Implicit modeling of abrasion process based on potential field

Charly Euzenat, Sylvain Lavernhe, Christophe Tournier

Implicit modeling of abrasion process based on potential field

Charly Euzenat[a], Sylvain Lavernhe[a], Christophe Tournier[a]

[a]*LURPA, ENS Paris-Saclay, Univ. Paris-Sud, Université Paris-Saclay, 94235 Cachan, France.*

Abstract

This article presents a novel numerical model to generate abrasion surface topography. The main difficulty encountered regarding abrasion simulation is the numerical complexity caused by the number of interactions. In addition, the resulting surface topography is not only the result of cutting but also repelling and deforming matter. Thus the geometry is not strictly defined by the intersection between the grains and the mechanical part. Considering both numerical and physical constraints, an implicit geometrical model has been developed. The features of an implicitly defined surface make it a suitable candidate to address abrasion removal phenomenon. Within this formulation, the surface is embedded in a volumetric data set and is retrieved as the set of points having the same field value. Thus, the correlation between a scalar field and the specimen surface has been leveraged to model the abrasive grain actions. The removal action occurs throughout the penalization of the volumetric data field. The extension of each particle interaction range to model the plastic flow is done with negligible additional cost.

Keywords: Abrasion simulation, Implicit modeling, 3D topography, Potential field

1. Introduction

In an industrial context where production quality must be increased within a controlled impact on production costs, the numerical chain must be mastered at each step with suitable and relevant prediction methods. When it comes to complex surfaces obtained as a result of an abrasion process, the possibilities to predict the geometry are often limited to empirical rules. The simulation of abrasive topography remains a complex issue. The number of interactions involved in the process raises the numerical complexity so that it would require a large computational power. The simulation results are not assured in most cases, let alone obtained in a reasonable time. Mechanical approaches are directly concerned by this complexity. Nevertheless, abrasion processes have been commonly used to improve functional surface quality from raw machined surfaces. The respect of the geometrical definition is mandatory without dragging additional prohibitive costs. In this context, the development of both an efficient and relevant simulation framework for abrasion would lead to increase production quality.

The level of development regarding CAM software in the industry at this time allows simulating machining operations to ensure neither global collisions between tools and mounting devices nor local unwanted interactions with the machined part are encountered. The details scale in this context is not thinner than few millimeters. The time spent to compute the surface may vary from minutes to hours against the number of tool paths involved in the operation. The virtual visualization of the marks, at a lower order of magnitude, induced by the tool against the followed path is achievable in laboratory conditions [1]. The main algorithms developed to compute the machining surface rely either on dexel, voxel or z-buffer. The performance of these methods has been demonstrated with their implementation on general purpose graphical processing unit [2]. Industrial implementation is still required for these laboratory-scaled algorithms to be integrated. Nevertheless, results obtained in this context are compatible with manufacturer expectations in terms of interactivity and rendering speed. These methods are not directly operable for abrasion simulations as the core functions handling the interaction between the tool and the material assumes boolean operations [3]. While this hypothesis permits high-performance computing, the relevance is incompatible with a description of the abrasion process.

On the other hand, applying the concept of continuum mechanics to abrasion is under high investigation in the literature. It defines a framework of equations available to model the inner mechanical equilibrium structuring matter. Single grain scratching and impact have been successfully simulated using both FEM and meshless numerical methods [4, 5]. The efforts are driven toward a better understanding of the mechanisms of abrasion. Output surface topography and efforts correspond to experimental measurements. Having regard to the time encountered to simulate the interaction between a single grain and a sample of a few square millimeters, the application of these mechanical methods to a complete abrasion operation remains unreachable given an industrially compatible computer capability. Based on the previous considerations and the implied shortcomings, an innovative approach to simulate abrasion has been developed. The method presented remains a geometric simulation. Mechanical behavior considerations are emulated through potential equilibrium. The advantages of using implicit surface representation are suited to take into account the complexity of the abrasion interactions while maintaining an efficient computational cost. The application of this framework is already conceivable in an industrial context.

This article is organized as follows: section 2 reviews the currents state of the art regarding abrasion modeling and simulation. The reasons to develop an implicit simulation framework are then discussed against the difficulties encountered with abrasion simulation. The core equations of the model and the associated numerical methods are presented in section 3. Finally, a case study is presented in section 4 to confront the numerical outcomes and the model parameters relative influence with experimental data.

2. Process description and simulation

2.1. State of the art

The abrasion process is characterized by the removal of matter under constraints generated by the path of sharp particles. Contrary to machining which is a process where the tool geometry is mastered to cut the material under a well-defined shape, abrasion resulting geometry remains due to arbitrary interactions. The chip thickness, the induced grinding force along with the thermal effects models have been developed. A review of these grinding models has been proposed by [6] and a multiscale review of the experimental and simulation approaches has been proposed by [7]. Several ways of defects have been identified by [8, 9]. Three hypotheses have been made to describe the removal mechanics. The flow hypothesis and the abrasion hypothesis are mechanical interactions inducing high local strain and stress to remove and repel matter. The third category is the chemical action and which is not taken into account in the present study.

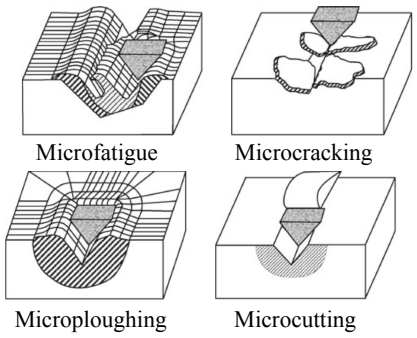

Figure 1: Abrasion defect mechanisms [8, 9]

Among mechanical induced defects, four main schemes have been identified. Microploughing is the result of plastic deformation occurring while a grain gets in contact with the surface. It repels the material which agglomerates at the sides of the groove. This type of deformation induces no material removal. Microfatigue occurs when several grains paths are located in the same area until the material fails. Brittle material like ceramics is more subjected to microcracking as a result of high stress induced by abrasive particles. During the abrasion of hardened steel, microcutting prevails which cut the material in a clean surface. The removal rate is also an influencing parameter. Microploughing occurs during the polishing of soft metals while the removal rate is low, contrary to microcutting which can be observed while the abrasive grains are sharp. The pile up model developed in [10] accounts for the plastic deformation which repels matter at the edge of the scratch in microploughing. The type of interactions encountered depends on the specimen, the abrasive topography and relative hardness. The surface roughness is affected by the type of abrasion mechanisms and results in the summation of this phenomenon whose relative influence is not quantifiable.

To predict the surface obtained following a finishing process from an empirical point of view, several models exist. One of the most common to induce the removal rate is the Preston model [11] (Eq. 1).

$$\frac{dz}{dt} = K_p.P.V \tag{1}$$

This equation reflects the fact that the removal rate is related to both the contact pressure P and the relative velocity V. The Preston coefficient K_p is the equivalent for the efficiency of the abrasive tool against process parameters. Several other laws derived from the Preston model have been proposed to improve the prediction relevance.

$$\frac{dz}{dt} = K_p.P^{\alpha}.V^{\beta} \tag{2}$$

Klocke [8] has developed a general-purpose model for abrasion (Eq. 2) with two additional degrees of freedom in the equation. This model requires more effort for identification yet offers a wider range of possibilities while modeling various tool-workpiece pairs. Whether are the form of the equations, the laws identified above are mainly used as a description of the interactions but are not yet implemented in simulation on their own.

Mechanical simulations including a material behavior model and a numerical EDP solver have highlighted both phenomena of repelling and cutting material. The numerical implementations are known as *finite element method* or *smooth particle hydrodynamics* whether the discretization relies on a mesh or on particles. These methods are the closest numerical way to emulate a specimen under manufacturing process constraints. The simulation results in Fig. 2 by Leroch et al. [12] are in good agreement with the experimental measures. While providing valuable information concerning strain and stress in the specimen, these algorithms imply high computation expenses, thus long simulation time. In this context, using mechanically based algorithms to simulate a complete process abrasion operation is out of reach given a reasonable computing power.

The topography in grinding has been modeled in a significant number of articles as the cinematic mapping of the grinding wheel on the surface [13, 14]. The simulation of the workpiece topography has been numerically obtained with the Z-Buffer algorithm. This boolean mapping has been improved with both abrasive characteristics of the grain and workpiece Brinell hardness by [15].

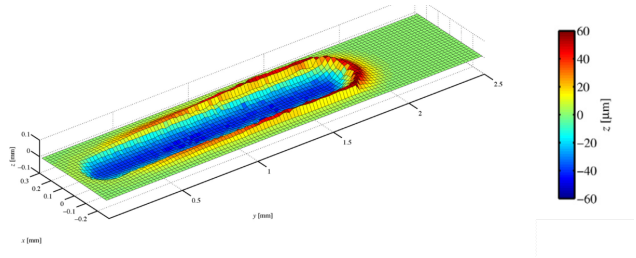

Figure 2: Single grain scratching simulation [12]

2.2. Proposed approach

The proposed implicit model tends to simulate material cutting and repelling: *microcutting* and *microploughing* in an efficient geometrical model. The surface topography obtained with this model is the results of theses two actions.

The geometric abrasion simulation presented requires the use of a numerical surface description. Two major surface representation methods are commonly used in computer graphics: *parametric* or *implicit* descriptions. Parametric surfaces are defined by an explicit evaluation of a function whereas implicit surfaces are extracted as the iso-level contour of a volumetric data set. The material removal during abrasion process modifies the surface. Given an explicit representation, it is mandatory to find a 3D displacement field to move the skin surface. To be relevant, this vector field is often related to a mechanical behavior model. The dynamic equations are integrated to compute a current position of the surface. To the best of our knowledge, there is no definition of displacement vector field which is not based on an underlying mechanical behavior model, apart from machining simulation modeled by boolean operations. Explicit boolean operations lead to efficient algorithms with considerable shortcomings considering physical phenomena which prevent them from being used for abrasion simulation.

An implicit formulation has been chosen to overcome the computation of an explicit deformation field. The implicit formulation has been preferred in computer graphics concerning smooth deformations of objects or intuitive modeling of complex surfaces [16]. The features of implicit surface representation have been successfully adapted to complex surfaces modeling throughout intuitive user interaction and free-form sculpting [17]. Indeed, implicit surfaces have blending properties [18]. The summation of implicit primitives leads to smooth transitions when the addition operator between field values is used. These features are suitable for native modeling of the geometries encountered with microploughing actions. Moreover, this framework allows simulating exact contact between flexible objects and extended deformation beyond the contact area [19]. Finally, using implicit representation is not common in the computer-aided manufacturing community and process simulation. Compared to mechanical simulations, implicit representation of surfaces offers compromises between numerical efficiency and physical emulation relevance.

An implicit surface Γ is the set of points $\mathbf{X}$ that is the solution to a scalar equation (Eq. 3). The surface is embedded in a volumetric data set F_t which contains both the position of the surface and numerical values in the domain.

$$\Gamma(t) = \{\, \mathbf{X} \mid F_t(\mathbf{X}) \;=\; 0 \,\} \tag{3}$$

$$F_t : \left| \begin{array}{ccc} R^3 & \longrightarrow & R \\ \mathbf{X} & \longmapsto & F_t(\mathbf{X}) \end{array} \right. \tag{4}$$

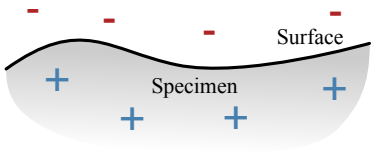

Figure 3: Specimen skin surface

Eq. 3 means that the variable skin surface of the specimen is defined where the scalar field is zero. Thus, the modification of the volumetric data set F_t (Eq. 4) throughout time leads to an evolving surface. The deformation of the surface is done against the action of the abrasive tool under an interaction scheme. The implicit formalism defines the surface where the scalar field F_t is null. Thus, the sign of the field is different inside or outside the matter. Within the proposed model, the sign has been positively defined inside the workpiece. This principle is pictured in Fig. 3. The construction of the initial scalar field F_{t_0}, given the initial specimen surface, is required to initialize the simulation. This process is an issue which has been addressed in the literature and is often referred as implicitization [20]. The complementary problem of extracting an iso-surface from a scalar field requires solving the equation with numerical methods. This issue has been first addressed by Lorensen and Cline [21] thanks to the well-known marching cube method.

3. Implicit model

3.1. Particle interaction scheme

In the abrasion process, the free surface of the mechanical part evolves under the action of abrasive particles. This evolution has been modeled by means of the deformation of the scalar field F_t rather than explicitly interacting with the surface. This implicit model takes into account the fact that the abrasion topography is the result of an equilibrium between particles removal action and the specimen resistance. Considering that a scalar penalty field W is attached to each abrasive particle as defined by Eq. 5, the removal action consists in subtracting this field to the specimen. Eq.6 defines this temporal evolution scheme where F_t is the image of the surface according to time. In this context, a basic integration leads to Eq. 7 which defines the evolution of F_t from initial to final time.

$$W : \left| \begin{array}{ccc} R^3 & \longrightarrow & R \\ \mathbf{X} & \longmapsto & W(\mathbf{X}) \end{array} \right. \tag{5}$$

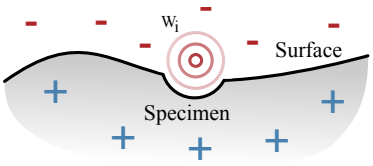

Figure 4: Abrasive potential W_i of a particle

$$F_t(\mathbf{X}) = F_{t-1}(\mathbf{X}) - W_i(\mathbf{X}) \tag{6}$$

$$F_t(\mathbf{X}) = F_{t_0}(\mathbf{X}) - \sum_i W_i(\mathbf{X}) \tag{7}$$

This scheme of surface deformation is parallel and can be implemented efficiently on parallel computational architecture. The initial scalar field F_{t_0} is responsible for the specimen behavior while the set $\{W_i | i \in [1; n]\}$ accounts for each abrasive particle effect on the surface.

3.2. Abrasive potential

Each abrasive particle owns a defined scalar potential W_i which models its effect on the specimen, which is reproduced by Eq. 7 and pictured in Fig. 4. Its value modifies the scalar field associated with the workpiece surface. The right definition and parameterization of the particle potential remains mandatory to propose a relevant model of the abrasion process. For performance issues, it is not suitable to have a volumetric set of value rather than a function to generate the effects during the process simulation. Process driven constraints have led to the proposed following parameterization.

The penalty field W is the composition of two functions $K \circ D$. Function K has been named the kernel of the abrasive grain and defines its removal influence. A generic form for the particle kernel function is given in Eq. 9.

$$K : \left| \begin{array}{ccc} R & \longrightarrow & R \\ l & \longmapsto & K(l) \end{array} \right. \tag{8}$$

The kernel function K has been defined as the sum of gaussian kernels. This formulation is commonly used in computer graphics to construct implicit surfaces [22, 18]. The proposed kernel has a compact support over the characteristic distance h (Eq. 9). According to potential F_t definition, a positive kernel value leads to matter removal while a negative one accounts for repelling matter. The corresponding graphical representation for the two gaussian functions is plotted in Fig. 5. This function may be defined once for any particle or vary depending on the own characteristics of each particle. The abrasion interaction follows the general rules below:

1. The action is limited within a defined range which means its potential must be zero far from its center.
2. The maximum influence is encountered close to the center of the particle.
3. Each particle has a characteristic influence radius defined by its own size.
4. The influence on the machined surface decrease with the distance

The schematic representation of these assumptions is presented in Fig. 6. The four points show the general shape of an abrasive kernel given the constraints of abrasion.

$$\begin{cases} K(l) = \sum_{k=1}^{2} \alpha_k \exp\left(-\dfrac{(l-\mu_k)^2}{\sigma_k^2}\right) & l \leq h \\ 0 & l > h \end{cases} \tag{9}$$

The influence of the three parameters (α, μ, σ) used to define W is pictured in Fig. 6. Coefficient α accounts for the intensity of the potential. σ allows to modify the influence distance and thus represents the abrasive particle radius. The position of the kernel is modified with the definition of a non-zero value for μ. Other parameterization could have been considered such as kernel based on polynomials expression.

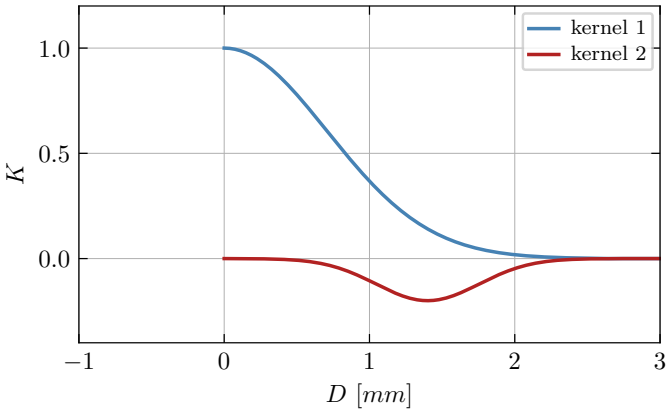

Figure 5: Gaussian kernel definition

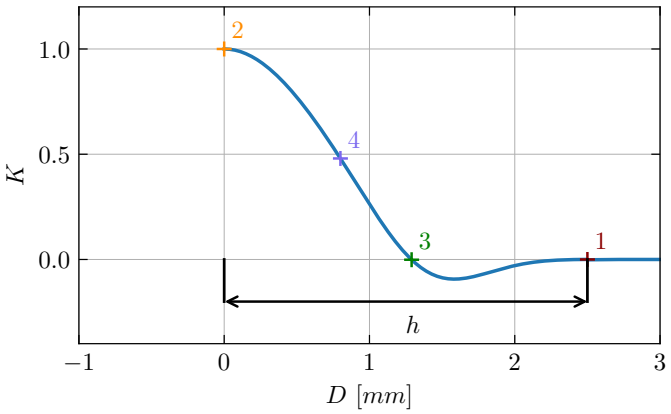

Figure 6: Gaussian kernel definition

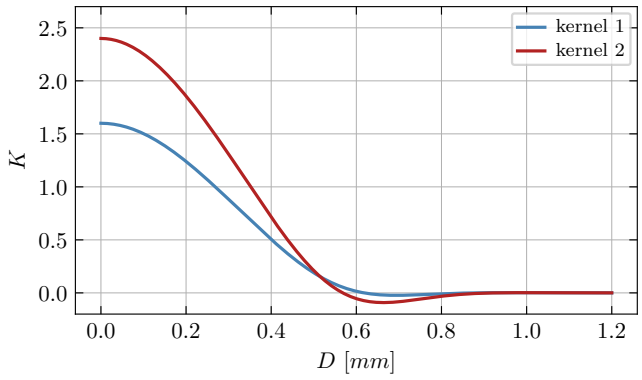

Figure 7: Kernels definition

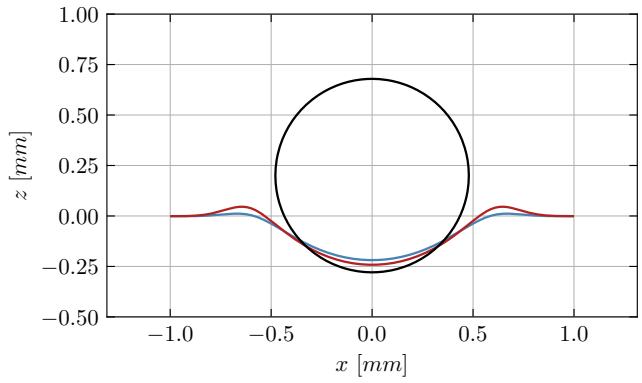

Figure 8: Corresponding iso-surfaces

Both Fig. 7 and Fig. 8 highlight the influence of the kernel definition on the response surface. The matter repelled during the process can be modeled by the negative part of the kernel between point 1 and 3 on Fig. 6. The base particle radius has been plotted in black in Fig. 8. The resulting surface is not the exact contour of the particle that would have been expected from a boolean operation. The implicit equilibrium between the surface potential and the particle kernel leads to an intermediate contour between the original surface and the particle-based radius.

D is the function which gives the distance between the considered point in 3D space $\mathbf{X}$ and the particle center point $\mathbf{P}_i$. Thus, the modeling of each abrasive grain leads to a single parameterization for function D. Using Eq. 10, the propagation of the abrasive action is centered on the point $\mathbf{P}$ as the function D is based on radial distance. In addition, the distance function can be modified to model the path followed by each grain. For a line, point $\mathbf{P}_i$ and unit vector $\mathbf{U_i}$ define the position and orientation of the path in the workspace coordinate system (Eq. 11). This avoids sampling the path as a set of close points with radial basis influence and allows to compute the passage of a grain in a single operation. More complex paths could be modeled in this manner, nevertheless it is suitable that the distance can be expressed as an analytical function. For more complex paths, a local approximation by lines or circles is conceivable.

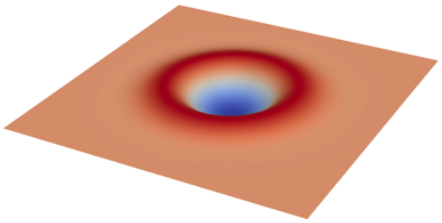

Figure 9: Indentation (Eq. 10)

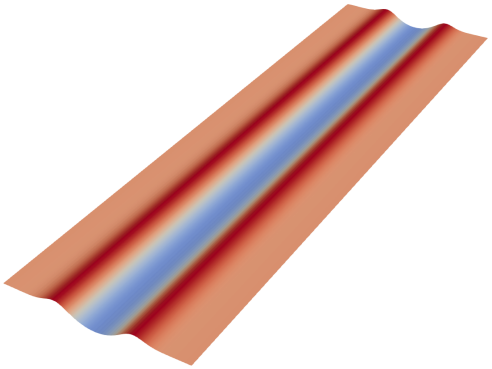

Figure 10: Linear scratching (Eq. 11)

Fig. 9 and Fig. 10 show the surfaces resulting from the two previous distance functions. These surfaces have smooth characteristics since the kernel function K is based on Gaussian kernels.

$$D_i(\mathbf{X}) = \| (\mathbf{X} - \mathbf{P}_i) \| \tag{10}$$

$$D_i(\mathbf{X}) = \| (\mathbf{X} - \mathbf{P}_i) \ \wedge \mathbf{U_i} \| \tag{11}$$

3.3. Initial potential definition

The abrasion simulation presented relies on the definition of an initial potential as stated in the Eq. 7 by the term F_{t_0}. This field accounts for the raw surface of the specimen before the operation. Its construction is made from the initial surface which is supposed to be known before the beginning of the simulation. The uniqueness of the field is not part of the implicitization method so that other constraints must be imposed. The signed distance field is often used to obtain a solution for this problem. Apart from numerical issues, the value of the field has no influence on the underlying iso-surface for graphical applications. Concerning the specimen modeling, this offers the opportunity to take into account mechanical properties. The field has been defined by Eq. 12 and the corresponding graph is plotted in Fig. 11.

$$F_{t_0}(\mathbf{X}) = F_{t_0}(z) = b\ (\ 1 - exp\ (\frac{z}{a})\ ) \tag{12}$$

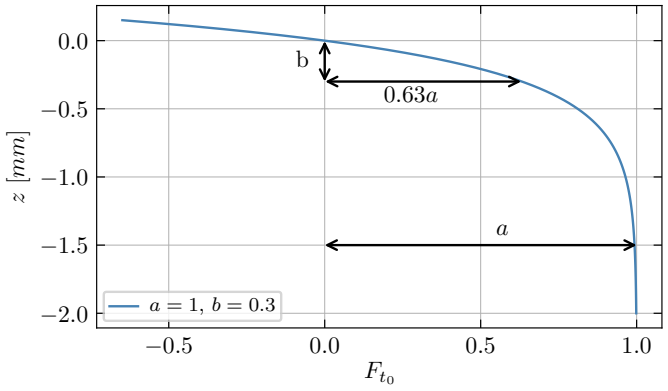

Figure 11: Initial distance function

3.4. Numerical implementation

An implementation as been done with the PyOpencl library [23], which is a binding between *Python* and *OpenCl*. This allows both implementation efficiency and computational performances. The surface extraction from the scalar field defined above in Eq. 3 requires the use of a numerical spatial discretization. The scene is defined as a rectangular bounding box which is sliced according to a given step as shown in Fig. 12. This leads to a set of points on a grid. Each of these points carry a scalar field value and thus allows using the marching cube algorithm [21]. The parallel implementation in question in this study has been done following the algorithm 1. The complete simulation requires the computation of the penalty field induced by P abrasive grains over a grid of N points. The complexity of the algorithm is thus defined by $C = N.P$. Contrary to mechanical simulations which requires a step by step resolution of the equations, the implicit model proposed allows to obtain the surface topography as a result of an acausal simulation. The main limitation of the implicit representation comes from the number of grid points N which may reach high value if the volume area which contains the workpiece surface is important.

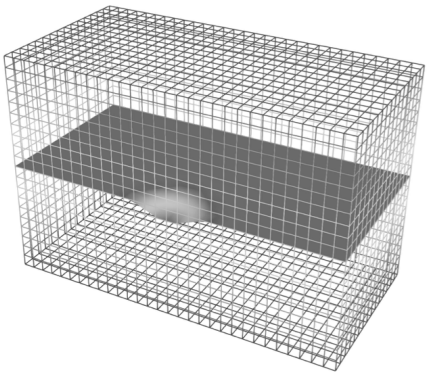

Figure 12: Spatial grid for marching cubes

Algorithm 1: Pseud-code for simulation

One grid point = one process
for *each abrasive particle* $\mathbf{X}_a$ **do**
  compute distance D between $\mathbf{X}_g$ et $\mathbf{X}_a$
  evaluate K according to D
  subtract K from $F(\mathbf{X}_g)$
end
extract surface from scalar field F

4. Numerical investigations

The objective of this study is to highlight the main features of the presented model against the main involved parameters. For this study, a small rectangular area is investigated and the path of the grains have been assumed to be linear and evenly distributed over the spatial domain. The resulting interactions between grains and workpiece aim at improving the surface roughness by a succession of operations with increasingly finer grains. The macroscopic surface parameters that have been studied are the mean height of the surface and the surface roughness. The mean height of the surface is defined in Eq. 13. Eq. 14 *(ISO 25178-2)* defines the surface roughness as the integral of the mean height z relative to the median plane of the surface. These criteria are used to provide a global overview of the surface topography.

$$z_{mean} = \frac{1}{A}\int_S z\, ds \tag{13}$$

$$S_a = \frac{1}{A}\int_S |z - z_{mean}|\, ds \tag{14}$$

Two cases have been considered for this first evaluation of the model. Among the parameters, the number of grain paths has been studied to observe the evolution of the surface characteristics during the abrasion operation. The relative height of the grains as well as grain radius have been changed to investigate their respective influence. The investigation presented in the following sections is divided in three main scenarios. The first one consists in paths at constant height, the second is linear varying height against the number of paths and in the third one, the grain size influence is investigated.

4.1. Path generation

The test surface is a plane which normal $\vec{z}$ is upward and vertical. The relative motion between the tool and the workpiece has been modeled by an uniform distribution of lines. Let Δ_x and Δ_y be the dimensions of the bounding box of the surface. The paths of the grains are assumed to be parallel to the surface with an evenly distributed orientation and position. To obtain such a set of lines, the angle α for the line orientation is a random variable following a uniform distribution between 0 and π. The position is determined with only one parameter as the distance to the center of the surface domain. This distance r is also a random variable between 0 and R.

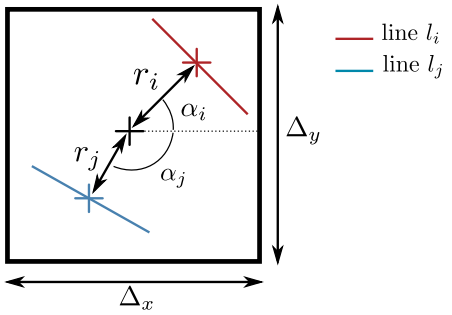

Figure 13: Path generation

$$\begin{array}{rclcrcl} x & = & r.cos(\alpha) & & u & = & -sin(\alpha) \\ y & = & r.sin(\alpha) & & v & = & cos(\alpha) \\ z & = & z_k & & w & = & 0 \end{array} \quad (15)$$

The rectangular area which has been isolated is 2 *mm* wide. The set of Eq. 15 defines the path generation and completes the Fig. 13. The two parameters r and α are used to generate the lines followed by each grain. Each line is defined by one point and one normal, which leads to the following set of n lines:

$$L_k = \{x_i, y_i, z = z_k, u_i, v_i, w = 0 \mid i \in [1; n_k]\} \quad (16)$$

Since the lines are assumed to be parallel to the surface, coefficient w is set to zero for any line. The relative height of the lines z is constant for one set of lines L_k. Thus, this set defines a scalar field as the sum of each contained line (Eq. 17).

$$W_{L_k} = \sum_{i=1}^{n_k} W_i \quad (17)$$

The definition of grain paths requires to compute the number of paths against time given a certain abrasive distribution in order to compare the results for different grain sizes. The abrasive paper characteristics are defined by the *(ISO 6344)* standard.

$$n_g = V.J_g.l.(t - t_0) \quad (18)$$

Eq. 18 defines the number of grain paths given process parameters and time. V is the relative velocity between the abrasive and the workpiece, J_g is the density of grains per surface unit and l is the perpendicular length of the workpiece relative to the velocity. The geometric modeling proposed in this section may not accurately reflect the characteristics of real statistical distributions encountered during abrasion. The problem of path identification given a real process is not assumed. There is not reason against the usage of the model with more complicated and relevant paths depending on the application.

4.2. Simulations

The results presented have been simulated on a desktop computer which processor is an *Intel i7-4770HQ*. For all the simulations considered in this study, a single spatial grid has been used. Its characteristics are given in Table 1. Fig. 18 pictures one example of the surface rendering after line scratching by multiple grains. In these conditions, the simulation time to compute the topography resulting from the abrasion is approximately a few milliseconds, whose majority is used to extract the surface using the *marching cubes* algorithm. Several approaches in the literature can be found to optimize the performances and complexity of the extraction algorithm [24, 25]. In the various performed tests, the number of grain paths does not influence the calculation time since it is much lower than the number of points in the spatial grid.

Dimensions	length x	width y	height z
Intervals [*mm*]	$[-0.5, 0.5]$	$[-0.5, 0.5]$	$[-0.2, 0.04]$
Grid step [*mm*]	0.02	0.02	0.001
Number of points	626841		

Table 1: Workspace characteristics

The simulations can be run several times successively. The number of lines n_k in one set and the height of the lines z_k of each line set W_{l_k} can be changed from one step to another to model evolution of the distribution of the grains. After surface extraction, the main characteristics z_{mean} and S_a are exported at each step. The abrasion simulation has been done with the three types of abrasive paper presented in Table 4.

4.3. Number of paths influence

The proposed approach of constant height has been studied to evaluate the numerical limit of the model under high number of paths. The functional characteristics enhancing is the result of multiple grain paths. These paths have been generated according to the algorithm presented in section 4.1. In a first approach, the height z_k of the grains has been fixed. Fig. 19 shows the evolution of the surface topography at the beginning of the simulation. An increased radius of the grains leads to deeper scratches and the depth of the paths is more important at intersections. This is due to the fact that the simulated surface is the result of a summation over the abrasive field, thus the value of the scalar field is more impacted at lines intersections locations.

Given an initial field F_{t_0}, the influence of the number of paths is studied. The evolution of surface roughness is plotted in Fig. 14. The repetition of paths first generates scratches, then the peaks are worn progressively while the depth of the valleys remains constant. As a consequence, the surface texture is damaged in the early stage of the simulation and the maximum surface roughness value is related to the grain size. While the number of lines reached increases, the surface roughness tends to zero whether the size of the grains or the relative surface height. As great numbers of paths are reached, the distribution leads to a uniform penalization field. Thus the resulting surface tends toward a smooth plane which explains the tendency of the surface roughness toward zero. The final numbers of paths are listed in Table 2.

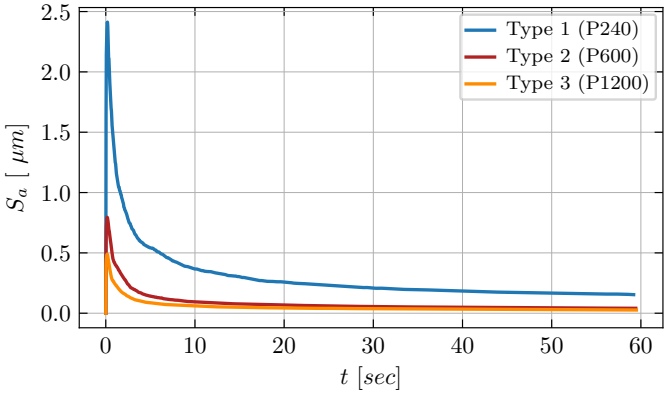

Figure 14: Surface roughness resulting from constant height paths

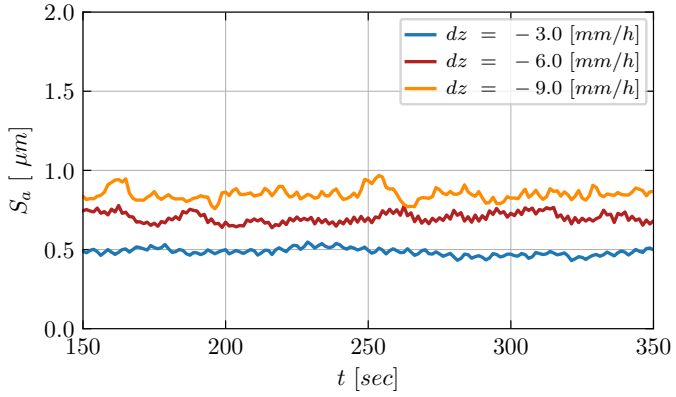

Figure 15: Surface roughness

Constant height gives unrealistic results while a certain number of lines are reached because it does not allow to reproduce the stagnation of surface roughness [26]. Experimentally, it is not possible to obtain mirror finish surface with low surface roughness using coarse abrasive tools.

Abrasive types	Type 1	Type 2	Type 3
Total paths number	23675	59429	118996

Table 2: Final number of paths at the end of the simulations

4.4. Abrasion rate influence

In real abrasion process, the contact pressure is imposed so that the grain path is established according to the pressure exerted and the surface in contact. As the surface wears, the abrasive tool follows the surface and the grains descend. Since the model presented is based on geometry, it was first chosen to emulate this behavior by varying the height of the paths. The relevance of this path generation strategy to simulate abrasion is questioned in this section.

One grain size has been fixed for all the simulations (Type 1). The required input value for the simulations is the abrasion rate according to number of paths. Regarding abrasion experiments, this rate is generally given in $mm.h^{-1}$, thus it has to be converted in mm per grain.

$$\delta_z = \frac{1}{\phi_g}\frac{dz}{dt} \tag{19}$$

$$\phi_g = V.J_g.l \tag{20}$$

Eq. 19 defines the rate of descent δ_z per grain path. ϕ_g represents the flux of grains over the surface. The flux of grain is defined as the time derivative of the grain path number function (Eq. 18). Thus, the flux can be computed as the product of the relative velocity V, the grit paper density J_g and the length of the domain l. Using the flux of grains over the surface, the Preston wear rate can be related to the rate of descent δ_z per grain. This allows to link process parameters to number of paths for the generation of line set Wl_k.

Three descent rates have been studied and are listed in Table 3. Fig. 15 shows the evolution of surface roughness for the three different descent rates. The surface roughness does not tend to zero with this approach because the surface is continuously worn. Considering one grain size, increasing the abrasion rate leads to a higher surface roughness value. Within this approach, peak are flatten yet new valley are created from lower paths. This generates a constant texture state over time. The relation between surface texture and the rate of descent is coherent with a geometrical analysis of the problem.

Descend rate	Case 1	Case 2	Case 3
dz [mm/h]	−3.0	−6.0	−9.0
δ_z [$mm/grain$]	$-2.08.10^{-6}$	$-4.16.10^{-6}$	$-6.24.10^{-6}$

Table 3: Descent rate equivalence

4.5. Grain size influence

The overall polishing process is composed of severals steps. Given an abrasive size, experimental investigations showed the surface roughness stagnates at a certain value [26]. Thus, the required final surface roughness is obtained by several operations with increasingly finer grit size. In the simulation, this behavior is modeled by the succession of line sets L_k with different kernel parameters.

In this section the influence of grain size is evaluated. Although the grain kernel parameters have been set once for each type of abrasive paper. It would be possible to define the kernel for each grain independently. The simulations have been run successively without reinitializing the surface when grain size are changed. The descent rate have been fixed for every cases at $-20\ mm/h$. The number of grain paths is increased until the surface roughness stagnates, then the grit size is changed. The evolution of S_a is plotted in Fig. 16. The surface roughness stagnates at three different levels. The evolution of the simulated surface condition is comparable and coherent in general form to the experimental results obtained by roughness measurements

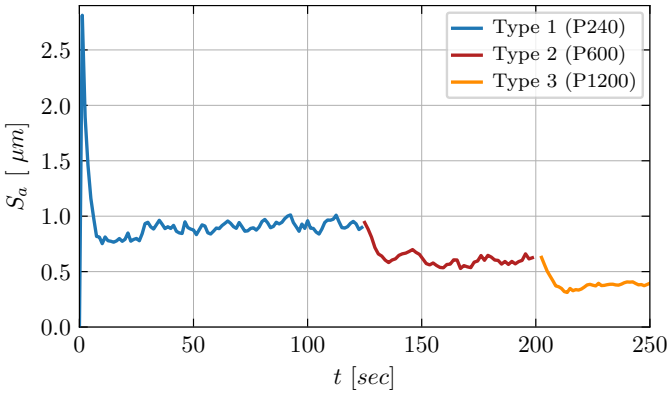

Figure 16: Simulated surface roughness for an abrasion process with three types of abrasive paper

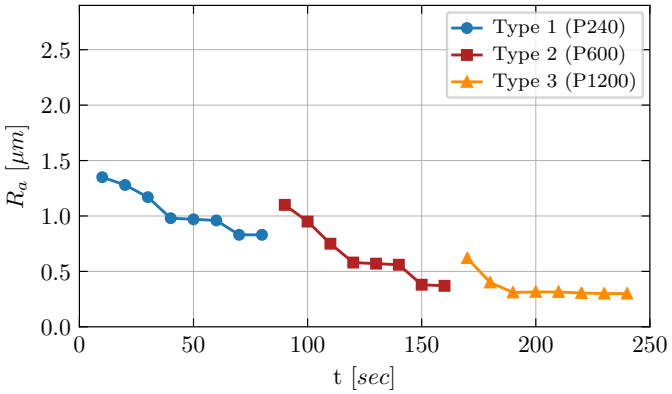

Figure 17: Evolution of Ra according to the time [26]

(Fig. 17). Finer grain radius allows to obtain smoother surfaces while coarse paper allows to remove the greater defects. Differences remain between the simulation and the experimental results, probably due several physics phenomena, such as abrasive wear, grinding sludge, possible reduction of the removal rate according to time, which are not currently included in the simulation.

5. Conclusion

An implicit model of the abrasion process has been developed to simulate surface topography. This model proposes a geometrical definition of the interaction between the grains and the part. The characteristics of the implicit surfaces and the associated deformations allow to natively model the cutting and repelling phenomena observed during abrasion. Moreover, there are minor additional calculation costs involved in pushing the material on either side of the scratching mark. Relevant preliminary results have been obtained with the numerical implementation presented. Topographies of abraded surfaces of a few square millimeters were simulated under the action of hundreds of thousands of grains. Moreover, the implicit formulation allows to simulate the topography resulting from abrasion operations under industrially compatible time. The macroscopic surface roughness evaluation shows the surface becomes smooth if the number of paths is increased without reducing the relative height. Both grain size and relative distance definition during the process have an influence on the surface roughness. The proper physical evolution of the relative height remains under investigation as it is linked to the pressure contact between material and abrasive tool. Within the proposed kernel parametrization generated surface are smooth and sharp edges are not taken into account in the model. Modifications of the kernel formulation and the numerical scheme to update the scalar field are under investigations to address this question.

References

[1] A. Sullivan, H. Erdim, R. N. Perry, and S. F. Frisken, "High accuracy NC milling simulation using composite adaptively sampled distance fields", Computer-Aided Design 44(6) (2012) 522–536.

[2] M. Jachym, S. Lavernhe, C. Euzenat, and C. Tournier, "Effective NC machining simulation with OptiX ray tracing engine", The Visual Computer (2018) 1–8.

[3] D. Tost, A. Puig, and L. Pérez-Vidal, "Boolean operations for 3d simulation of CNC machining of drilling tools", Computer-Aided Design 36(4) (2004) 315–323.

[4] M. Takaffoli and M. Papini, "Finite element analysis of single impacts of angular particles on ductile targets", Wear 267(1-4) (2009) 144–151.

[5] M. Takaffoli and M. Papini, "Material deformation and removal due to single particle impacts on ductile materials using smoothed particle hydrodynamics", Wear 274-275 (2012) 50–59.

[6] H. Tönshoff, J. Peters, I. Inasaki, and T. Paul, "Modelling and simulation of grinding processes", CIRP Annals 41(2) (1992) 677–688.

[7] E. Brinksmeier, J. Aurich, E. Govekar, C. Heinzel, H.-W. Hoffmeister, F. Klocke, J. Peters, R. Rentsch, D. Stephenson, E. Uhlmann, K. Weinert, and M. Wittmann, "Advances in modeling and simulation of grinding processes", CIRP Annals 55(2) (2006) 667–696.

[8] F. Klocke, O. Dambon, and B. Behrens, "Analysis of defect mechanisms in polishing of tool steels", Production Engineering 5(5) (2011) 475–483.

[9] K. Z. Gahr, "Microstructure and wear of materials", volume 10 of Tribology Series. Elsevier (1987).

[10] X. Chen and W. B. Rowe, "Analysis and simulation of the grinding process. part II: Mechanics of grinding", International Journal of Machine Tools and Manufacture 36(8) (1996) 883–896.

[11] F. W. Preston, "The theory and design of plate glass polishing machine", J. soc. Glass technol 11 (1927) 214–256.

[12] S. Leroch, M. Varga, S. Eder, A. Vernes, M. R. Ripoll, and G. Ganzenmüller, "Smooth particle hydrodynamics simulation of damage induced by a spherical indenter scratching a viscoplastic material", International Journal of Solids and Structures 81 (2016) 188–202.

[13] A. Zahedi and B. Azarhoushang, "FEM based modeling of cylindrical grinding process incorporating wheel topography measurement", Procedia CIRP 46 (2016) 201–204.

[14] P. Koshy, A. Iwasald, and M. Elbestawl, "Surface generation with engineered diamond grinding wheels: Insights from simulation", CIRP Annals 52(1) (2003) 271–274.

[15] H.-Q. Chen and Q.-H. Wang, "A novel approach to simulate surface topography based on motion trajectories and feature theories of abrasive grains", The International Journal of Advanced Manufacturing Technology 99(5-8) (2018) 1467–1480.

[16] L. Barthe, N. A. Dodgson, M. A. Sabin, B. Wyvill, and V. Gaildrat, "Two-dimensional potential fields for advanced implicit modeling operators", Computer Graphics Forum 22(1) (2003) 23–33.

[17] A. Raviv and G. Elber, "Three-dimensional freeform sculpting via zero sets of scalar trivariate functions", Computer-Aided Design 32(8-9) (2000) 513–526.

[18] J. Bloomenthal and B. Wyvill, "Interactive techniques for implicit modeling", SIGGRAPH Comput. Graph. 24(2) (1990) 109–116.

[19] M.-P. Gascuel, "An implicit formulation for precise contact modeling between flexible solids", In Proceedings of the 20th annual conference

Parameters	V	J_g	l	σ_1	μ_1	α_1	σ_2	μ_2	α_2	a	b
	$[m.min^{-1}]$	$[grain.cm^{-2}]$	$[cm]$	$[\mu m]$	$[\mu m]$		$[\mu m]$	$[\mu m]$			
Type 1 (P240)	1	240	1	29.25	0	0.6	14.625	40.95	−0.018	1	0.02
Type 2 (P600)	1	600	1	12.9	0	0.6	6.45	18.06	−0.018	1	0.02
Type 3 (P1200)	1	1200	1	7.65	0	0.6	3.825	10.71	−0.018	1	0.02

Table 4: Kernel parameters for simulations

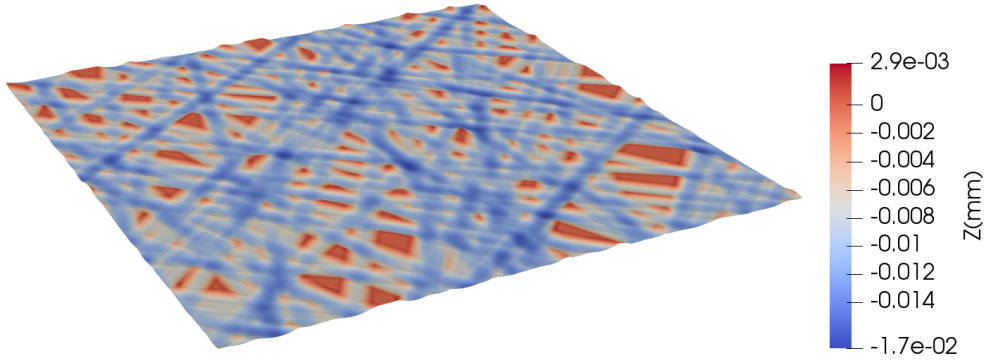

Figure 18: Simulation rendering

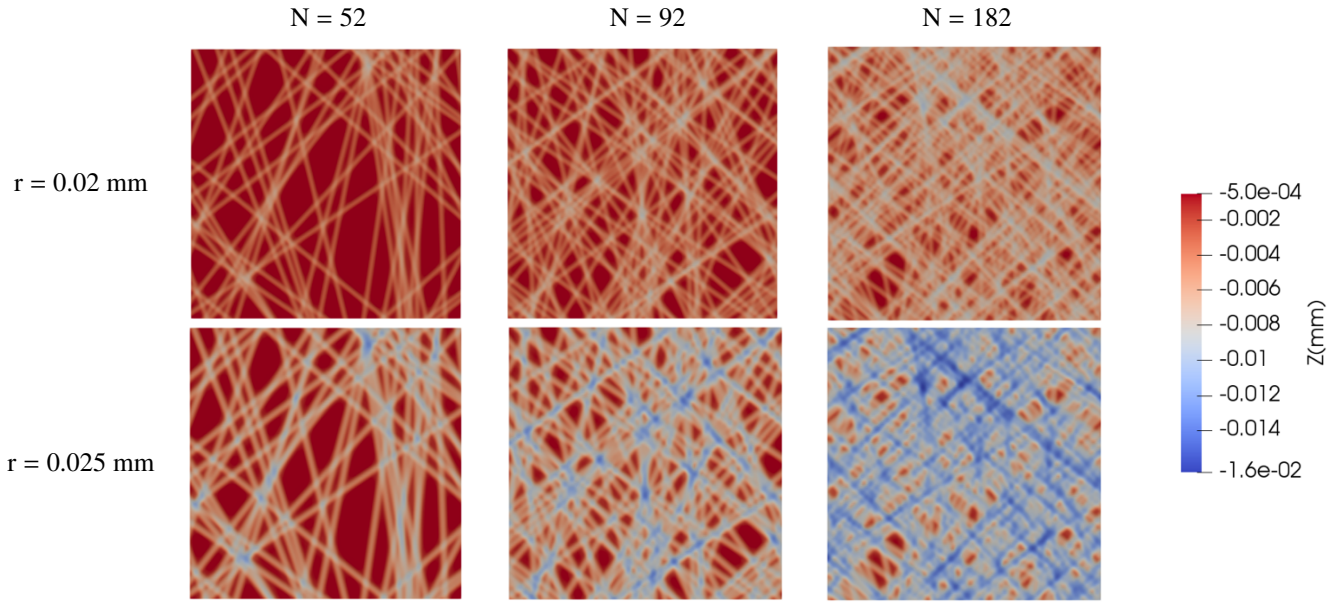

Figure 19: Surface examples for different grain radius and number of paths

on Computer graphics and interactive techniques - SIGGRAPH '93. ACM Press.

[20] B. Bastl and F. JeÅŸek, "Comparison of implicitization methods", Journal for Geometry and Graphics 9(1) (2005) 11–29.

[21] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm", In Proceedings of the 14th annual conference on Computer graphics and interactive techniques - SIGGRAPH '87. ACM Press.

[22] J. F. Blinn, "A generalization of algebraic surface drawing", ACM Trans. Graph. 1(3) (1982) 235–256.

[23] A. Klöckner, N. Pinto, Y. Lee, B. Catanzaro, P. Ivanov, and A. Fasih, "PyCUDA and PyOpenCL: A Scripting-Based Approach to GPU Run-Time Code Generation", Parallel Computing 38(3) (2012) 157–174.

[24] C. Andújar, P. Brunet, A. Chica, I. Navazo, J. Rossignac, and À. Vinacua, "Optimizing the topological and combinatorial complexity of isosurfaces", Computer-Aided Design 37(8) (2005) 847–857.

[25] N. Tatarchuk, J. Shopf, and C. DeCoro, "Real-time isosurface extraction using the GPU programmable geometry pipeline", In ACM SIGGRAPH 2007 courses on - SIGGRAPH '07. ACM Press.

[26] V. Lacharnay, C. Tournier, and G. Poulachon, "Design of experiments to optimise automatic polishing on five-axis machine tool", International Journal of Machining and Machinability of Materials 12(1-2) (2012) 76.