International Journal of CAD/CAM Vol. 5, No. 1, pp. 1~ 10 (2005)

INTERNATIONAL JOURNAL OF CAD/CAM
www.ijcc.org

On 5-Axis Freeform Surface Machining Optimization: Vector Field Clustering Approach

Chu A My*, Erik L J Bohez, Stanlislav S Makhanov[1], M Munlin[1], Huynh N Phien and Mario T Tabucanon
Industrial Systems Engineering Program, School of Advanced Technologies, Asian Institute of Technology, P.O box 4, Klongluang, Pathumthani 12120, Thailand
[1]Department of Information Technology, Sirindhorn International Institute of Technology, Pathumthani 12121, Thailand

Abstract – A new approach based on vector field clustering for tool path optimization of 5-axis CNC machining is presented in this paper. The strategy of the approach is to produce an efficient tool path with respect to the optimal cutting direction vector field. The optimal cutting direction maximizes the machining strip width. We use the normalized cut clustering technique to partition the vector field into clusters. The spiral and the zigzag patterns are then applied to generate tool path on the clusters. The iso-scallop method is used for calculating the tool path. Finally, our numerical examples and real cutting experiment show that the tool path generated by the proposed method is more efficient than the tool path generated by the traditional iso-parametric method.

Keywords: **5-axis CNC machining, Tool path optimization, Vector field clustering, Iso-scallop**

1. Introduction

Nowadays, products used in aerospace, automotive and shipbuilding industries are becoming more and more specialized and complicated to meet increasing demand of customers. The products, composed of various kinds of raw material, may include numerous complex freeform surfaces for fine features and functional requirements. To manufacture those surfaces, 5-axis CNC machine has been proven to be the most efficient tool.

Five degrees of freedom (5-axis) are the minimum *required to obtain maximum flexibility in tool-workpiece orientation. This means that the tool of the machine and the workpiece can be oriented relative to each other under any angle. In theory, the tool tip is able to contact with the workpiece at any point without changing the current setup status. This not only reduces the number of setup times but also increases much machining accuracy. Besides, innovations in the fields of mechanical engineering and CAD/CAGD/CAM have also enhanced the involvement of the machines in manufacturing such the products.

Devoted to the development of the applications of the machine, there has been a massive research work that deals with practical as well as theoretical issues. Topics of practical applications that are dealt with include the tool path generation, the post processor, the error compensation, the cutter selection, etc. In particular, the optimization of the tool path needs further researches to improve the machining efficiency and the productivity.

Optimization of a tool path of the 5-axis milling machine could be performed with regard to the cutting time, the scallop height, adherence to the required surface, the surface roughness, the length of the tool path, etc. The optimization invokes constraints related to the scallop heights, local and global accessibility, the machine range, etc. The independent variables characterize the tool positions and orientations, a way the orientations are being achieved through the rotations, the shape and the size of the tool, etc. Usually the tool visits the prescribed positions following the zigzag or the spiral pattern. However, the optimization could also utilize complicated patterns adapted in such a way that a certain cost function is minimized or at least decreased.

Given the general context above we consider an optimization with regard to the length of the tool path, namely,

$$\underset{\Theta}{\text{minimize}}(L) \tag{1}$$

subject to

$$h \le h_0 \tag{2}$$

$$|\varepsilon| \le \varepsilon_0 \tag{3}$$

where Θ is the tool path represented by a structured set of the tool positions and tool orientations (the so called CL data), h the scallop height and h_0 the maximum allowed scallop height. The scallop height can be defined as the height of material left between two adjacent cutter paths. Note that if the adjacent machining strips do not overlap, the remaining areas are considered as

*Corresponding author:
Tel: +66 – (0)2-524-6606
Fax: +66 – (0)2-524-5189
E-mail: mychuanh@yahoo.com, or st028278@ait.ac.th

scallops as well. The first constraint (2) controls the scallops and provides that the machining strips cover the entire surface. The second constraint (3) requires that the error of the linear interpolation ε does not exceed a prescribed value ε_0

The minimization of L is performed in such a way that Θ maximizes the width of the machining strip. In turn, increasing the machining strip makes it possible to cut the surface by traveling along a shorter path.

A variety of methods for 5-axis tool path optimization have been introduced in the Literature. The most popular are the iso-planar [11, 28, 29], the iso-parametric [6, 13, 18, 19] and the iso-scallop approach [10, 24, 26, 27]. More sophisticated but time consuming techniques include the elliptic grid generation method [1, 4], the dynamic cutter inclination approach [5, 21], the neural network modeling approach [31], the Voronoi diagram technique [32] and space filling curves [33].

The idea to decompose the part surface into sub-regions having similar characteristics has been initiated in [12, 22]. The curvature is used to decompose the surface into concave, convex and flat regions.

The potential field approach [9] has been introduced to decompose the surface into sub-regions where the zigzag pattern is applied to generate the tool path [9]. characterizes every point on the surface by an optimal feed direction vector. The vectors constitute a piecewise continuous vector field which could be further analyzed. There exists a number of ways to define the optimal direction, in particular, it can be chosen in such a way that it maximizes the width of the machining strip [9, 15, 24]. The choice of the optimal direction depends on the local principal curvatures of the surface and the cutter parameters. Furthermore, a tool path along the streamlines of the vector field maximizes the machining strip and therefore minimizes or at least decreases the path length. [9] introduces an "initial" tool path which has the largest averaged machining strip. The zigzag tool path is constructed by propagating the initial path inside the region while it does not substantially deviate from the streamlines. Another new initial tool path is searched when the ratio between the length of the path and the average machining strip is less than a certain threshold. However, a complicated surface produces a complicated, non uniform vector field. So, such analysis may not be accurate from the viewpoint of the global optimization since it may be sensitive to local variations in the optimization criteria. In that case, not only the zigzag pattern should be taken into account but also the spiral tool path should be considered. Moreover, the search for the best initial path constitutes the shortest path, which is NP hard problem. Therefore, a more systematic approach to the surface partition should be proposed

This paper extends the above ideas by introducing a new technique to partition the surface into clusters having the streamlines of the vector fields similar to the conventional zigzag or spiral patterns. The advantage of the proposed approach is that within the cluster the tool always follows the near optimal direction. Clustering provides a better criteria of how the decomposition should be performed since it characterizes the subsurface in the global sense. Clustering makes it possible not only to decompose but to recognize tool path patterns such as the zigzag, and the spiral pattern as well. Finally, since the vector field is piecewise continuous the spatial positions of the CC points (cutter contact points) in the vector field are calculated in such a way that the scallop height constraint is satisfied as well.

The proposed clustering is based on the spectral normalized cut method [34, 35] combined with the analysis of the streamlines and the identification of spiral centers [38, 39, 40] to recognize patterns. The iso-scallop method is employed to calculate the final tool path.

We also propose a heuristic procedure to connect the tool paths from each sub regions into a continuous curve without interruptions. However, when the procedure is not applicable the tool retractions must be included. Finally, we consider clustering with reference to the traditional iso-parametric schemes. The numerical experiments complemented by the real machining show that the proposed procedure is more efficient.

In this paper, the free form part surface, $\mathbf{S}(u,v)$, is assumed to be regular, and its parametric equation is represented in NURBS form [17]. The cutter surface Π_c follows the APT standard [2]. The cutter parameters R_1, R_2, β_1 and β_2 are illustrated in Appendix A (Fig. A1).

2. Vector Field of Optimal Cutting Directions

At a given CC point, we can rotate the tool around the surface normal vector 360° and get an infinite number of cutting directions. Cutting in the different directions produces different strip widths because of the normal curvature changing around the CC point (except umbilical point). In fact, the strip width not only depends on the cutting direction but also on the cutter parameters and the tool axis inclination. Therefore, at the CC point, an optimal cutting direction corresponding to the maximum strip width exits. As for the whole surface, a vector field of the optimal cutting directions exits obviously.

This section presents the cutting profile and the strip width determination, the local mill-ability, and an algorithm for searching the optimal directions.

2.1. Cutting profile and machining strip width

We consider the case of the cutter at a given CC point O (Fig. 1), where

- **n**: the unit normal vector of the part surface;
- **f** : the unit feed direction vector;
- $O_s = (O, x_s, y_s, z_s)$: *local part coordinate system*; z_s axis points in the normal vector direction; x_s axis

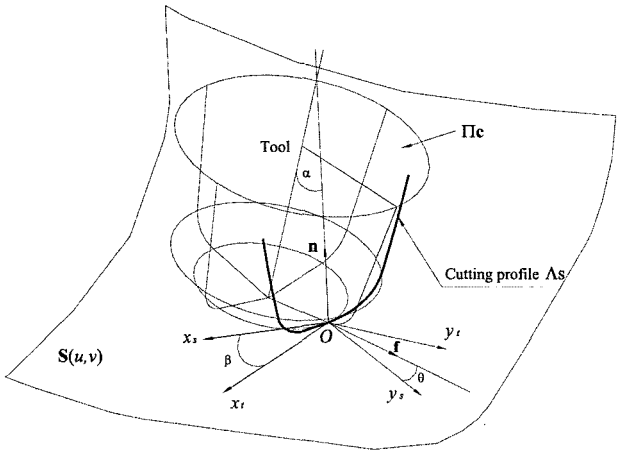


Fig. 1. Relative position between the tool Π_c and the part surface $S(u,v)$.

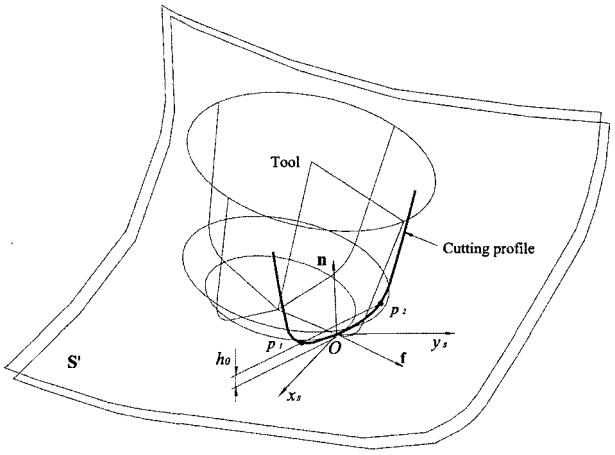


Fig. 2. Machining strip width.

and y_s axis point in the maximum and minimum principal directions of the part surface, respectively;

- $O_t = (O, x_t, y_t, z_t)$: *local tool coordinate system*; z_t axis points in the normal vector direction; x_t axis points in the principal direction which is tangent to the cutter surface parallel through the CC point; y_t is determined by the right hand rule;
- α : *tilt angle*, the angle between the tool axis and **n**;
- β : *roll angle*, the angle between x_s and x_t;
- θ: *cutting direction angle*, the angle between **f** and y_s.

We follow Jensen *et al.* [20] to define the tool orientation (α, β) and the feed direction **f**.

Recently, several definitions of the cutting profile have been used. The cutting profile is defined as

- the bottom circle of the flat end cutter [21],
- the projection of the bottom circle onto the normal plane (the plane through the CC point and normal to the feed direction) [5, 18],
- the circular arc approximated from the projected ellipse [24], or
- the intersection between the cutter surface and the normal plane [7].

The last one is flexible when using general APT tool. So we use this for formulating the cutting profile. If we denote Λ_s for the cutting profile, according to the selected definition Λ_s is obtained by solving:

$$\Pi_s \cdot \mathbf{f} = 0, \tag{4}$$

where Π_s is the parametric equation of the cutter surface represented in the coordinate system O_s. Π_s is transformed from the original parametric equation Π_c [2] to O_s through O_t.

The machining strip width Ω is determined as the distance between two intersection points p_1, p_2 of the cutting profile and the part surface offset (see Fig. 2). The offsetting distance equals to the scallop height limit h_0. p_1 and p_2 are obtained by solving:

$$\Lambda_s = \mathbf{S} + h_0.\vec{\mathbf{n}} \tag{5}$$

and

$$\Omega = |p_1 p_2| \tag{6}$$

Solving Eq. (5) requires a scheme for computing the NURBS offset, such as the one presented by Piegl & Tiller [30]. However, to reduce the computational time we use the approximation of the surface region surrounding the CC point, S': $z_s = 1/2(K_1 x_s^2 + K_2 y_s^2)$, where K_1 and K_2 are the maximum and minimum principal curvatures of the part surface at the CC point. So Eq. (5) is rewritten:

$$\Lambda_s = \mathbf{S}' + h_0.\vec{\mathbf{n}} \tag{7}$$

Λ_s, S' and h_0 are known, so p_1, p_2 are obtained by Eq. (7), and Ω is obtained by Eq. (6).

2.2. Local mill-ability

The local mill-ability means that there is a neighborhood of the CC point such that the solid bound of the cutter and the part surface do not have a point in common in this neighborhood except the CC point; and both the part and the cutter surfaces do not have second order contact at the CC point. So in identifying the optimal directions, the local mill-ability must be considered.

By using the Dupin indicatrices of the cutter surface and the part surface, Yoon *et al.* [23] proved that the part surface, **S**, is locally millable by the cutter surface, Π_c, at a given CC point, if

$$k_1 + k_2 > K_1 + K_2 \tag{8}$$

and

$$(k_1 - K_1)(k_2 - K_2) > \cos^2\beta (k_1 - k_2)(K_1 - K_2) \tag{9}$$

where,

$$k_1 = \frac{1}{R_2} \tag{10}$$

$$k_2 = \frac{\sin\alpha}{R_1 - R_2\cos\beta_2 + R_2\sin\alpha} \tag{11}$$

k_1 and k_2 are the principal curvatures of the APT cutter surface at the CC point. k_1 and k_2 are defined along y_t and x_t, respectively (see Fig. 1). Eq. (10) and Eq. (11) are obtained by extending the formulations of the principal curvatures of the fillet end cutter presented in [23]. Let $f_1 = R_2$ and $f_2 = R_1 - R_2 \cos\beta_2$. Substituting f_1 and f_2 in Eq. (8), Eq. (9) and then combining them, we obtain:

$$\sin\alpha > \frac{f_2(K_1-K_2)}{(1-f_1K_1)(1-f_1K_2)}\cos^2\beta + \frac{f_2K_2}{1-f_1K_1} \tag{12}$$

Eq. (12) shows that the local mill-ability of a CC point relates to the local principal curvatures of the part surface, the cutter parameters and the tool orientation. Since $\cos^2\beta$ is always positive, all cutter positions are locally mill-able when $K_1 < 0$ and no cutter position is locally mill-able when $K_2 > max\{k_2\} = 1/(R_1 - R_2\cos\beta_2 + R_2)$. The mill-ability guarantees the existence of the neighborhood, but we cannot say anything about the neighborhood size only with the local geometry.

2.3. Algorithm for searching the optimal cutting directions

Finding the maximum strip width Ω_{max} at a CC point leads to a complex problem, since Ω depends on several factors: the cutting direction, the tool inclination and the cutter parameters. Instead, we propose in this section an algorithm for searching Ω_{max} and the respective optimal direction locally. We assume that the cutter parameters are pre-selected. As for the cutter selection issue the readers can refer to Refs. [6, 8, 20]. The algorithm consists of following steps:

1. Generate a grid on the part surface.
2. Check the mill-ability for all grid points by Eq. (12), and obtain all tilt angles α_i. α_i at grid point i is determined as:

$$\alpha_i = \sin^{-1}\left(\frac{f_2(K_1-K_2)}{(1-f_1K_1)(1-f_1K_2)} + \frac{f_2K_2}{1-f_1K_1}\right) \tag{13}$$

Eq. (13) is obtained from Eq. (12) by setting $\beta = 0^0$. The roll angle $\beta = 0^0$ means that the right hand side of Eq. (12) is maximized. After checking for all grid points, a tilt angle α is chosen as:

$$\alpha = Max\{\alpha_i\} \tag{14}$$

This value will be used for next steps in determining the optimal directions at all grid points.

3. With $\alpha = \alpha$, we rotate the tool around the CC point (vary the roll angle β) from -90^0 to 90^0 with step of 1^0. Corresponding to each roll angle β_i, we change the cutting direction **f** in such a way that the cutting direction angle θ is varied from -90^0 to 90^0 with step of 1^0 also. Corresponding to β_i and θ_{ij}, the machining strip width Ω_{ij} is computed by Eq. (6). Ω_{max} is then searched among Ω_{ij}. The cutting direction vector **f**(x, y, z) corresponding to Ω_{max} is the optimal cutting direction vector.
4. Repeat the step 3 for all grid points we obtain the vector field of the optimal feed directions.
   We denote the field as: **F**$(x, y, z) = (fx(x, y, z), fy(x, y, z), fz(x, y, z))^T$.

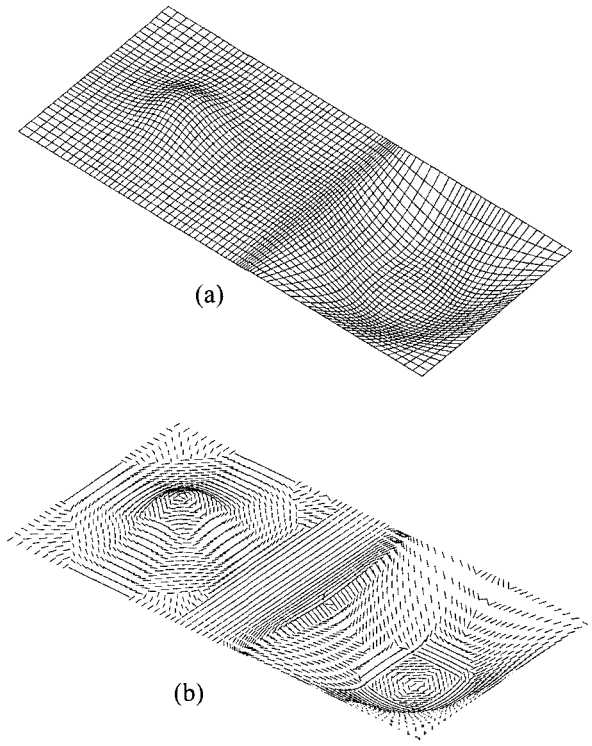


Fig. 3. Example. (a) NURBS surface, (b) Vector field of optimal cutting directions.

Fig. 3 show a surface and the field **F** generated with fixed cutter parameters (R_1 = 6 mm, R_2 = 1 mm, β_1 = 10^0, β_2 = 10^0).

Remark 1: The grid size reflects the likelihood (the similarity) between every two adjacent vectors. If the grid size decreases, the density of the field will increase and the likelihood will vary smoother. Accordingly, the weight matrix of the graph, which is set from the field, will contain very 'rich' information for clustering process. However, a big number of grid points will consume a long computational time. On the other hand, if a few grid points are generated, the weight matrix could reflect 'poor' information that could compact on clustering process though the computational time is shorten.

So a reasonable grid size is a compromise between the computational time and the effectiveness of the clustering process.

In this paper we select the parametric grid size as ($\Delta u \times \Delta v$), which are the parametric increments calculated by the iso-parametric method [13]. Δu is

determined providing that the tool path flow is selected in u direction, and Δv is determined providing that the tool path flow in v direction.

Δv is determined as

$$\Delta v = \frac{l}{S_v},$$

where $l = 2(h_0(2R_{tool} - h_0))^{1/2}$ is the side step, and S_v is the total arc length given constant u_c, that is the length of the curve $\mathbf{S}(u_c, v)$. Since S_v may vary as the parameter value u_c changes, the arc length S_v is calculated at $u_c = 0$, 0.25, 0.5, 0.75 and 1, respectively. The greatest value calculated is chosen as the value needed to achieve the given scallop height limit or less.

Δu is determined in similar way.

In the example above the increments are calculated as $\Delta v = 0.0134$, and $\Delta u = 0.0281$.

3. Vector Field Clustering

As introduced in section 1, the obtained vector field needs to be partitioned into influence sub-regions (clusters) on which a single tool path pattern (spiral or zigzag) can be recognized. The clustering research community [36, 37] has offered us several agglomerative and divisive algorithms. The algorithm based on the graph theory is practically efficient when used for image segmentation [34] and vector field clustering [35]. Unlike the most commonly used K-means, the normalized cut is a non-average technique that avoids potential problems caused by the K-means. When using K-means, the choice of the number of initial clusters (K) is crucial; quite different kinds of clusters may emerge when K is changed. Good initialization of the cluster centroids (average vectors) may also be crucial; some clusters may even be left empty if their centroids lie initially far from the distribution of the vector data. In particular, the normalized cut can be used to extract spiral clusters where the directions of the vectors are different, even opposite. The main operation of the normalized cut refers to solving a standard eigensystem that is not so time consuming compared to the other complex clustering methods such as the fuzzy clustering, the fuzzy K-means clustering, etc.

The normalized cut technique considers the field $\mathbf{F}$ needed to be clustered as a weighted graph G, where the nodes correspond to the vectors and the edges (connections) are formed between every pair of nodes. The weight on each edge, w_{ij}, is a function of similarity between node i and node j. The weights reflect the similarity between nodes. The matrix $\mathbf{W} = w_{ij}$ is called the weights matrix. We seek to partition the set of nodes (the graph) into disjoint sub-sets (groups) by measuring the similarity among nodes. The similarity within a sub-set is high and across different sub-sets is low. The technique gives a criterion to partition the graph into groups G_1 and G_2 as:

$$Ncut(G_1, G_2) = \frac{cut(G_1, G_2)}{assoc(G_1, G)} + \frac{cut(G_1, G_2)}{assoc(G_2, G)}, \tag{15}$$

where $cut(G_1, G_2) = \sum_{l \in G_1, k \in G_2} w_{lk}$, $assoc(G_1, G) = \sum_{l \in G_1, t \in G} w_{lt}$ is total connectionsfrom nodes in G_1 to nodes in entire G and $assoc(G_2, G)$ is defined similarly. *Ncut* is a criterion for measuring the goodness of a graph partition. So the problem is finding minimized value of *Ncut*. With this definition of disassociation between groups, if the $cut(G_1, G_2)$ minimizes and the $assoc(G_1, G)$ maximizes, the *Ncut* value will minimize. This minimization problem refers to the Rayleigh quotient, and finally to a standard eigensystem [34]:

$$\mathbf{C}^{-1/2} \cdot (\mathbf{C} - \mathbf{W}) \cdot \mathbf{C}^{-1/2} z = \lambda z, \tag{16}$$

where $c_i = \sum_{j=1}^{N} w_{ij}$ and $\mathbf{C} = \begin{bmatrix} c_1 & 0 & \dots & 0 \\ 0 & c_2 & \dots & \dots \\ \dots & \dots & \dots & 0 \\ 0 & \dots & 0 & c_N \end{bmatrix}$.

N: total number of vectors.

Constructing the weights matrix $\mathbf{W}$ is detailed in Appendix B.

Furthermore, $\mathbf{C}^{-1/2} \cdot (\mathbf{C} - \mathbf{W}) \cdot \mathbf{C}^{-1/2}$ is symmetric positive semidefinite since $(\mathbf{C} - \mathbf{W})$, also called the Laplacian matrix, is positive and semidefinite [34]. The real solution for the problem is the second smallest eigenvalue. The eigenvector of the second smallest eigenvalue is used as indicator to partition the graph. A similarly argument can also be made to show that the eigenvector with the third smallest eigenvalues is the real valued solution that optimally subpartitions the first two parts. In fact, this line of argument can be extended to show that one can subdivide the existing graph, each time using the eigenvector with the next smallest eigenvalue.

In short, following steps are used for clustering:

- Set up the weighted graph G from the vector field $\mathbf{F}$ (constructing the weights matrix $\mathbf{W}$).
- Solve the standard eigensystem: $\mathbf{C}^{-1/2} \cdot (\mathbf{C} - \mathbf{W}) \cdot \mathbf{C}^{-1/2} z = \lambda z$.
- Use the signs of the components of the second smallest eigenvector as the indicators to partition the vector data set. Vectors associated with the same sign are placed in the same cluster. Calculate the value of *Ncut* by Eq. (15).
- Recursively repartition the segmented parts if *Ncut* is below the pre-specified value.

The partitioning is controlled by the pre-specified value of *Ncut*. However, to improve the quality of a partition we introduce a refinement procedure based on the squared error criterion. The squared error is defined as:

$$SE = (f_{ix} - f'_{ix})^2 + (f_{iy} - f'_{iy})^2 + (f_{iz} - f'_{iz})^2, \tag{17}$$

where $\mathbf{f}_i = [f_{ix}\ f_{iy}\ f_{iz}]^T$: the original vector located at $\mathbf{x}_i = [x_i\ y_i\ z_i]^T$; and $\mathbf{f'}_i = [f'_{ix}\ f'_{iy}\ f'_{iz}]^T$: the approximated vector located at the same point $\mathbf{x}_i$. The approximated vector is determined by the first order Taylor expansion about a point $\mathbf{x}_c$:

$$\mathbf{f}_i' = \mathbf{A}(\mathbf{x}_i - \mathbf{x}_c) + \mathbf{f}_c, \tag{18}$$

where $\mathbf{f}_c$, located at $\mathbf{x}_c$, is a neighbor of $\mathbf{f}_i$, and $\mathbf{A}$ is the Jacobian matrix evaluated at $\mathbf{x}_c$ (see App. B for details).

There are two neighboring communities of a bordering vector (the vector nearest the border between two segmented children clusters). One is in the children cluster where it is located; the other is in the opposite children cluster.

The refinement aims at checking whether the bordering vector is more 'similar' to which one of the communities. If it is less 'similar' to the first community in the current cluster, it will be moved to the opposite cluster where it is more 'similar' to the second community.

After a partition, following steps are executed to refine the quality of the two children clusters:

- Each bordering vector $\mathbf{f}_i$ in the first cluster is approximated as $\mathbf{f'}_i$ by using Eq. (20); where $\mathbf{f}_c$ is chosen in the first neighboring community, and $\mathbf{A}$ is determined by the vectors collected in the first community also.
- Evaluate the squared error SE_1 between $\mathbf{f}_i$ and $\mathbf{f'}_i$ by using Eq. (19).
- The vector $\mathbf{f}_i$ now is approximated as $\mathbf{f''}_i$; where $\mathbf{f}_c$ is chosen in the second community, and $\mathbf{A}$ is determined by the vectors collected in the second community also.
- Evaluate SE_2 between $\mathbf{f}_i$ and $\mathbf{f''}_i$ by using Eq. (19).
- If $SE_1 > SE_2$ the vector $\mathbf{f}_i$ is moved to the second cluster.
- Repeat all the steps above until all the bordering vectors of the first clusters do not need to move.

The same statement is also applied for the second children cluster.

4. Spirals Detection

The spiral tool path pattern is seldom dealt with in the previous work in the literature, even in the researches following the idea of the surface decomposition [9, 12, 22, 25]. However, the spiral pattern plays an important role in improving the efficiency of the tool path planning for complex surfaces [1, 3, 4].

According to the current techniques of spiral center identification of the discrete vector field [38, 39, 40], the center of the spiral streamlines pattern is the point where the eigensystem of the Jacobian matrix $\mathbf{A}$ displays one real and one pair of complex-conjugate eigenvalues. The real eigenvector points in the direction about which the streamlines swirl. When all the clusters are determined, we search spiral center for each of the clusters as follows:

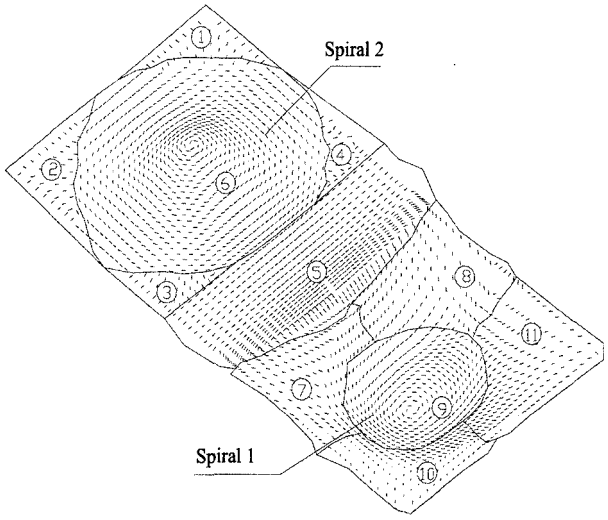


Fig. 4. Clusters with two spirals.

At every grid points of the concerned cluster the matrix $\mathbf{A}$ is evaluated locally (see App. B). The center of a spiral is the point that satisfies the criteria of one real and a pair complex-conjugate eigenvalues. We assume that the spiral center is unique; therefore, if the cluster contains more than one spiral center, it must be partitioned into smaller clusters. However, if there exist two adjacent grid points with one real and a pair complex-conjugate eigenvalues, we select the point nearest to the centroid. Furthermore, if the number of vectors in a certain cluster is too small it will join a neighboring. Fig. 4 shows an example of vector field which is partitioned into clusters and spirals.

5. Implementation and Example

We use the iso-scallop method [10, 24, 26, 27] for generating tool paths on the separated clusters. The iso-scallop tool path can be defined as a tool path which produces a constant scallop height across the machined surface [10, 24, 26, 27]. The main advantage of the iso-scallop method is that the redundant machining produced by the conventional method can be avoided. The spiral tool paths are generated on the spiral clusters, and the zigzag tool paths are generated on the other clusters. Machining sequence of the clusters is performed by following heuristic: the cluster nearest to the last CC point of the prior cluster is the next. This is carried out up to the last cluster. Also notice that, the clustering process produces a common border in between every two adjacent clusters. The machining boundaries of the corresponding sub-regions are determined based on the borders. The machining boundaries are calculated in such a way that if the tool cuts along the boundaries of two adjacent sub-regions, the scallop height does not exceed h_0.

We present one numerical example accompanied by

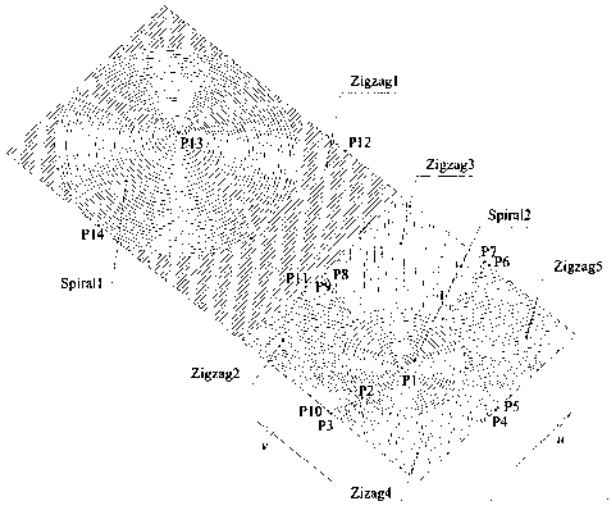


Fig. 5. CC path.

Fig. 6. Real cut part.

a corresponding real cutting experiment. The experiment is run on MAHO 600e 5-axis CNC machine. The cutter is selected with the parameters: $R_1 = R_2 = 5$ mm. $\beta_1 = \beta_1 = 0^0$. The CC path is shown in Fig. 5; and the real cut part is shown in Fig. 6. Information of the clusters is presented in Fig. 4 and Table. 1. Table. 2 shows the lengths of the CC paths of the presented method and the conventional method. The conventional CC paths are calculated in both directions: u and v. Although the presented method produces more turns the CC path length is reduced by 7.8% and 9.4%, when compared with the conventional CC path lengths in u and v directions, respectively.

Table 2. Comparisons of the 5-axis CC path lengths calculated by the presented method and by the conventional iso-parametric method (in both u and v directions). $h_0 = 0.1$ mm.

		CC path length (mm)	Number of turns
Presented method		18675.45	300
Conventional method	*In u Dir.*	20255.37	296
	In v Dir.	20616.16	152

Remark 2: As for the spiral cluster, the tool path propagates on the cluster area and reaches as near as possible to the border. No CC point is located out of the border. The common border between two zigzag regions is lincarized in the parametric domain. The cutting border of a zigzag region adjacent to another zigzag is determined such that if the tool cuts along two 'parallel' borders of the clusters, the scallop height left on the surface must satisfy the scallop height constraint. The cutting border of a zigzag region adjacent to the existing spiral tool paths is determined using the same constraint.

6. Conclusion

We have proposed a new method for 5-axis tool path optimization by clustering the optimal cutting direction vector field. The NURBS surface and the APT tool have been considered thoroughly in the optimization. By taking full advantage of the normalized cut clustering technique, a more systematic analysis is introduced to partition the complex surface into sub-regions where the spiral and the zigzag patterns can be applied to generate the nearly optimal tool paths. The combination of the spiral and the zigzag patterns produces more efficient tool path as compared with the traditional iso-parametric method (Table. 2). The proposed method can be used to improve and automate 5-axis sculptured surface machining for CAD/CAM systems. Analysis of more complex critical points such as in potential field fluid flow to derive more complex tool path patterns on more complicated surfaces should be the future work.

Table 1. Information of the machining clusters

Patterns	Clusters	Scallop (mm)	Number of turns	Machining sequence	Start point	End point
*Zigzag*1	1,2,3,4,5	0.1	130	6	P11	P12
*Spiral*1	6	0.1	0	7	P13	P14
*Zigzag*2	7	0.1	41	5	P9	P10
*Zigzag*3	8	0.1	40	4	P7	P8
*Spiral*2	9	0.1	0	1	P1	P2
*Zigzag*4	10	0.1	44	2	P3	P4
*Zigzag*5	11	0.1	45	3	P5	P6

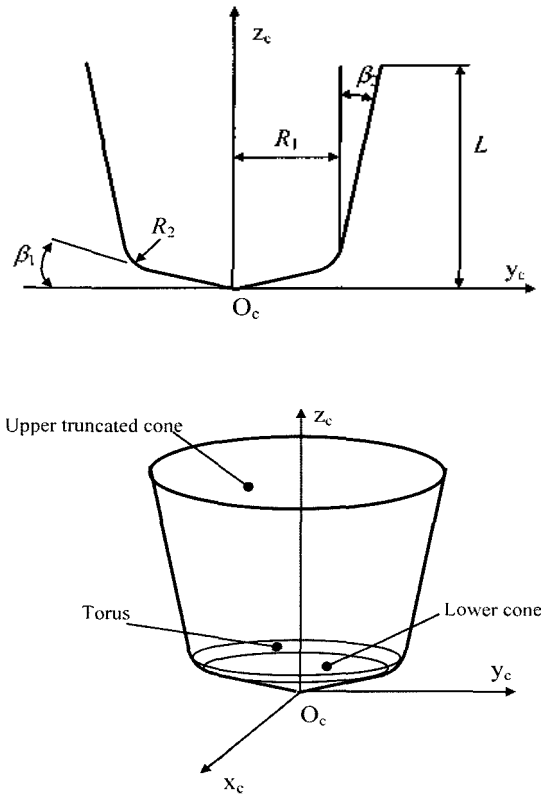


Fig. A1. Generalized APT cutter geometry.

Appendix A: APT cutter

Appendix B: Construct the weights matrix of the graph *G* from the vector field F

The weight w_{ij} between two nodes i and j (two vectors $\mathbf{f}_i = [f_{ix}\ f_{iy}\ f_{iz}]^T$ and $\mathbf{f}_j = [f_{jx}\ f_{jy}\ f_{jz}]^T$ located at $\mathbf{x}_i = [x_i\ y_i\ z_i]^T$ and $\mathbf{x}_j = [x_j\ y_j\ z_j]^T$ respectively) is calculated as [35]:

$$w_{ij} = \rho.e^{-dist_{ij}} + (1-\rho).e^{-diff_{ij}},$$

where $0 \le \rho \le 1$, $dist_{ij} = \sqrt{(x_i-x_j)^2+(y_i-y_j)^2+(z_i-z_j)^2}$

and $diff_{ij} = \sqrt{(f_{ix}-f_{jx})^2+(f_{iy}-f_{jy})^2+(f_{iz}-f_{jz})^2}$. (B.1)

The parameter ρ is also required to vary between 0 and 1. A small value of ρ emphasizes the difference in direction and magnitude while a large value of ρ places more weight on distance between vector locations.

To smooth the changing of the desirable weight values when the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ does not excess a predefined valued r, the vector $\mathbf{f}_j$ at $\mathbf{x}_j$ is replaced by an approximated vector $\mathbf{f}_j'$. $\mathbf{f}_j'$ is evaluated at the same point $\mathbf{x}_j$ through the first order Taylor expansion about the point $\mathbf{x}_i$: $\mathbf{f}_j' = \mathbf{A}(\mathbf{x}_j - \mathbf{x}_i) + \mathbf{f}_i$. The matrix **A** is the Jacobian matrix of the field $\mathbf{F}(x, y, z) = (fx(x, y, z), fy(x, y, z), fz(x, y, z))^T$ evaluated at $\mathbf{x}_i$ (The tensor of the deformation rate of the field).

$$\mathbf{A} = a_{ij} = \begin{bmatrix} \frac{\partial fx}{\partial x} & \frac{\partial fx}{\partial y} & \frac{\partial fx}{\partial z} \\ \frac{\partial fy}{\partial x} & \frac{\partial fy}{\partial y} & \frac{\partial fy}{\partial z} \\ \frac{\partial fz}{\partial x} & \frac{\partial fz}{\partial y} & \frac{\partial fz}{\partial z} \end{bmatrix} \quad \text{(B.2)}$$

The components of **A** are obtained by solving an over-determined system of equations. The system of equations is formulated by taking the Taylor expansion with m neighboring vectors around the vector $\mathbf{f}_i$. The neighboring vectors are denoted by $\mathbf{f}_k$ located at $\mathbf{x}_k$, where $k = 1, 2, \ldots, m$. The Taylor expansion: $\mathbf{f}_k = \mathbf{A}(\mathbf{x}_k - \mathbf{x}_i) + \mathbf{f}_i$.

Let $\Delta\mathbf{f}_k = \mathbf{f}_k - \mathbf{f}_i$ and $\Delta_k = \mathbf{x}_k - \mathbf{x}_i$, so $\Delta\mathbf{f}_k = \mathbf{A}\Delta_k$. This expression is used to formulate the over-determined system of equation:

$$\begin{bmatrix} \Delta x_1 & 0 & 0 & \Delta y_1 & 0 & 0 & \Delta z_1 & 0 & 0 \\ 0 & \Delta x_1 & 0 & 0 & \Delta y_1 & 0 & 0 & \Delta z_1 & 0 \\ 0 & 0 & \Delta x_1 & 0 & 0 & \Delta y_1 & 0 & 0 & \Delta z_1 \\ \vdots & & & & & & & & \vdots \\ \vdots & & & & & & & & \vdots \\ \vdots & & & & & & & & \vdots \\ \Delta x_m & 0 & 0 & \Delta y_m & 0 & 0 & \Delta z_m & 0 & 0 \\ 0 & \Delta x_m & 0 & 0 & \Delta y_m & 0 & 0 & \Delta z_m & 0 \\ 0 & 0 & \Delta x_m & 0 & 0 & \Delta y_m & 0 & 0 & \Delta z_m \end{bmatrix} \cdot \begin{bmatrix} a_{11} \\ a_{21} \\ a_{31} \\ a_{12} \\ a_{22} \\ a_{32} \\ a_{31} \\ a_{32} \\ a_{33} \end{bmatrix} = \begin{bmatrix} \Delta f_{1x} \\ \Delta f_{1y} \\ \Delta f_{1z} \\ \vdots \\ \vdots \\ \vdots \\ \Delta f_{mx} \\ \Delta f_{my} \\ \Delta f_{mz} \end{bmatrix} \quad \text{(B.3)}$$

References

[1] Bohez, E., Makhanov, S.S. and Sonthipermpoon, K. (2000), Adaptive nonlinear tool path optimization for five-axis machining, *Int. J. Prod. Res.*, **38**(17), 4329-4343.

[2] Bohez, E.L.J., Hong Minh, N.T., Ben Kiatsrithana, Peeraphan Natasukon, Huang, R.Y. and Le Thanh Son. (2003), The stencil buffer sweep plane algorithm for 5-axis CNC tool path verification, *Computer-Aided Design*, **35**(12), 1129-42.

[3] Makhanov, S.S. and Ivanenko, S.A. (2003), Grid generation as applied to optimize cutting operations of a five-axis milling machine, *Applied Numerical Mathematics*, **46**, 353-377.

[4] Makhanov, S.S., Batanov, D., Bohez, E., Sonthipaumpoon, K., Anotaipaiboon, W. and Tabucanon M. (2002), On the tool-path optimization of a milling robot, *Computers & Industrial Engineering*, **43**, 455-472.

[5] Choi, B.K., Park, J.W. and Jun, C.S. (1993), Cutter-location data optimization in 5-axis surface machining, *Computer-Aided Design*, **25**(6), 377-386.

[6] Lee, Y.S., Choi, B.K. and Chang, T.C. (1992), Cut distribution and cutter selection for sculptured surface cavity machining, *Int. J. Prod. Res.*, **30**(6), 1447-1470.

[7] Lee, Y.S. (1998), Mathematical modeling using different end mills and tool placement problems for 4 and 5-axis complex

surface machining, *Int. J. Prod. Res.*, **36**(3), 785-814.
[8] Lee, Y.S. and Chang, T.C. (1996), Automatic cutter selection for 5-axis sculptured surface machining. *Int. J. Prod. Res.*, **34**(4), 977-998.
[9] Chiou, C.J. and Lee, Y.S. (2002), A machining potential field approach to tool path generation for multi-axis sculptured surface machining, *Computer-Aided Design*, **34**, 357-371.
[10] Lee, S.G. and Yang, S.H. (2002), CNC tool path planning for high-speed high-resolution machining using a new tool-path calculation algorithm, *Int. J. Adv. Manuf. Technol.* **20**, 326-333.
[11] Xu, X.J., Bradley, C., Zhang, Y.F., Loh, H.T. and Wong, Y.S. (2002), Tool-path generation for five-axis machining of free-form surfaces based on accessibility analysis, *Int. J. Prod. Res.*, **40**(14), 3253-3274.
[12] Elber G. Freeform surface region optimization for 3-axis and 5-axis milling. (1994), *Computer-Aided Design*, **27**(6), 465-470.
[13] Loney, G.C. and Ozsoy, T.M. (1987), NC machining of free form surfaces, *Computer-Aided Design*, **19**(2), 85-90.
[14] Rao, N., Ismail, F. and Bedi, S. (2000), Integrated tool positioning and tool path generation for five-axis machining of sculptured surfaces, *Int. J. Prod. Res.*, **38**(12), 2709-2724.
[15] Marciniak, K. (1987), Influence of surface shape on admissible tool positions in 5-axis face milling, *Computer -Aided Design*, **19**(5), 233-36.
[16] Faux, I.D. and Pratt, M.J. (1979), *Computational Geometry for Design and Manufacturing*. Ellis Horwood Limited.
[17] Piegl, L.A. and Tiller, W. (1997), *The NURBS book, 2nd Ed.*, Springer.
[18] Wang, Y. and Tang, X. (1999), Five-Axis NC machining of sculptured surface, *Int. J. of Adv. Manuf. Technol.*, **15**, 7-14.
[19] Aekambaram, R. and Raman, S. (1999), Improved tool path generation, error measures and analysis for sculptured surface machining, *Int. J. of Prod. Res.*, **37**(2), 413-431.
[20] Jensen, C.G., Red, W.E. and Pi, J. (2002), Tool selection for five-axis curvature matched machining, *Computer-Aided Design*, **34**, 251-266.
[21] Kruth, J.P. and Klewais, P. (1994), Optimization and dynamic adaption of the cutter inclination during five-axis milling of sculptured surfaces, *Annals of the CIRP*, **43**, 443-448.
[22] Lauwers, B., Kruth, J.P. and Dejonghe, P. (2001), An operation planning system for multi-axis milling of sculptured surfaces, *Int. J. Adv. Manuf. Technol.*, **17**, 799-804.
[23] Yoon, J.H., Pottmann, H. and Lee, Y.S. (2003), Locally optimal cutting positions for 5-axis sculptured surface machining. *Computer-Aided Design*, **35**, 69-81.
[24] Lo, C.C. (1999), Efficient cutter path planning for five-axis surface machining with a flat-end cutter, *Computer-Aided Design*, **31**, 557-566.
[25] Kim, T. and Sarma, S.E. (2002), Toolpath generation along directions of maximum kinematics performance; a first cut at machine-optimal paths, *Computer-Aided Design*, **34**, 453-468.
[26] Feng, H.Y. and Li, H. (2002), Constant scallop-height tool path generation for three-axis sculptured surface machining, *Computer-Aided Design*, **34**, 647-654.
[27] Tournier, C. and Duc, E. (2002), A surface based approach for constant scallop height too-path generation, *Int. J. Adv. Manuf Technol.*, **19**, 318-24.
[28] Tam, H.Y., Xu, H. and Zhou, Z. (2001), Iso-planar interpolation for the machining of implicit surface, *Computer-Aided Design*, **33**, 307-319.
[29] Lartigue, C., Thiebaut, F. and Maekawa, T. (2002), CNC tool path in terms of B-spline curves, *Computer-Aided Design*, **34**, 453-468.
[30] Piegl, L. and Tiller, W. (1999), Computing offsets of NURBS curves and surfaces, *Computer-Aided Design*, **31**, 147-156.
[31] Suh, S.H. and Shin, Y.S. (1996), Neural network modeling for tool path planning of the rough cut in complex pocket milling, *Journal of Manufacturing Systems*, **15**, 295-324.
[32] Jeong, J. and Kim, K. (1999), Generating tool paths for free-form pocket machining using z-buffer-based Voronoi diagram, *Advanced Manufacturing Technology*, **15**, 182-187.
[33] Sarma, R. (2000), An assessment of geometric methods in trajectory synthesis for shape creating manufacturing operations, *Journal of Manufacturing Systems*, **19**, 59-72.
[34] Shi, J. and Malik, J. (2000), Normalized cuts and image segmentation, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **20**(8), 888-905.
[35] Chen, J.L., Bai, Z., Hamann, B. and Ligocki, T.J. (2003), A Normalized-cut algorithm for hierarchical vector field data segmentation, *Visualization and Data Analysis Proceedings*, 79-90.
[36] Kaufman, L. and Rousseeuw, P.J. (1990), *Finding groups in data*, John Wiley & Sons.
[37] Jain, A.K. and Dubes, R.C. (1988), *Algorithms for clustering data*, Prentice Hall.
[38] Jeong, J. and Hussain. (1995), On the identification of a vortex, *Journal of Fluid Mechanics*, **285**, 69-94.
[39] Kenwright, D. and Haimes, R. (1997), Vortex identification-applications in aerodynamics: a case study, *IEEE Visualization '99 Proceedings*, 413-416.
[40] Sujudi, D. and Haimes, R. (1995), Identification of swirling flow in 3D vector fields, *AIAA Computational Fluid Dynamics Proceedings*, 95-1715.

Chu Anh My is Doctoral Student of Industrial Systems Engineering Program at Asian Institute of Technology (AIT), Thailand. He obtained his Bachelor degree in Mechanical Engineering from Le Qui Don University of Technology, Vietnam in 1996, and Master degree in Manufacturing Systems Engineering from Asian Institute of Technology , Thailand in 2001. His research areas are CAD/CAM/CNC, Image Processing, and Geometric Modeling.

Huynh Ngoc Phien is Dean and Professor of the Computer Science Program, School of Advanced Technologies, Asian Institute of Technology (AIT), Thailand. He obtained his Bachelor Degree in Mathematics from Hue Institute, Vietnam in 1971, and PhD in Water Resource Engineering from Asian Institute of Technology, Thailand in 1978. His research area is Geometric Modeling, CAD/CAGD/CAM.

Stanislav S. Makhanov is Associate Professor of the Information Technology Program at Sirindhorn International Institute of Technology (SIIT), Thammasat University, Thailand. He obtained his Bachelor degree in M. Appl. Math. from Moscow State University, Russia and PhD in Applied Mathematics from Computer Center of the Russian Academy of Science, Russia. His research areas are Robotics, Image Processing, Grid Generation, and Computational Fluid Dynamics.

Mud-Armeen Munlin is Assistance Professor of the Information Technology Program at Sirindhorn International Institute of Technology (SIIT), Thammasat University, Thailand. He obtained his Bachelor degree in Physics from Prince of Songkhla University, Thailand in 1990 and PhD in Computer Science from School of Computer Studies, the University of Leeds, UK in 1995. He is teaching courses in Computer Graphics, Internet Technologies and Computer Science. His research area is CAD/CAM, solid modeling and CNC simulators.

Erik L J Bohez is Associate Professor of the Industrial Systems Engineering Program at Asian Institute of Technology (AIT), Thailand. He obtained his Bachelor Degree in Electro-Mechanical Engineer from Higher Technical Institute Saint Antonius Ghent, Belgium in 1976, Masters Degree in Electro-Mechanical Engineering from State University of Ghent, Belgium in 1977, and Degree of Professional Engineer in Electro-Mechanical Engineering from State University of Ghent, Belgium in 1979. His research area is CAD/CAGD/CAM/CNC.

Mario T. Tabucanon is Provost and Professor of Industrial Systems Engineering, Asian Institute of Technology. He has been a faculty member of AIT since July 1978 and has previously held the positions at AIT of Vice President (Academic), Dean of the School of Advanced Technologies, and Chairman of the then Division of Industrial Engineering and Management. He is credited with 200 publications in the areas of multiple criteria decision making, production management, and project management. He is on the Editorial Board of several international refereed journals in cluding the International Journal of Production EConmomics, Computers & Industrial Engineering, Manufacturing Systems, Operations & Quantitative Management, Engineering Valuation & Cost Analysis, Engineering Design & Automation, among others. He obtained his education from the Cebu Institute of Technology (B.S.E.E. and B.S.M.E.) and the Asian Institute of Technology (M.Eng. and D.Eng.).

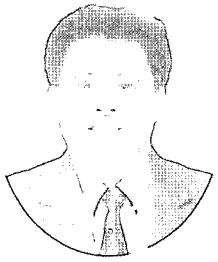
Chu Anh My

Erik L J Bohez

Stanislav S. Makhanov

Mud-Armeen Munlin

Huynh Ngoc Phien



Mario T. Tabucanon