum value of θ_i (i.e., the LRA of the current mesh) is saved. In the end, that node corresponds to the node that contacts the pinion mesh first. On the other hand, the calibration of the pinion mesh implies the same operation, but with respect to the gear mesh.

Step 5 consists in selecting the triangles that are going to be refined. Since the algorithm must refine the area around the contact point to solve the contact problem accurately, the objective of this step is to select the triangles that are close to the current contact point of the meshes. For this task, a new parameter called *limit value of the angle-of-rotation-to-the-contact* (θ_{lim}) is used. To explain this parameter, let us consider the two dimensional problem of figure 4a. In that problem there is a pinion tooth at the position of analysis and a gear tooth at the GSRP. The required angle of rotation for the gear to contact the pinion is the unknown of the problem. A discretization of the contact surfaces has been obtained, showing the nodes of the gear mesh (that is a polygon in this 2D example) in figure 4a. The gear mesh has been calibrated and figure 4a shows the angle of rotation to contact the pinion (θ_i) for nodes 2 to 8 (nodes 0 and 1 do not contact the pinion tooth for that position). The lowest value of θ_i ($i = 2, ..., 8$) is the LRA and, in this case, it corresponds to node 5 ($\theta_{min} = \theta_5$) meaning that node 5 is the first node contacting the pinion mesh, as it can be observed after rotating the gear mesh (figure 4b). The elements that are close to the contact are selected by using the *angle-of-rotation-to-the-contact tolerance for refinement* ($\Delta\theta$), that is an input parameter of the algorithm. The limit for refinement is computed as $\theta_{lim} = \theta_{min} + \Delta\theta$, so the nodes with associated value of *angle-of-rotation-to-the-contact* under this limit (nodes i where $\theta_i < \theta_{lim}$) are the nodes that are close to contact the pinion mesh. For the value of $\Delta\theta$ used in

this example, only nodes 4, 5 and 6 are under this limit, so they are the closest nodes to the pinion mesh as can be observed in figure 4b. Consequently, all elements of the gear mesh including at least one of these nodes would be selected for refinement.

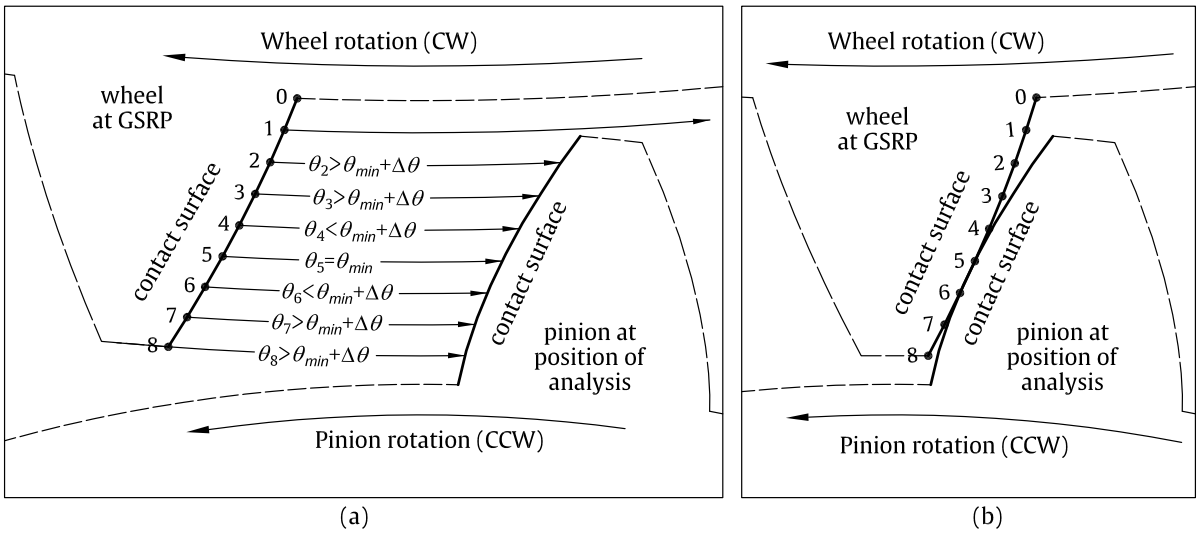


Figure 4: Contact analysis refinement criterion in a 2D case.

In the 3D problem, the algorithm works the same way except for the fact that the elements are triangles instead of segments. Furthermore, in addition to the *angle-of-rotation-to-the-contact* criterion, the triangle size criterion is also considered to select the triangles. In the end, only triangles with size higher than the triangle size tolerance and having at least one node with associated value of *angle-of-rotation-to-the-contact* under the calculated limit (θ_{lim}) are selected. And this is done separately for pinion and gear.

After performing step 5, if the triangle selection set of the pinion mesh is empty, this would mean that the refinement objective for the pinion mesh would have been reached (the pinion would have been refined up to the required element size), so the refinement work for this mesh would finish. In that case, steps 6 and 7 would be skipped.

Step 6 has the task of refining the selected triangles from the pinion mesh. Thus, each selected triangle is split adding a new node at the midpoint of its longest edge (this also splits the adjacent triangle). Additionally, to prevent the degeneration of the mesh during the refinement, the algorithm takes into account that none of the selected triangles is split more than once in the same iteration of the main loop. The spatial and parametric coordinates of the new node are linearly interpolated from the end nodes of the edge that was split. Consequently, the new node is not

(necessarily) on the tooth surface. Thus, to make the refinement geometrically adaptive, the spatial coordinates (x, y, z) of the node are, then, updated by means of the parametric function of the contact surface (resource R1) using the interpolated parametric coordinates (u, v). Then, since the parametric function returns the spatial coordinates in the LRP, the transformation matrix stored in step 3 ($T_{acc}^{(p)}$) is used to transform the new nodes to the current position of the pinion. Finally, for each new node the resource R4 is used to compute the angle of rotation to contact the gear mesh and the index of the contacted triangle, finishing the task of updating the data of the new nodes.

Since the pinion mesh has been modified in the last step, the values of the angle of rotation to contact the pinion mesh associated to the nodes of the gear mesh are outdated. But, updating the information of all nodes is unnecessary and inefficient because only nodes close to the contact point are relevant for the problem. Then, since the selection performed in step 5 contains the elements and nodes that are close to the contact, **step 7** updates the angle of rotation to contact the pinion mesh associated to the nodes of the current triangle selection of the gear mesh. Before the update, each one of these nodes is storing the index of the previous contact triangle in the pinion mesh. Then, a simple iterative algorithm was implemented to traverse from the previous contact triangle to its neighboring triangles (and so on) to find the new contact triangle. With the use of a half-edge data structure, the algorithm is simple and very efficient.

After refining the pinion, the algorithm does the same with the gear. It starts checking if the triangle selection set of the gear mesh is empty. If so, the refinement objective for the gear would have been reached and no more refinement would be required, so steps 8 and 9 would be skipped.

Otherwise, **step 8** performs the refinement of the gear mesh the same way that step 6 did for the pinion. And, after refining the gear mesh, **step 9** updates the values associated to the nodes of pinion mesh the same way step 7 did for the nodes of the gear mesh.

After step 9, the algorithm checks if the triangle selection sets from pinion and gear meshes are both empty. If so, the refinement objective would have been reached for both meshes and the algorithm would jump to step 11 to compute the final results. Otherwise, it would continue with step 10.

In the **step 10**, the *angle-of-rotation-to-the-contact tolerance for refinement* ($\Delta\theta$) is reduced. This implies that the area of refinement (that is, the group of triangles and nodes that are close to contact the other mesh) is reduced progressively in each refinement iteration. The reduction factor

that has been used in this work is 2, therefore:

$$\Delta\theta_{iter+1} = \frac{\Delta\theta_{iter}}{2} \tag{3}$$

After this step, the algorithm continues with step 5 initiating a new loop iteration.

Step 11 finishes the algorithm computing the results: the LRA and the contact points on both meshes.

7. Algorithm to compute the instantaneous contact area (algorithm ICA)

The objective of this algorithm is to compute the instantaneous contact area for a specific position of the gear set in which the pinion and gear tooth surfaces are in rigid contact. The instantaneous contact area of a gear is the locus of points of the contact surface whose distance to the contact surface of the opposite tooth is lower than the virtual marking compound thickness (δ).

The proposed algorithm consists in 10 steps (figure 5). It requires a fully defined gear set and also the following input parameters:

- *The angular position of the pinion* (ψ_p). It defines the angular position of the gear set in which the instantaneous contact area is going to be computed. This angle is measured from the GSRP of the pinion.

- *The angular position of the gear in contact with the pinion* (ψ_g). It is the angle that the gear must be rotated around its own axis, measured from its GSRP, for the contact surface of its reference tooth to be in rigid contact with the contact surface of the pinion reference tooth. This value is one of the results provided by the algorithm CP after solving the contact problem.

- *The virtual marking compound thickness* (δ). This value defines the instantaneous contact area that is going to be computed, as explained previously.

- *The maximum and minimum distance limits for refinement* (d_{max}, d_{min}). These values delimit the area that it is going to be refined in each loop iteration. They enclose the value of δ, so it is necessary that $d_{min} < \delta < d_{max}$. The influence of these parameters in the algorithm is explained below, in step 5.

- *The triangle size tolerance for refinement*(S_{tol}). This value represents the objective of refinement and it is used by the algorithm to finish the loop. Therefore, the refinement loop is continued until the size of all triangles refined in the last loop iteration is lower than this value.

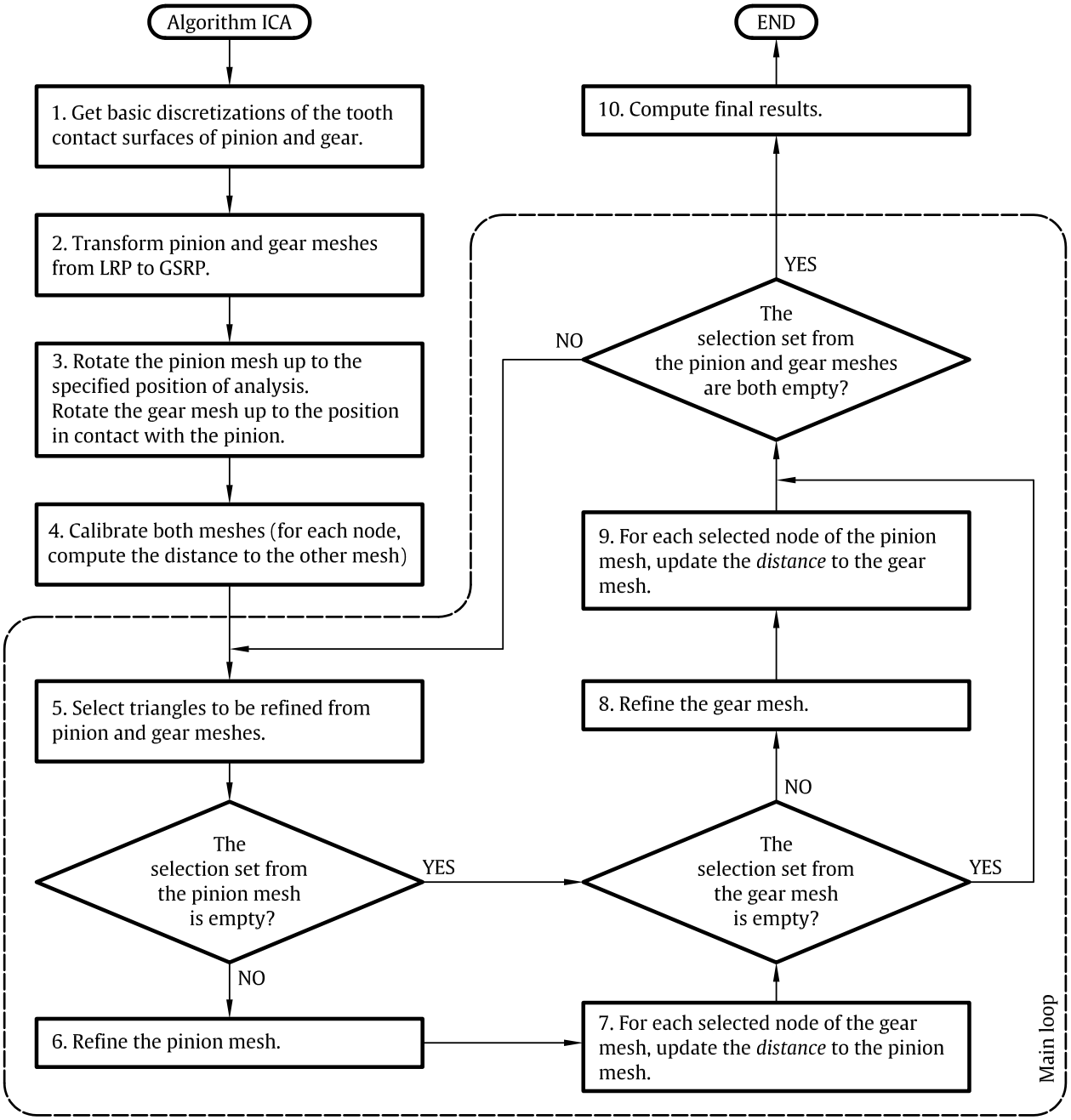


Figure 5: Algorithm to compute the instantaneous contact area (ICA).

Steps 1, 2 and **3** deal with obtaining meshes from both pinion and gear contact surfaces (using resource R2) and positioning them in contact at the position specified by the input parameters. The successive geometric transformations applied to pinion and gear are stored in two respective transformation matrices ($T^{(p)}_{GS-L}$ and $T^{(g)}_{GS-L}$) for later use.

Both pinion and gear meshes are, then, calibrated (**step 4**). The calibration of the gear mesh implies that, for each node i of that mesh, the distance to the pinion mesh is computed by using resource R3. After that, each node i stores the distance to the pinion mesh (d_i) and the index of the triangle of minimum distance in the pinion mesh (t_i) as additional data. Then, the calibration of the pinion mesh implies the same but with respect to the gear mesh.

Considering that there is a distance associated to each node in both meshes, the instantaneous contact area on each mesh is the level curve associated to a distance equal to the virtual marking compound thickness (δ). But, although this level curve could be easily extracted by linear interpolation using the current distance associated to the nodes, if the mesh is coarse, the obtained level curve would also be. So, to obtain accurate and well formed instantaneous contact areas, the algorithm has to perform a mesh refinement. Thus, **step 5** consists in selecting the triangles that are going to be refined. But, for the refinement to be efficient, only triangles around the level curve of δ should be selected for refinement. For this reason, the area of refinement is delimited by the input parameters d_{min} and d_{max} having values around δ. Considering also that only triangles with size higher than the requested size must be refined, the final criterion for selecting triangles from each mesh is: all triangles with size higher than S_{tol} and having at least one node i where $d_{min} \leq d_i \leq d_{max}$ are selected.

If the triangle selection set of the pinion mesh is empty, this would mean that the refinement objective would have been reached for the pinion and, then, steps 6 and 7 would be skipped. If the selection set is not empty, **step 6** would refine the pinion mesh. The approach is similar to step 6 of the algorithm CP but computing *distances* to the gear mesh instead of *angles of rotation to contact* the gear mesh. Thus, after splitting the triangles, for each new node i of the pinion mesh, the resource R3 is used to compute the distance to the gear mesh (d_i), the index of the gear triangle (t_i) where the distance is minimum and the point of minimum distance in that triangle (P_{g_i}), finishing the update of the information associated to the new nodes.

Since the pinion mesh has been modified, the value of distance to the pinion mesh associated to the nodes of the gear mesh are outdated. Therefore, the **step 7** updates this value for all nodes

of the selected triangles in the gear mesh. Each one of these nodes of the gear mesh is storing the index of the previous pinion triangle (t_i) where the distance is minimum. Thus, a simple iterative algorithm was implemented to traverse from the old triangle of minimum distance to its neighboring triangles (and so on) to find the new triangle of minimum distance. The approach updates the three values associated to the node: the distance to the pinion mesh (d_i), the index of the pinion triangle (t_i) where the distance is minimum and the point of minimum distance in that triangle (P_{p_i}).

Then, if the triangle selection set of the gear mesh is empty, this would mean that the gear mesh would have already reached the refinement objective and steps 8 and 9 would be skipped. Otherwise, **step 8** would refine the gear mesh as step 6 did for the pinion and **step 9** would update the pinion mesh as step 7 did for the gear.

After that, a new check is performed. If both selection sets are empty, the refinement objective would have been reached for both meshes and the algorithm would jump to step 10. Otherwise, the algorithm would jump to step 5 to perform a new loop iteration.

Finally, the **step 10** computes the instantaneous contact area, that is the level curve associated to the virtual marking compound thickness (δ). This curve is obtained from the distance value of each node by linear interpolation, applied only to triangles having nodes with distance values over and under δ. The half-edge data structure used to store the mesh information makes very easy to traverse from one triangle to their adjacent triangles, so the approach to construct the level curve segment by segment is simple and efficient.

8. Performance test

Taking into account that the most difficult cases for any TCA algorithm are those with line contact and edge contact (as justified in section 1), two of these cases have been selected to test the proposed approach. Besides that, a third test case has been added to demonstrate the robustness of the method with gears with micro-geometry modifications having a non-edge point contact in both gears.

The first case (case 1 in table 1) corresponds to a standard spur gear set (figure 6a) where pinion and gear have the same face width and are perfectly aligned. According to this definition, this case will provide a full line contact along the face width. In case 2 (table 1) a standard helical gear set (figure 6b) is used. In this case, the pinion has a face width higher than the gear (60 mm

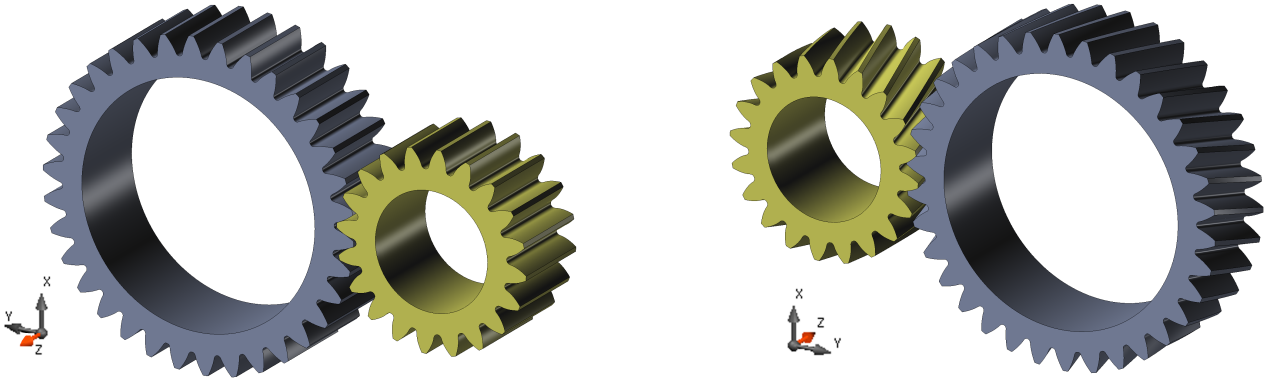
	Case 1		Case 2	
	Pinion	Gear	Pinion	Gear
Transm. Type	Spur (standard)		Helical (standard)	
Module	5 mm		5 mm	
Pressure angle	25 deg		25 deg	
Helix angle	-		15 deg	
Helix hand	-	-	left	right
Number of teeth	20	34	20	34
Step angle	18°	10.588°	18°	10.588°
Pitch diameter	100 mm	170 mm	103.528 mm	175.997 mm
Addendum	5 mm	5 mm	5 mm	5 mm
Dedendum	6.25 mm	6.25 mm	6.25 mm	6.25 mm
Face width	50 mm	50 mm	60 mm	50 mm
Center distance	Standard		Standard + 0.1 mm	
Alignment status	Aligned		Misaligned	
Rotation of the gears for misalignment	(none)	(none)	(none)	+0.02°around the Y axis

Table 1: Transmission data for cases 1 and 2.

vs. 50 mm), being both gears centered in Z direction. The center distance has been increased 0.1 mm, and the gear has been misaligned respect to the pinion with a rotation angle of +0.02°around the Y axis. This way, the face width difference will provide a truncated contact pattern in the pinion and the misalignment will provide an edge contact in the gear.

In both cases, to compute the TCA along the gearing cycle, 9 positions per step angle have been considered. For each position, the algorithm CP was used to solve the contact problem and, after that, the algorithm ICA was used to compute the instantaneous contact area. Both algorithms started obtaining a basic mesh of the tooth contact surfaces with 8 elements in profile direction and 18 elements in longitudinal direction, providing an element size (length of the diagonal) around 4 mm. On the other hand, the values used for the required input parameters of both algorithms are shown in tables 2 and 3.

Figure 7 shows the results of the algorithm CP corresponding to case 1 for an intermediate



(a) Gear set for test case 1 (b) Gear set for test case 2

Figure 6: Gear sets used in the test cases.

Angle-of rotation-to-contact tolerance for refinement	$\Delta\theta$	0.1°
Triangle size tolerance for refinement	S_{tol}	0.2 mm

Table 2: Input parameters of algorithm CP.

position (5 of 9) of the gearing cycle. Figure 7a shows the starting basic meshes, where the pinion mesh is at the position of computation and the gear mesh is at the GSRP. After executing the refinement loop, the resulting meshes and the obtained least rotation angle (LRA) are shown in figure 7b. It can be observed how the area around the theoretical contact line has been refined progressively to find the contact point (figure 7c). Finally, figure 7d shows both meshes in rigid contact after rotating the gear.

Similarly, figure 8 shows the intermediate results of the algorithm CP corresponding to case 2. Figure 8a shows the initial basic meshes and figure 8b shows the refinement performed to solve the contact problem and to find the contact points. It can be observed how the refinement process converged to the expected contact point according to the misalignment. That point is on the edge of the gear but it is not on the edge of the pinion because of its higher face width, as expected. The regularity of the refinement and the resulting contact point on the pinion can be observed in

Virtual marking compound thickness	δ	0.0065 mm
Initial minimum distance limit for refinement	d_{min}	δ−0.005 mm
Initial maximum distance limit for refinement	d_{max}	δ+0.005 mm
Triangle size tolerance for refinement	S_{tol}	0.2 mm

Table 3: Input parameters of algorithm ICA.

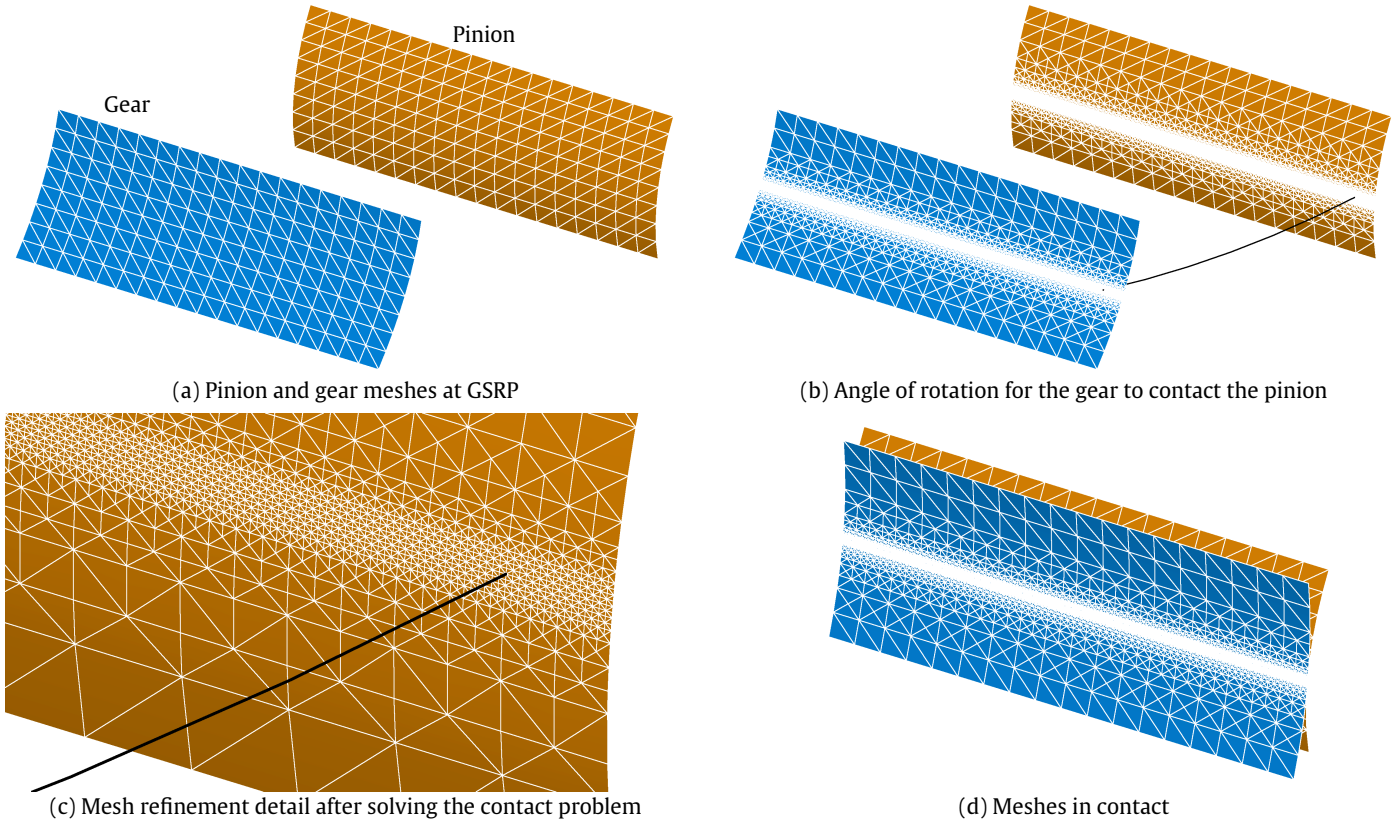
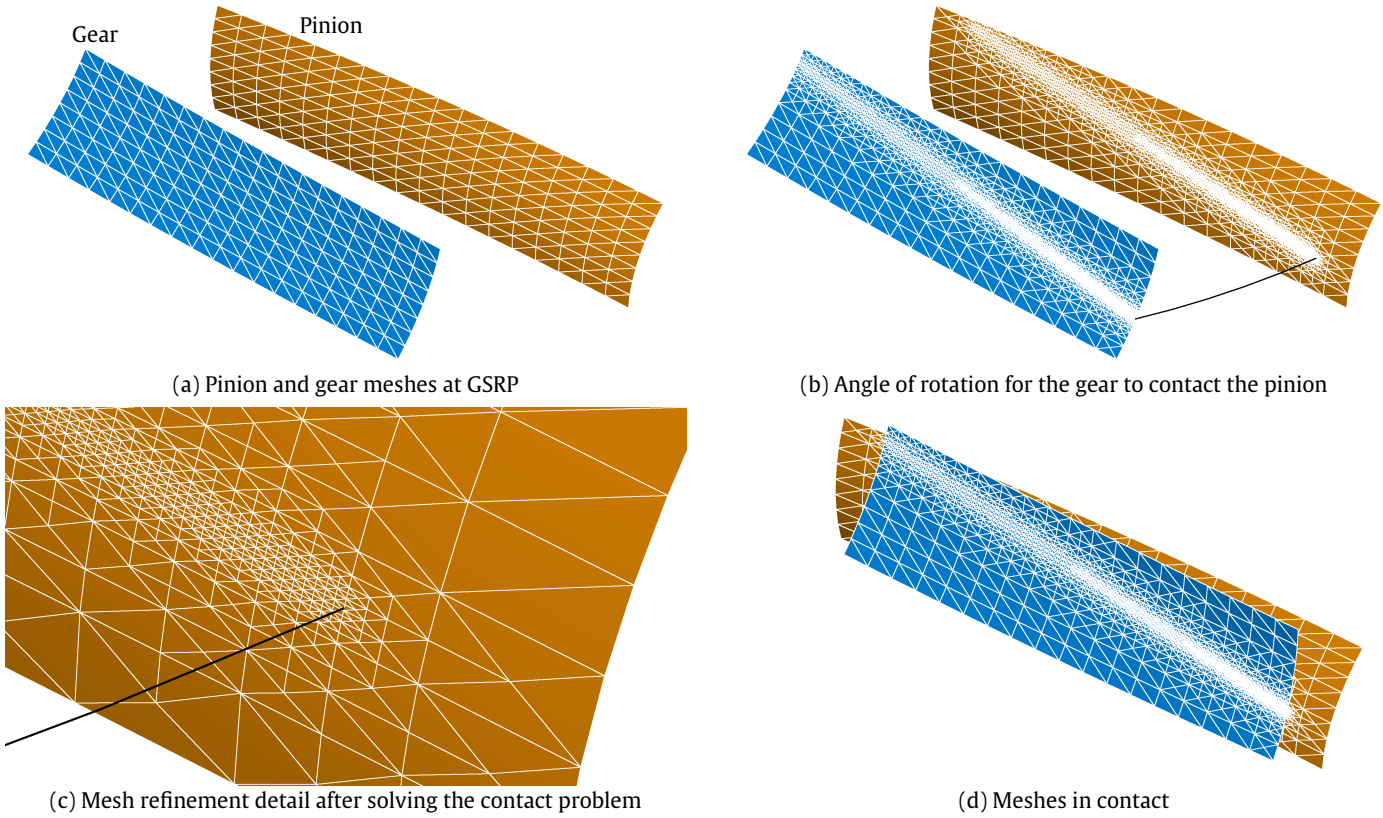


Figure 7: Results of the algorithm CP for case 1.

figure 8c and, finally, figure 8d shows both meshes in perfect rigid contact.

After solving the contact problem, the algorithm ICA is applied to compute the instantaneous contact area for the considered position of the gear set. In case 1, this algorithm starts obtaining the same initial meshes (figure 7a). Then, the gear mesh is rotated to be in contact with the pinion according to the result of the algorithm CP. After the refinement loop, the border of the instantaneous contact area is obtained (figure 9a). Figure 9b shows a detail of the adaptive refinement performed by the algorithm around the border of the instantaneous contact area. The obtained results show that the contact area agrees with what was expected for this line-type contact problem.

Similarly, in case 2 the algorithm ICA starts with the same initial meshes than algorithm CP (figure 8a), transforming the gear to be in contact with the pinion at the position specified by the pinion angle. After the refinement loop, the instantaneous contact area is computed (figure 10a) and it is possible to observe that the obtained area agrees with what was expected for the imposed misalignment. Finally, figure 10b shows the quality of the adaptive refinement process and a detail of the interpolated border of the contact area.



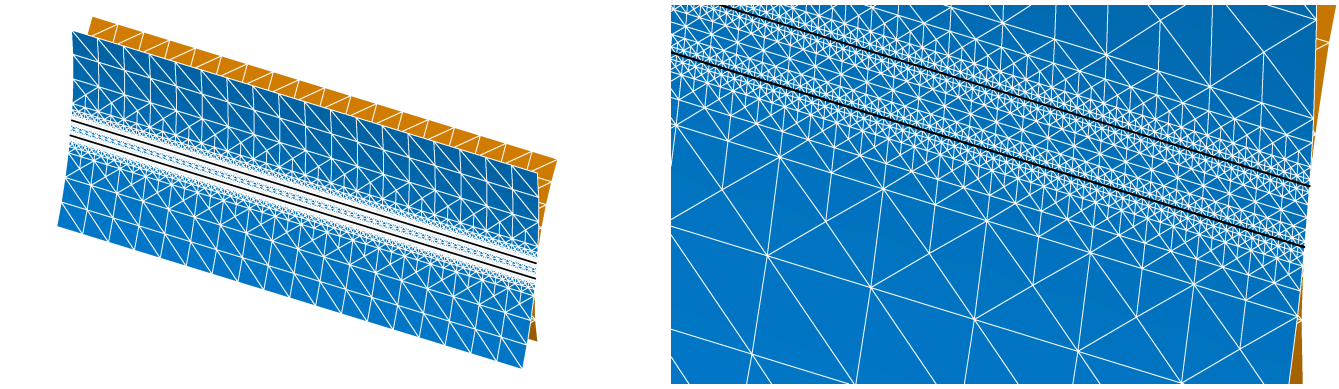
(a) Pinion and gear meshes at GSRP

(b) Angle of rotation for the gear to contact the pinion

(c) Mesh refinement detail after solving the contact problem

(d) Meshes in contact

Figure 8: Results of the algorithm CP for case 2.

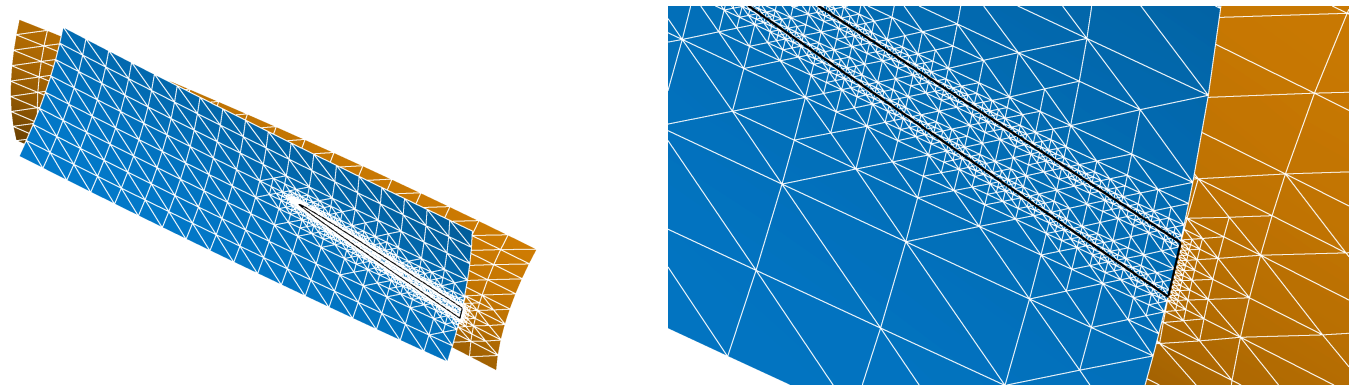


(a) Meshes after computing the instantaneous contact area

(b) Mesh detail after computing the inst. contact area

Figure 9: Results of the algorithm ICA for case 1.

The TCA algorithm repeats the computation of the contact and the computation of the instantaneous contact area for the different positions along the gearing cycle. As a result, the contact pattern and the function of transmission errors are obtained. In case 1, figure 11a shows the obtained contact pattern for the pinion. It can be observed that the instantaneous contact areas

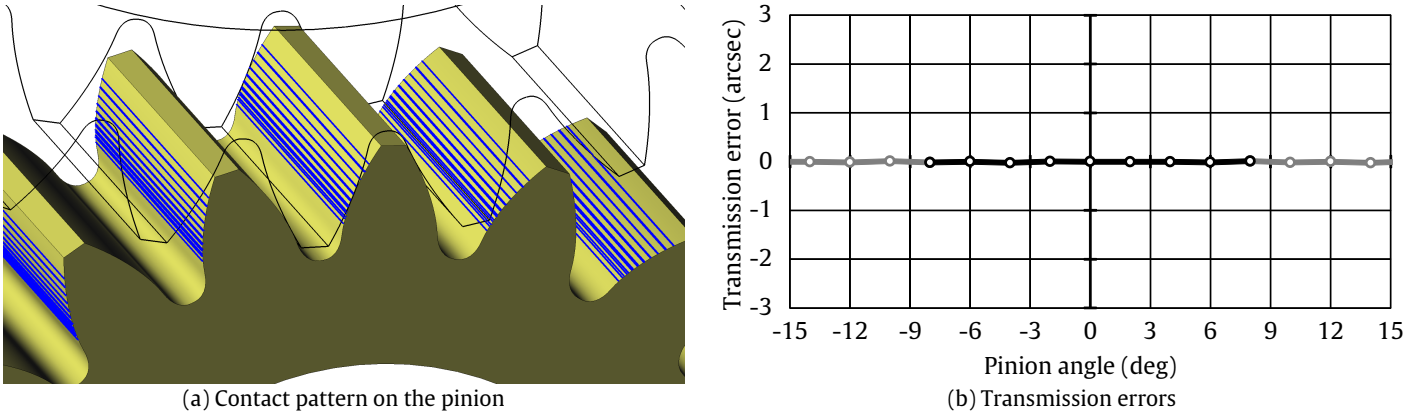


(a) Meshes after computing the instantaneous contact area

(b) Mesh detail after computing the inst. contact area

Figure 10: Results of the algorithm ICA for case 2.

of the 9 computed positions fill the face width of the pinion, as expected. On the other hand, case 1 corresponds to an involute spur gear transmission with perfectly aligned gears; so the expected transmission error is zero for all positions of the gear set. Figure 11b shows the accuracy of the obtained function of transmission errors, being the numerical error lower than 0.1 arc sec ($2.78 \cdot 10^{-5}$ degrees). The figure also shows the almost nonexistent numerical noise associated to the computation of the function of transmission error for the specified level of refinement.



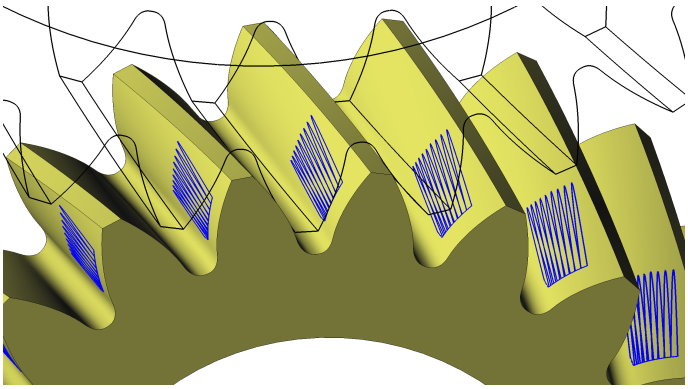
(a) Contact pattern on the pinion

(b) Transmission errors

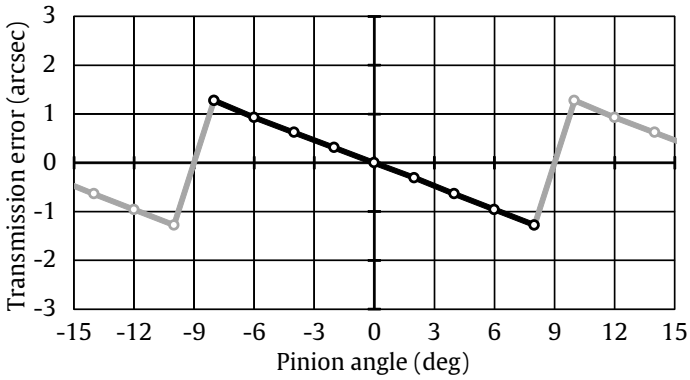
Figure 11: Results of the TCA algorithm for case 1.

In case 2, figure 12a shows the contact pattern on the pinion, formed by the 9 truncated ellipses associated to the 9 considered positions of the gear set and figure 12a shows the obtained function of transmission errors. In this case, a well formed sawtooth function with an amplitude of 1.2 arcsec was obtained, agreeing with the misalignment of the gears.

In test case 3, a transmission similar to case 1 has been used, but the gears have been misaligned:



(a) Contact pattern on the pinion



(b) Transmission errors

Figure 12: Results of the TCA algorithm for case 2.

the center distance has been increased 0.1 mm (by displacing the gear in Y direction) and the gear has been rotated 0.05°around axis Y (figure 6a). To improve the tolerance of the transmission to the imposed misalignment, the tooth surfaces of the pinion have been modified by a grinding wheel (with a pitch radius of 50 mm) while the gear is kept as a standard involute spur gear.

In first place, to avoid discontinuities in the function of transmission errors, tip and bottom reliefs have been imposed in the tooth profile of the pinion. Thus, the profile of the grinding wheel (figure 13a) has been defined by a blade of a rack-cutter where both left and right profiles have been designed parabolic (equation 4) in their respective coordinate systems. Then, using this blade, the profile of the grinding wheel has been computed by using the gearing equation.

$$y = \frac{0.005}{mm} x^2 \tag{4}$$

In second place, to limit the edge contact in the gear set, a small crowning has been applied to the pinion. Thus, the grinding wheel has been moved (in the plane YZ) through a parabolic path instead of a straight line (figure 13b). This path is defined by equation 5 in the local coordinate system of the pinion, where D_p is the pitch diameter of the pinion and D_{gw} is the pitch diameter of the grinding wheel.

$$y = \frac{D_p + D_{gw}}{2} - \frac{0.0001}{mm} z^2 \tag{5}$$

After applying the TCA algorithm to case 3 with the same parameters used in cases 1 and 2, figure 14a shows the resulting contact pattern for the 9 positions along the gearing cycle and 14b the obtained transmission errors. It can be observed the quality of the shape of the contact areas

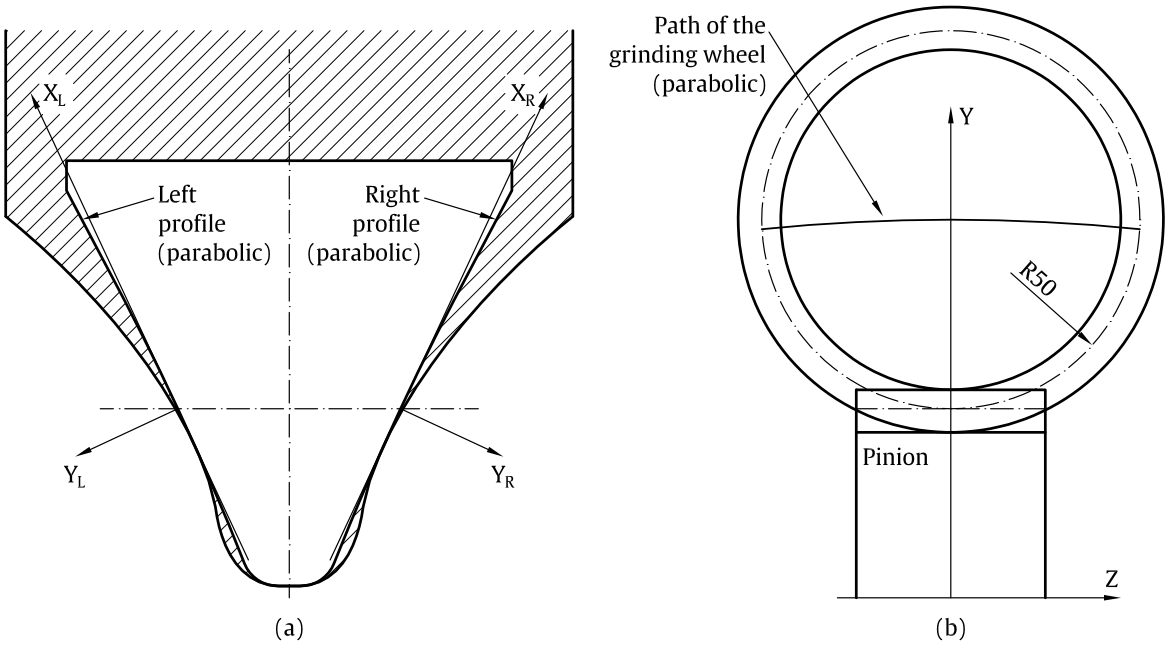


Figure 13: Definition of the grinding wheel for case 3.

and also the low numerical noise of the parabolic shape of the transmission error function.

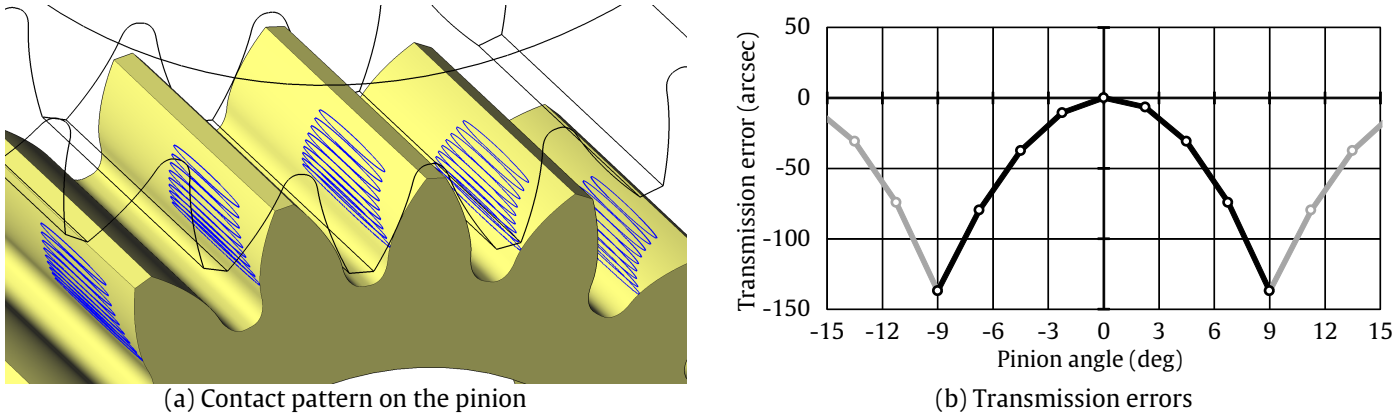


Figure 14: Results of the TCA algorithm for case 3.

9. Comparison with other approaches

The proposed algorithm has been validated with the TCA approach proposed by Sheveleva [32]. This algorithm also tackles the TCA problem by solving the contact problem and by computing the instantaneous contact area for different positions of the gear set along the gearing cycle. Then, the contact pattern is obtained by overlapping the different instantaneous contact areas.

To solve the contact problem, the algorithm of Sheveleva uses a grid of points in each contact surface and another grid in a plane tangent to the gear. Then, it computes the distances between the points of the pinion grid to the contact surface of the gear grid by bilinear interpolation using the distance field of the grid in the tangent plane. The distances are projected in the velocity vector of each point of the gear to estimate the least angle that is necessary to rotate the gear for contacting the pinion. However, the obtained angle is not exact because of the approximations, so an iterative process is necessary to obtain an accurate value of this angle.

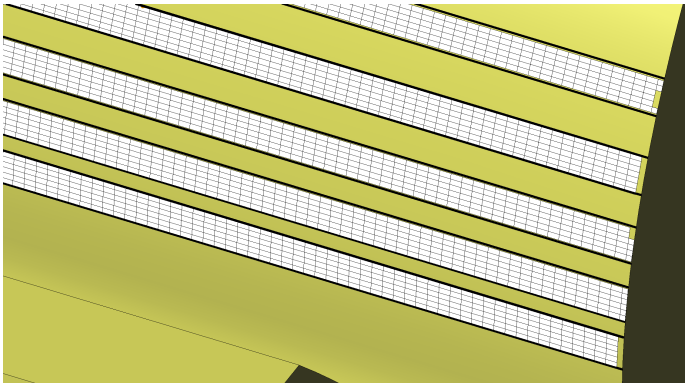
Once the contact problem has been solved, the instantaneous contact areas in both gears are obtained from the field of distances associated to each point of the surface meshes. Each point has an associated area of influence (rectangle), if the distance of the point to the other surface mesh is lower than $\delta cos(\theta)$ (where δ is the virtual marking compound thickness, and θ the angle between the normal of the pinion surface at that point and the normal vector of the gear tangent plane) , the rectangle is considered as part of the contact area, otherwise the rectangle is discarded. Repeating this check for all points, an discrete estimation of the instantaneous contact area is obtained.

The main differences of the algorithm of Sheveleva with the proposed approach is that the first one linearizes the rotating movement of gear to compute the angle of rotation to contact the pinion and that it does not refine the mesh to solve the contact problem and to compute the instantaneous contact area.

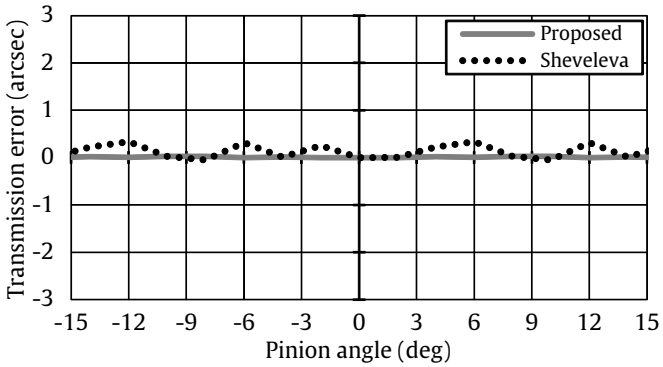
For the validation of the results of cases 1, 2 and 3, the algorithm of Sheveleva has been executed with a grid of 161 points in longitudinal direction and 81 points in profile direction. Using the same geometric model of the tooth surfaces, the obtained function of transmission errors and contact pattern have been compared to the results obtained in the performance test (section 8). The results of this comparison are presented in figure 15.

Regarding the contact patterns (figures 15a, 15c and 15e), only positions 1, 3, 5, 7 and 9 (of 9) have been shown to avoid overlapping of the contact areas. The contact areas provided by the algorithm of Sheveleva are displayed as a white grid of rectangles (corresponding to the grid of points used by that algorithm) and the contact areas obtained by the proposed algorithm are displayed as a black border (same border as in the performance test). In the case of transmission errors (figures 15b, 15d and 15f), the values of the 9 positions are included in the graphs.

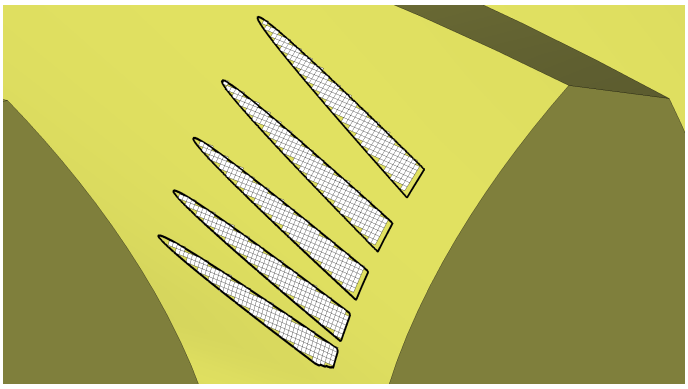
In these figures it is possible to observe the high level of coincidence in the instantaneous contact areas for the 3 cases. Furthermore, the comparison reveals the higher ability of the adaptive



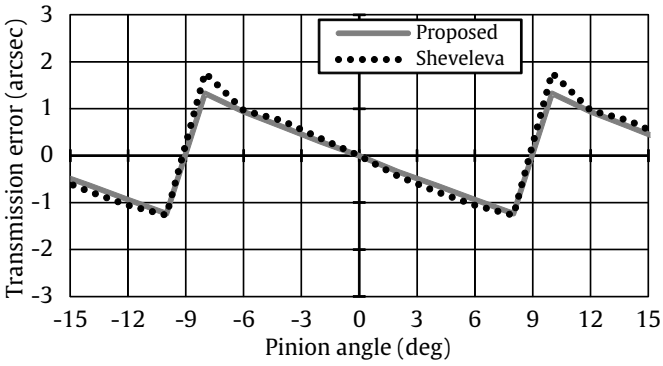
(a) Contact patterns for case 1



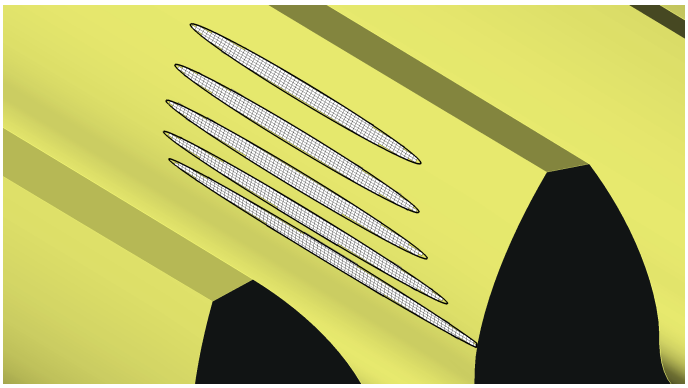
(b) Transmission errors for case 1



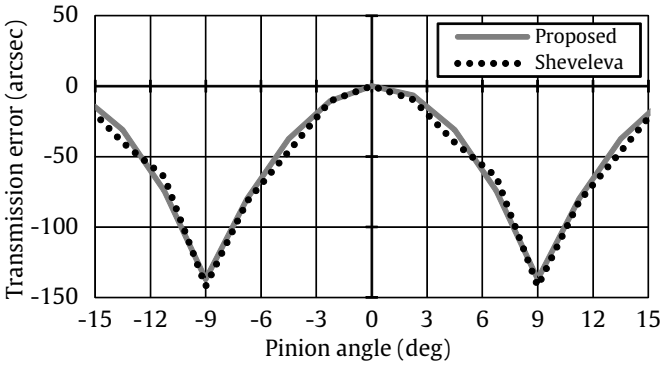
(c) Contact patterns for case 2



(d) Transmission errors for case 2



(e) Contact patterns for case 3



(f) Transmission errors for case 3

Figure 15: Comparison of results with the algorithm of Sheveleva.

refinement technique to compute a well defined border. And this difference is more significant when the border of the contact pattern is not aligned with the grid lines, as in case 2 (figure 15c).

Regarding the transmission errors, it can also be observed the coincidence of the results obtained by the two approaches. However, the functions provided with the algorithm of Sheveleva have a lower accuracy that is, again, due to the use a discrete grid of points and to the lack of refinement.

In addition to the above, the proposed approach has also been compared to the previous ap-

proach of the authors [36], that supposed a first approximation to the problem with the aim of testing the possibility of using this strategy. In that approach, the flowchart was simple and different steps were done massively (consuming an important time of computation in operations with nodes and triangles that were away from the area of interest) without considering the computational cost. While both (proposed and previous) approaches provide similar results, in the approach presented here the algorithms have been developed for the best performance. Taking this into account, the main differences between the two approaches are the following.

- In the previous approach, after performing a refinement step on the pinion mesh, the data field of the gear mesh was not updated. Then, the adaptive refinement of the gear mesh was done on the basis of an outdated data field, affecting the rate of convergence of the algorithm. The proposed approach was improved introducing this update step to ensure that each refinement step of a mesh is done from updated data according to the current geometry of the meshes at the moment of refining.

- In the previous approach, after each refinement step of the pinion and gear meshes, the computation of the data associated to the nodes of each mesh (angle of rotation to contact the other mesh in the algorithm CP and distance to the other mesh in the algorithm ICA) was done massively checking each node with each triangle of the opposite mesh, what is robust but not very efficient. In the current approach, an optimized procedure for updating the data of each node has been implemented, as explained in sections 6 and 7. The procedure has been designed for minimizing the number of operations without losing robustness, that is, ensuring that the data is updated with total accuracy.

- In the previous approach there were not mesh traverse operations (the operations were done massively), so the meshes were stored in the simplest form (array of nodal coordinates and array of connectivity of triangles). In the currently proposed algorithm, traverse operations are an important basis of the algorithm and a half-edge data structure [39] has been used to store the meshes. This structure is very convenient for these operations and supposed an important part of the improvement of the global efficiency of the algorithm.

- Furthermore, other minor operations have been optimized to lower the computational cost. For example, in the algorithm ICA, a traversing procedure has been implemented to compute the border of the instantaneous contact area from the cloud of segments obtained from the

refined triangles, instead of comparing each segments with all the rest to detect the next connected segment.

To compare the efficiency of the three approaches (the proposed one, the previous approach of the authors and the approach proposed by Sheveleva), the time of computation in cases 1 and 2 have been measured (case 3 is between cases 1 and 2 with respect to time of computation). For a higher detail, the measurement has been performed separately for the contact problem (CP) and for computing the instantaneous contact area (ICA) and for each position of the gear set along the gearing cycle. The average time per position is shown in figures 16a for case 1 and 16b for case 2. This time is highly dependent from the computer used, so only relative differences are considered significant.

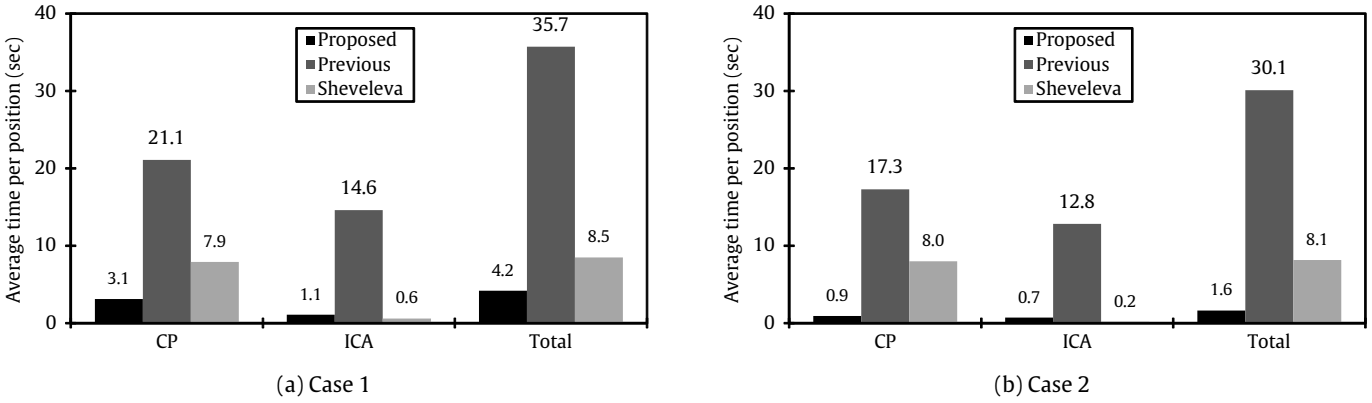


Figure 16: Average times of computation per position of the gear set.

From this comparison, several conclusions can be obtained:

- Computationally speaking, the more expensive approach is the previous one, followed by the approach of Sheveleva and the proposed approach is the least expensive.
- The algorithm of Sheveleva has a similar computational cost in both cases. In this algorithm, the time of computation is mainly dependent from the number of elements of the used grid and it is almost independent from the type of contact or the relative position of the gears. In the case of the proposed and previous algorithms, the computational cost depends on the size of the areas that must be refined and this size is highly dependent from the type of contact and the alignment status of the gears.

- The development of the proposed approach with respect to the previous approach allowed to lower the associated computational cost very significantly.
- The algorithm of Sheveleva takes most of the time to solve the contact problem (CP). After that, only an interpretation of the distance field in both meshes are required to estimate the contact pattern. This computation is very fast and takes almost no time. On the other hand, both proposed and previous algorithms takes a longer time to solve the contact problem (CP) than to compute the instantaneous contact area (ICA). This is related to the area that is necessary to refine to solve each problem and, in general, computing the border of the instantaneous contact area requires to refine a smaller area.

Considering the accuracy of the methods in figure 15 and their computational cost in figure 16, the proposed approach has a higher ratio accuracy vs. computational cost. With the use of the adaptive refinement technique, the time of computation is used in areas where it is necessary for solving the problems CP and ICA with a high accuracy.

10. Sensitivity of the algorithm to the virtual marking compound thickness

The sensitivity to the virtual marking compound thickness is an important feature of any algorithm for computing the contact pattern. If an algorithm has a low sensitivity to this parameter, the obtained instantaneous contact area will not be accurate and will not change with the parameter in a realistic way.

To verify the sensitivity of the proposed algorithm to the parameter δ, the same misaligned helical transmission of case 2 (table 1, figure 6b) has been used. The algorithm CP has been executed for an intermediate position of the gear set with the parameters presented in table 2, providing the position of the gear in perfect rigid contact with the pinion. Then, the algorithm ICA has been executed with five different values of δ (0.001, 0.005, 0.010, 0.015, 0.020, 0.025 mm), having the rest of parameters the values of table 3. The resulting instantaneous contact areas are presented overlapped in figure 17, alternating from black to white, where the smaller white area corresponds to $\delta = 0.001$ mm and the wider black area corresponds to $\delta = 0.025$ mm.

It can be observed that the evolution of the contact area with the increase of δ is progressive. The obtained areas are well formed and have no irregularities for the specified level of refinement. The shape and orientation of the ellipses are in accordance with the misalignment of the gears and their areas grow up to fill the face width of the gear when δ is increased.

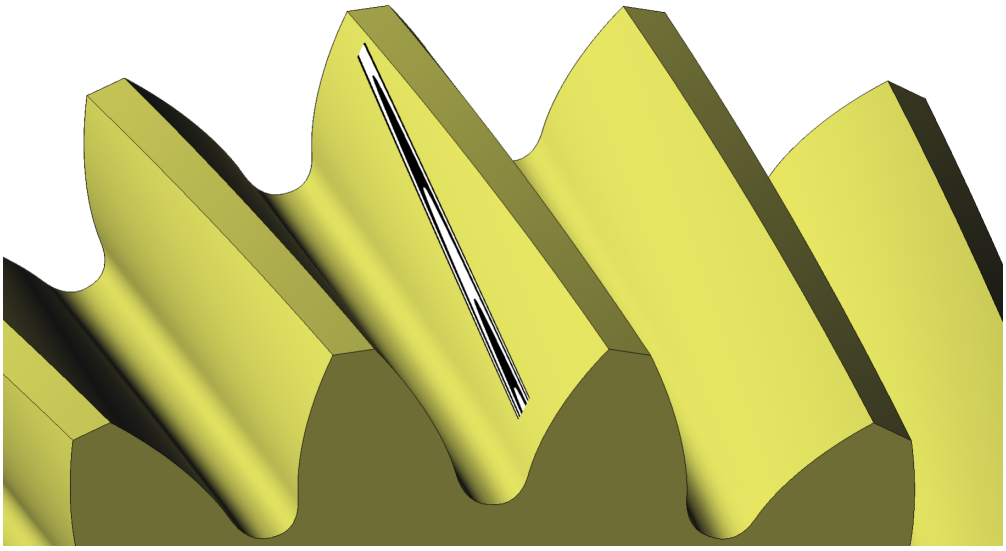


Figure 17: Instantaneous contact areas for different values of virtual marking compound thickness

This test also demonstrates how the obtained instantaneous contact area depends on δ. The optimal value of this parameter is the one that provides a contact area of a size that is similar to the size of the obtained contact area when considering elastic bodies (loaded tooth contact analysis). This optimal value mainly depends on the type of transmission, the material of the gears and the transmitted torque. The higher the torque and the softer the material, the higher value of δ that must be used for realistic results. In general, for steel-made gears transmitting a normal torque (according to the module of the transmission), a value of δ between 0.005 and 0.010 mm provides realistic results in most types of transmissions.

11. Computational cost versus accuracy

To verify the influence of the degree of refinement in both computational cost and accuracy, a new test has been performed. For this test, the helical standard transmission used in case 2 (table 1, figure 6b) has been selected. Furthermore, the TCA algorithm has been executed for this transmission in two situations: aligned and misaligned. In the misaligned case, the selected misalignment is the same that the one used in the performance test. In both cases, the TCA algorithm has been executed with 9 different values of the *triangle size tolerance for refinement* (S_{tol}): 0.10, 0.15, 0.20, 0.25, 0.30, 0.35, 0.40, 0.45, 0.50. This parameter represents the goal triangle size of the refinement loop and, therefore, it indicates the required degree of refinement.

The test has been executed in the same computer and the time of computation for both algorithms CP and ICA along the 9 different positions of the gearing cycle has been measured, obtaining an average value of time per position. This time of computation is highly dependent from the used hardware, so only relative differences are significant. Taking this into account, the average time for both algorithms CP and ICA, and for both alignment situations is presented in figure 18a.

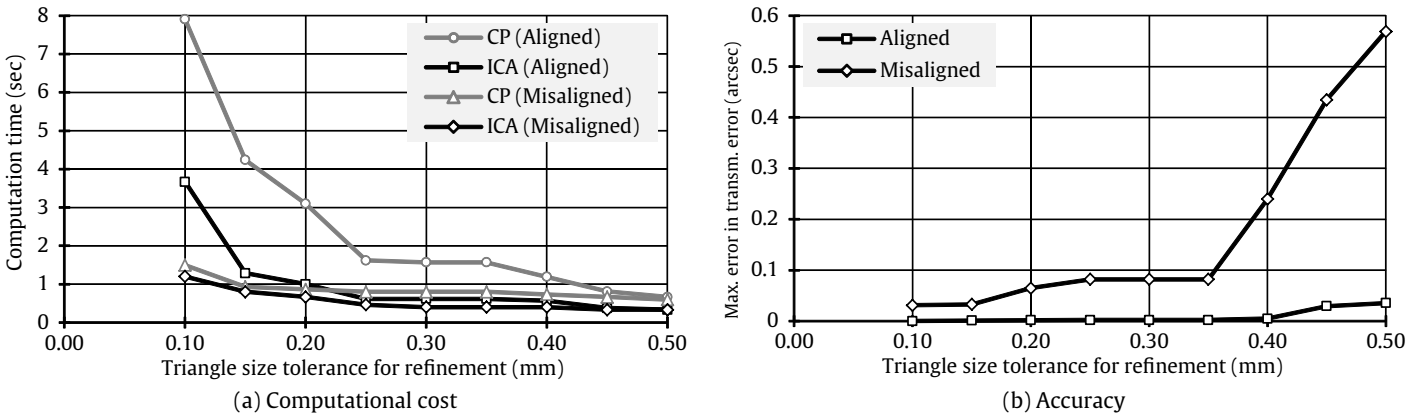


Figure 18: Computational cost and accuracy versus degree of refinement.

On the other hand, the accuracy of the function of transmission errors has also been estimated. In the aligned case, the theoretical transmission error is zero for all positions because it is the case of a standard involute transmission. Consequently, the error of the transmission error associated to a specific degree of refinement is defined as the maximum value of transmission error along the gearing cycle.

In the misaligned case, the expected function of transmission errors is a sawtooth function, as it was obtained in figure 12b. Thus, for a specific position of the gear set in the gearing cycle, the error of the transmission error is measured as the difference between the actual transmission error for that position and the value obtained from the linear regression of the different values of transmission error obtained along the gearing cycle. Then, the error of the function of transmission errors associated to a degree of refinement is the maximum error of transmission error along the gearing cycle. Taking into account these definitions, the accuracy of the computed transmission error versus the degree of refinement is presented in figure 18b.

From the results of this test, several conclusions can be obtained. According to figure 18a, the computation of the contact problem (algorithm CP) has a higher computational cost than the computation of the instantaneous contact area (algorithm ICA) in each tested case. The reason is

that area that has to be refined is usually smaller in the algorithm ICA.

It can also be observed in figure 18a that the computation of the contact problem in the aligned case has a higher cost than in the misaligned case. The reason, again, is that the area to refine is much wider in the aligned case (where the contact area extends along the full face width) than in the misaligned one (where the contact area is similar to a small truncated ellipse). And the results indicate that the time difference increases with the degree of refinement as the number of refined triangles do.

The computation of the instantaneous contact area in the aligned case has also a higher cost than in the misaligned case. This cost is related to the length of the border of the area, that must be refined to be computed with precision. And, in the aligned case, the border of the contact area is longer than in the misaligned one. However, the difference between both cases in the computation of the instantaneous contact area (algorithm ICA) is lower than the difference in the computation of the contact problem (algorithm CP).

The obtained function of transmission errors has a high degree of accuracy, being the error of this function lower than 0.6 arcsec in all tested cases and lower than 0.1 arcsec when the goal triangle size for refinement is under 0.4 mm. Evidently, the error increases when the degree of refinement decreases, but this increment is much more important in the misaligned case than in the aligned one.

It is proved that the accuracy of the results is parameter-dependent and can be controlled through the requested degree of refinement (S_{tol}). Obviously, if the requested degree of refinement is increased, the TCA algorithm provides results with a higher level of accuracy but at the cost of a longer time of computation. A good degree or refinement is an intermediate value where the ratio accuracy vs. computation time is optimal. In a specific case, this optimal value can be obtained testing different values from higher to lower up to the desired degree of accuracy. As a recommendation, the experience obtained with this work demonstrated that a value of S_{tol} between 3% and 6% of the *module* provides accurate results with moderate times of computation.

12. Conclusions

In this work, a new geometric approach for the Tooth Contact Analysis (TCA) of gear transmissions has been proposed. The TCA approach considers different positions of the gear set along the gearing cycle. For each position, the problem is divided into two sub-problems: (i) the computation

of the contact problem (algorithm CP) and (ii) the computation of the instantaneous contact area (algorithm ICA). Both algorithms have been designed independently but with a similar strategy that is based on the discretization of the tooth contact surfaces and the adaptive refinement of the meshes. Thus, for a specific position of the pinion, the refinement strategy is used by the algorithm CP to determine the angle of rotation that is necessary to apply to the gear to be in rigid contact with the pinion. Then, the transmission error associated to that position of the gear set is computed. On the other hand, the refinement strategy is used by the algorithm ICA to compute the instantaneous contact area associated to that position of the gear set. And, finally, the contact pattern is formed by the superposition of the contact areas obtained for the different positions of the gear set along the gearing cycle.

The TCA algorithm has been tested with three types of gear transmissions (involute spur, involute helical and modified spur) and with different alignment status (aligned and misaligned) providing different contact types (line contact, point contact and point edge-contact). The results have been compared with the TCA algorithm proposed by Sheveleva and with a previous algorithm of the same authors demonstrating the superior ratio accuracy vs computational cost of the proposed approach.

The sensitivity of the approach to the virtual marking compound thickness has been tested and the relation between accuracy and computational cost has been analyzed for different degrees of refinement.

The tests demonstrated the robustness of the approach, the fast convergence of the adaptive refinement strategy and the accuracy of the results. The cases studied corroborate that the proposed approach is applicable any type of gear, with standard or modified tooths and in aligned or misaligned situations.

13. Acknowledgments

The authors express their deep gratitude to the Spanish Ministry of Science and Innovation (MICINN) for the financial support of research project ref. DPI2013-47702-C2-2-P.

References

[1] F. Litvin, G. Kai, Investigation of conditions of meshing of spiral bevel gears (in Russian), in: Proceedings of Seminar of Theory of Mechanisms and Machines, 1962, pp. 92–93.

[2] F. Litvin, G. Kai, Improvement of conditions of meshing of spiral bevel gears (in Russian), in: Proceedings of Seminar of Theory of Mechanisms and Machines, 1964, pp. 98–99.

[3] M. Baxter, Basic geometry and tooth contact of hypoid gears, Industrial Mathematics 11 (2) (1961) 19–43.

[4] Understanding Tooth Contact Analysis, The Gleason Works, Rochester, NY, 1981.

[5] E. Klingelnberg, Kimos: Zahnkontaktanalyse für Kegelräder, Klingelnberg, 1996.

[6] H. Stadtfeld, Formgenaue Kegelradtriebsatze durch Oerlikon CDS-3D, Oerlikon Buhrle AG, Zurich, 1988.

[7] F. L. Litvin, J. Zhang, R. F. Handschuh, Crowned spur gears: Methods for generation and tooth contact analysis - Part I: Basic concepts, generation of the pinion tooth surface by a plane, Journal of mechanisms, transmissions, and automation in design 110 (3) (1988) 337–342.

[8] F. L. Litvin, J. Zhang, R. F. Handschuh, Crowned spur gears: Methods for generation and tooth contact analysis - Part II: Generation of the pinion tooth surface by a surface of revolution, Journal of mechanisms, transmissions, and automation in design 110 (3) (1988) 343–347.

[9] J. L. Chen, C. H. Tseng, C. B. Tsay, Bearing contact analysis of the involute spur planetary gear train, Transactions of the Canadian Society for Mechanical Engineering 25 (2) (2001) 163–190.

[10] J.-L. Li, S.-T. Chiou, Surface design and tooth contact analysis of an innovative modified spur gear with crowned teeth, Proceedings of the Institution of Mechanical Engineers 219 (2) (2005) 193–207.

[11] C. B. Tsay, Z. H. Fong, Tooth contact analysis for helical gears with pinion circular arc teeth and gear involute shaped teeth, Journal of mechanisms, transmissions, and automation in design 111 (2) (1989) 278–284.

[12] F. Litvin, J. Lu, Computerized design and generation of double circular-arc helical gears with low transmission errors, Computer Methods in Applied Mechanics and Engineering 127 (1-4) (1995) 57–86.

[13] S. L. Chang, C. B. Tsay, C. H. Tseng, Kinematic optimization of a modified helical gear train, Journal of Mechanical Design, Transactions of the ASME 119 (2) (1997) 307–314.

[14] P. H. Feng, F. L. Litvin, D. P. Townsend, R. F. Handschuh, Determination of principal curvatures and contact ellipse for profile crowned helical gears, Journal of Mechanical Design, Transactions of the ASME 121 (1) (1999) 107–111.

[15] W.-S. Wang, Z.-H. Fong, Tooth Contact Analysis of Longitudinal Cycloidal-Crowned Helical Gears With Circular Arc Teeth, Journal of Mechanical Design 132 (3) (2010) 031008.

[16] F. L. Litvin, Y. Zhang, J. C. Wang, R. B. Bossler, Y. J. D. Chen, Design and geometry of face-gear drives, Journal of Mechanical Design - Transactions of the ASME 114 (4) (1992) 642–647.

[17] F. L. Litvin, A. Fuentes, M. Howkins, Design, generation and TCA of new type of asymmetric face-gear drive with modified geometry, Computer Methods in Applied Mechanics and Engineering 190 (43-44) (2001) 5837–5865.

[18] F. L. Litvin, A. Fuentes, C. Zanzi, M. Pontiggia, R. F. Handschuh, Face-gear drive with spur involute pinion: geometry, generation by a worm, stress analysis, Computer Methods in Applied Mechanics and Engineering 191 (25-26) (2002) 2785–2813.

[19] C. Zanzi, J. I. Pedrero, Application of modified geometry of face gear drive, Computer Methods in Applied Mechanics and Engineering 194 (27-29) (2005) 3047–3066.

[20] F. L. Litvin, I. Gonzalez-Perez, A. Fuentes, D. Vecchiato, B. D. Hansen, D. Binney, Design, generation and stress analysis of face-gear drive with helical pinion, Computer Methods in Applied Mechanics and Engineering

194 (36-38) (2005) 3870–3901.

[21] F. L. Litvin, J. Zhang, H. T. Lee, R. F. Handschuh, Transmission errors and bearing contact of spur, helical, and spiral bevel gears, Gear Technology 7 (4) (1990) 8–13, 44.

[22] F. L. Litvin, A. G. Wang, R. F. Handschuh, Computerized generation and simulation of meshing and contact of spiral bevel gears with improved geometry, Computer Methods in Applied Mechanics and Engineering 158 (1-2) (1998) 35–64.

[23] F. L. Litvin, A. Fuentes, Q. Fan, R. F. Handschuh, Computerized design, simulation of meshing, and contact and stress analysis of face-milled formate generated spiral bevel gears, Mechanism and Machine Theory 37 (5) (2002) 441–459.

[24] V. Simon, Influence of tooth errors and misalignments on tooth contact in spiral bevel gears, Mechanism and Machine Theory 43 (10) (2008) 1253–1267.

[25] H. Li, W. Wei, P. Liu, D. Kang, S. Zhang, The kinematic synthesis of involute spiral bevel gears and their tooth contact analysis, Mechanism and Machine Theory 79 (2014) 141–157.

[26] C. Y. Lin, C. B. Tsay, Z. H. Fong, Mathematical model of spiral bevel and hypoid gears manufactured by the modified roll method, Mechanism and Machine Theory 32 (2) (1997) 121–136.

[27] S. Vilmos, The influence of misalignments on mesh performances of hypoid gears, Mechanism and Machine Theory 33 (8) (1998) 1277–1291.

[28] V. Simon, Optimal Tooth Modifications in Hypoid Gears, Journal of Mechanical Design 127 (4) (2005) 646.

[29] Q. Fan, Computerized Modeling and Simulation of Spiral Bevel and Hypoid Gears Manufactured by Gleason Face Hobbing Process, Journal of Mechanical Design 128 (6) (2006) 1315.

[30] Y.-P. Shih, A novel ease-off flank modification methodology for spiral bevel and hypoid gears, Mechanism and Machine Theory 45 (8) (2010) 1108–1124.

[31] C.-H. Lin, Z.-H. Fong, Numerical tooth contact analysis of a bevel gear set by using measured tooth geometry data, Mechanism and Machine Theory 84 (2015) 1–24.

[32] G. I. Sheveleva, A. E. Volkov, V. I. Medvedev, Algorithms for analysis of meshing and contact of spiral bevel gears, Mechanism and Machine Theory 42 (2) (2007) 198–215.

[33] M. Kolivand, A. Kahraman, A load distribution model for hypoid gears using ease-off topography and shell theory, Mechanism and Machine Theory 44 (2009) 1848–1865.

[34] S. M. Vijayakar, A Combined Surface Integral and Finite Element Solution for a Three-Dimensional, International Journal for Numerical Methods in Engineering 31 (1991) 525–545.

[35] A. Bracci, M. Gabiccini, A. Artoni, M. Guiggiani, Geometric contact pattern estimation for gear drives, Computer Methods in Applied Mechanics and Engineering 198 (17-20) (2009) 1563–1571.

[36] F. Sanchez-Marin, A. Fuentes, J. L. Iserte, I. Gonzalez-Perez, A new geometrically adaptive approach for tooth contact analysis of gear drives, in: International Gear Conference 2014, Lyon, France, 2014, pp. 486–494.

[37] F. Litvin, A. Fuentes, Gear Geometry and Applied Theory (2nd edition), Cambridge University Press, New York, 2004.

[38] V. Simon, Computer simulation of tooth contact analysis of mismatched spiral bevel gears, Mechanism and Machine Theory 42 (3) (2007) 365–381.

[39] M. Mäntylä, An Introduction to Solid Modeling, Principles of computer science series, Computer Science Press,

1988.