

Article

A Method for Design of Modular Reconfigurable Machine Tools

Zhengyi Xu ¹, Fengfeng Xi ^{2,*}, Lilan Liu ³ and Li Chen ²

¹ RectorSeal Corporation, Houston, TX 77055, USA; zhengyi.xu@utoronto.ca

² Department of Aerospace Engineering, RyersonUniversity, Toronto, ON M5B2K3, Canada; chen.li@utoronto.ca

³ RoboticResearch Center, Shanghai University, Shanghai 200444, China; lancy@shu.edu.cn

* Correspondence: fengxi@ryerson.ca; Tel.: +1-416-979-5000

Academic Editor: Dan Zhang

Received: 10 November 2016; Accepted: 23 January 2017; Published: 4 February 2017

Abstract: Presented in this paper is a method for the design of modular reconfigurable machine tools (MRMTs). An MRMT is capable of using a minimal number of modules through reconfiguration to perform the required machining tasks for a family of parts. The proposed method consists of three steps: module identification, module determination, and layout synthesis. In the first step, the module components are collected from a family of general-purpose machines to establish a module library. In the second step, for a given family of parts to be machined, a set of needed modules are selected from the module library to construct a desired reconfigurable machine tool. In the third step, a final machine layout is decided through evaluation by considering a number of performance indices. Based on this method, a software package has been developed that can design an MRMT for a given part family.

Keywords: modular design; reconfigurable machine tools; module identification; module determination; layout synthesis

1. Introduction

Today's machines have come a long way evolving from the mechanization era to replace human labor, to the automation era to replace human operation, and now entering a new era called autonomization to replace human thinking. The trend is to autonomize machines by using advanced technologies such as embedded computers, smart sensors and intelligent controls, so that eventually machines will be able to operate on their own without human intervention. Modular Reconfigurable Machines (MRMs) represent the highest level of these intelligent machines because they are capable of changing their structures to adapt promptly to new circumstances, perform new tasks, and even recover from damage [1]. Apart from these functional advantages, MRMs are economically advantageous over conventional machines. Built based on modules that can be mass-produced, overall MRM cost is lower. In addition, the recurring cost and lead-time is minimized because the need for re-investment of completely new machines is eliminated [1].

Although a number of MRMs have been developed around the world, they are still mainly based on the methods for traditional machines, with some modifications to MRMs on a case-by-case basis. Therefore, a new theory must be developed in order to deal with the varying topological nature of MRMs. Research into MRMs is an emerging field with research activities taking place in two areas: robotics and manufacturing. In the robotics area, a number of interesting modular reconfigurable robots (MRRs) have been put forward. Though these MRRs were proposed on an ad hoc basis, they may be broadly classified into three categories: *self-assembly*, *self-configuring* and *manual-configuring*. Self-assembly robots have the highest level of reconfigurability because they can

automatically assemble and disassemble from their own modules or even grow by themselves [2]. Self-configuring robots, though not self-assembly, can perform self-reconfiguration after a robot system is assembled with some form of manual assistance [3]. Manual-configuring robots are the lowest level of reconfigurable robots, and they can only be reconfigured with manual assistance [4]. In terms of industrial applications, however, the last type is the most economical and robust due to its structure simplicity.

In the manufacturing area, primary efforts have been spent on the development of reconfigurable machine tools (RMTs) [5]. Kota [6] developed a methodology for synthesis of reconfigurable machine tools (RMT) using the screw theory for kinematics analysis and graph theory for machine structural synthesis. Tilbury and Kota [7] further developed a methodology for the systematic design of reconfigurable machine tools and the associated control systems. Moon and Kota [8] introduced a new screw representation for kinematic modeling of RMT. In their work, module selection was accomplished through decomposition of the required machining operations into a set of task matrices, which were then mapped to the machine modules. However, there is no clear guideline available for task matrix determination. As the number of machining operations increases, it becomes rather difficult, if not possible, to determine a minimum yet sufficient set of task matrices.

Modular design of machine tools has been studied for several decades. Shinno and Ito [9] proposed a methodology for generating the structural configuration of machine tools. They decomposed machine tool structures into simple geometric forms, such as plates, beams, and boxes. Their research contributed to the design of machine tools for Flexible Manufacturing Systems (FMS). An overview of machine tool modular design was provided by Garro and Martin [10]. Later on, Ouyang et al. [11] adapted Ito's method for modular machine tool synthesis and developed a method for enumerating machine tool modules. Zatarain et al. [12] proposed a new method of modular machine tool synthesis using predetermined component information. Two major benefits of adopting modular structures are standardization of components and reconfigurability of devices. In particular, the advantages of modular design can be summarized as: (1) device reconfigurability; (2) speedy introduction of new devices; (3) maintainability and serviceability of devices; and (4) design information reuse [13]. Though the traditional machine tool design was documented in [14] and modular design was documented in [15], these methods were not discussed in the context of reconfigurable machine tools.

To overcome these problems, a method is presented here for the systematic design of modular reconfigurable machine tools (MRMTs). This method consists of three steps: module identification, module determination, and layout synthesis. In the first step, the module components are collected from a family of general-purpose machines to establish a module library.

In the second step, for a given family of parts to be machined, a set of the needed modules are selected from the module library to construct a desired reconfigurable machine tool. In the third step, a final machine layout is decided through evaluation by considering a number of performance indices. Based on this method, a software package has been developed that can design an MRMT for a given part family.

2. Problem Formulation and Methodology

A modular reconfigurable machine tool (MRMT) is designed based on building blocks, called modules. In this design problem, the input is a set of parts to be machined, called part family, each part sharing some machining features but differing in others. The output is the final design of a reconfigurable machine tool that can use a minimal set of modules through reconfiguration to machine all the machining features within the part family. Figure 1 provides a block diagram of the proposed design system. As shown in Figure 1, when a given family of parts is input into the design system, a set of machining features will be extracted and related to machine modules based on tool motion requirement. This is the first step, called module identification. Then, a set of machine modules will be selected and evaluated to form a minimum set of modules. This is the second step, called module

selection. Finally, a final design will be determined based on a number of performance indices. This is the third step, called layout synthesis. In what follows, the details of this design system are provided.

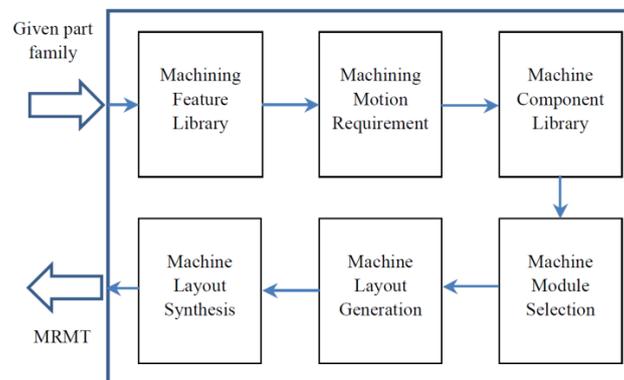


Figure 1. Design system for modular reconfigurable machine tools.

3. Module Identification and Selection

The theory for module identification and selection was reported in our previous paper [16]. For the completeness of the paper, a brief review is provided and then applied to MRMTs. The purpose of module identification is to establish two libraries and create a mapping between the two. The first library is called module component library, and the second one is called machining feature library. A mapping is created between the two libraries based on tool motion requirement.

3.1. Module Component Library

This library serves as a machine module database. It is established by considering all of the general-purpose machine tools including vertical and horizontal lathes as well as vertical and horizontal milling machines as shown in the first block of Figure 2. This consideration provides comprehensive module coverage for modular reconfigurable machine tool design. In our design system, function taxonomy is created for each machine in terms of two core functions: motion and support. Motion modules include rotary and linear motion systems; support modules include machine bed and column. In Figure 2, these modules are illustrated in the second block, their design attributes in the third block and motion requirement in the last block. A similar taxonomy structure can be created for actuators and controller based on machine tool control systems summarized in [17].

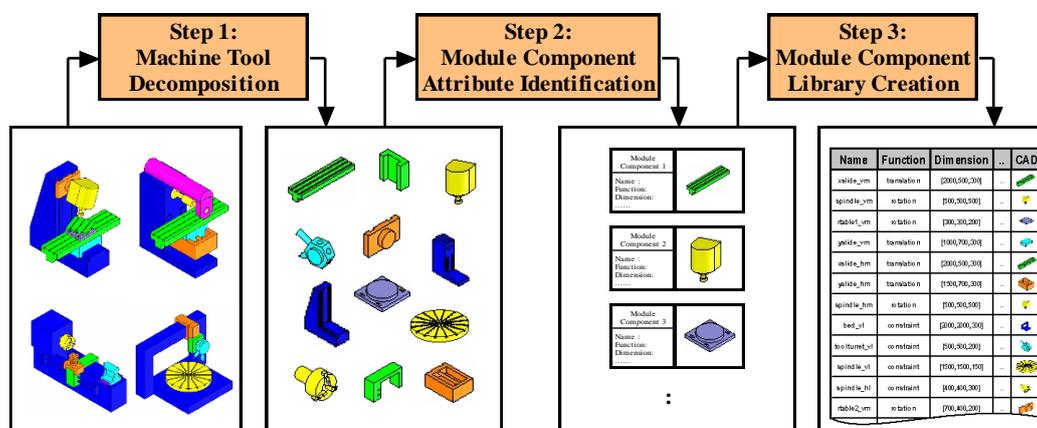


Figure 2. Module identification method.

3.2. Machining Feature Library

This library serves as a machining feature library. A machining feature identifies a volume of material that must be removed to obtain the final part geometry from the initial stock as well as the motion required to drive a tool to perform machining, commonly called tool approach direction (TAD). The machining process of a machined part can be represented by a set of machining features. The international standard STEP (Exchange of Product Model Data) [18] is used for this purpose. STEP AP 224 “Mechanical Product Definition for Process Planning Using Machining Features” defines a list of common machining features. Our machining feature library is built based on STEP. For machining features not included in STEP, feature recognition methods [19–22] can be applied to update our library.

3.3. Module Selection

For a given family of parts as shown in the first block of Figure 3, the associated machining features for each part will be extracted as shown in the second block and compared with the machining feature library shown in the third block to find the desired motion modules shown in the fourth block through motion matching, i.e., between TAD and machine motion module. Different parts will lead to different sets of modules shown in the fifth block.

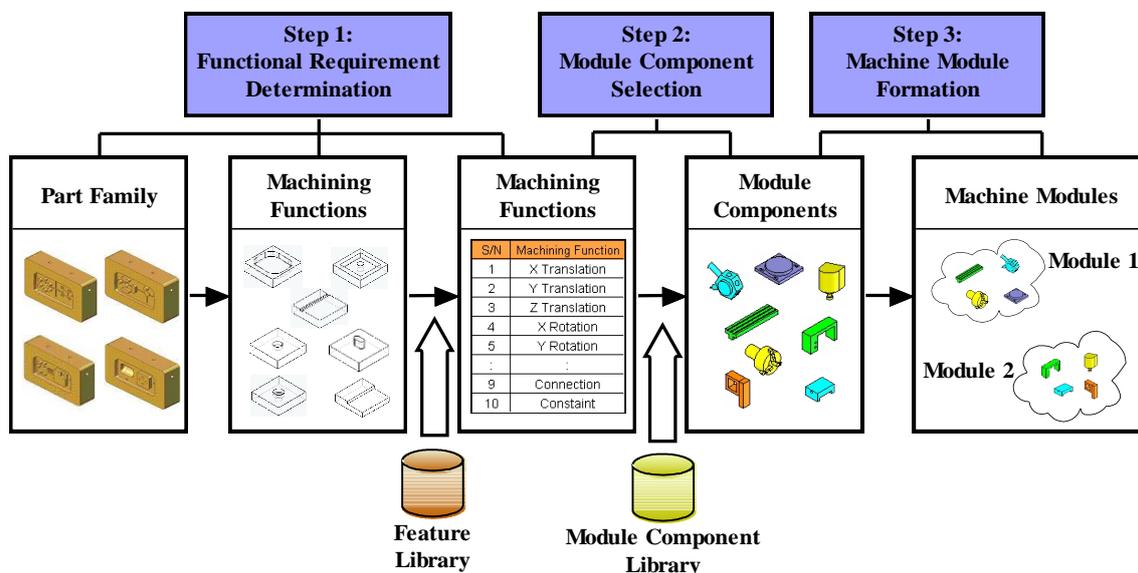


Figure 3. Flowchart of the module selection method.

As an illustrative example for an industrial application of mold making, Table 1 enlists the machining features that are extracted according to STEP for a family of four given parts. Part 1 includes five machining features: (i) two general pockets; (ii) 10 bosses within a pocket—cylindrical; (iii) two slots—rectangular profile; (iv) two blind holes—counter-bored; and (v) seven blind holes—cylindrical. Both parts 2 and 3 contain six machining features, respectively, and part 4 four machining features. All of these features can be lumped into seven machining features (MF) as noted in Table 2: MF1—“General Pocket”; MF2—“Boss within a Pocket—Cylindrical”; MF3—“Slot—Rectangular Profile”; MF4—“Blind Hole—Counter-Bored”; MF5—“Blind Hole—Cylindrical”; MF6—“Boss—General Profile”; MF7—“Step”; MF8—“Revolved Feature—Volume Removal”. Table 2 demonstrates how these machining features are related to the corresponding motion requirements, i.e., TAD. For example, MF1 (general pocket) is machined by mounting the spindle in z-axis and moving it in x-, y- and z-axis. For MF2 to MF8, their motion requirements are detailed in Table 2.

Table 1. Machining features extraction.

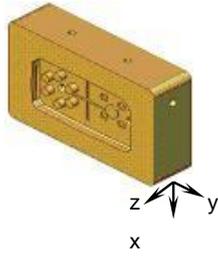
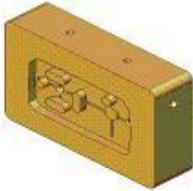
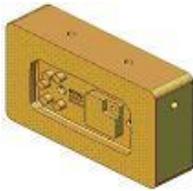
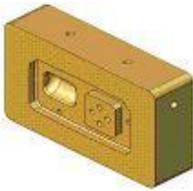
Part 1	Part 2	Part 3	Part 4
			
Machining Features:	Machining Features:	Machining Features:	Machining Features:
(1) General Pocket (2)	(1) General Pocket (1)	(1) General Pocket (1)	(1) General Pocket (1)
(2) Boss within a Pocket—Cylindrical (10)	(2) Boss—General Profile (3)	(2) Boss—General Profile (1)	(2) Boss—General Profile (1)
(3) Slot—Rectangular Profile (2)	(3) Boss within a Pocket—Cylindrical (3)	(3) Boss within a Pocket—Cylindrical (5)	(3) Revolved Feature—Volume Removal (1)
(4) Blind Hole—Counter-Bored (2)	(4) Slot—Rectangular Profile (2)	(4) Step (1)	(4) Blind Hole—Cylindrical (14)
(5) Blind Hole—Cylindrical (7)	(5) Blind Hole—Counter-Bored (2)	(5) Blind Hole—Counter-Bored (2)	
	(6) Blind Hole—Cylindrical (7)	(6) Blind Hole—Cylindrical (7)	

Table 2. Corresponding motion requirements.

Machining Feature	Spindle Axis	Motion Requirement
MF1	z	x-Translation, y-Translation, z-Translation
MF2	z	x-Translation, y-Translation, z-Translation
MF3	z	x-Translation, y-Translation, z-Translation
MF4	z	z-Translation
MF5	z	z-Translation
MF6	z	x-Translation, y-Translation, z-Translation
MF7	z	x-Translation, y-Translation
MF8	y	z-Translation

Note: MF1—“General Pocket”; MF2—“Boss within a Pocket—Cylindrical”; MF3—“Slot—Rectangular Profile”; MF4—“Blind Hole—Counter-Bored”; MF5—“Blind Hole—Cylindrical”; MF6—“Boss—General Profile”; MF7—“Step”; MF8—“Revolved Feature—Volume Removal”.

After analyzing the motion requirements, these machining features can be clustered into three machine configurations, as summarized in Table 3. Here, letter A represents an x -axis slide, B a y -axis slide, C a z -axis slide, D an x -axis rotation table, E z -axis spindle, and F a y -axis spindle. Configuration 1 is composed of three translations with the spindle along Z direction, i.e., (A, B, C, E). This is a three axis vertical milling machine configuration that can be used to machine MF1, 2, 3, 4, 6 and 7 for the case under study. Configuration 2 consists of one translation with the spindle along the y -direction, i.e., (C, F), i.e., a one-axis turning machine configuration that can be used to machine MF8. Configuration 3 is made up of two translations and one rotation with the spindle along z -direction, i.e., (A, C, D, E). This is a two axis milling machine configuration with a turning table that can be used to machine MF5.

Table 3. Clustered machine configurations.

	Part 1	Part 2	Part 3	Part 4
Configuration 1:	2 MF1	1 MF1	1 MF1	1 MF1
	10 MF2	3 MF6	1 MF6	1 MF6
	2 MF3	3 MF2	5 MF2	
	2 MF4	2 MF3	1 MF7	
		2 MF4	2 MF4	
	<i>x</i> -Translation	<i>x</i> -Translation	<i>x</i> -Translation	<i>x</i> -Translation
	<i>y</i> -Translation	<i>y</i> -Translation	<i>y</i> -Translation	<i>y</i> -Translation
	<i>z</i> -Translation	<i>z</i> -Translation	<i>z</i> -Translation	<i>z</i> -Translation
	Spindle Z	Spindle Z	Spindle Z	Spindle Z
Configuration 2:				1 MF8
				<i>z</i> -Translation, Spindle Y
Configuration 3:	7 MF5	7 MF5	7 MF5	7 MF5
	<i>z</i> -Translation	<i>z</i> -Translation	<i>z</i> -Translation	<i>z</i> -Translation
	<i>x</i> -Translation	<i>x</i> -Translation	<i>x</i> -Translation	<i>x</i> -Translation
	<i>y</i> -Rotation	<i>y</i> -Rotation	<i>y</i> -Rotation	<i>y</i> -Rotation
	Spindle Z	Spindle Z	Spindle Z	Spindle Z

where a numeral indicates the number of motions, for example, 2 MF1, indicates that there are two motions of MF1.

In our previous paper [16], the theory for module classification was provided to include common, similar, and singular module components. The common module components appear in all machine configurations. The similar module components appear only in some machine configurations. The singular module components appear in only one machine configuration. As shown in Figure 4, based on this theory, (C) is deemed a common module because it appears in all three configurations; (A, E) were deemed similar modules because they appear in configuration 1 and 3; (B, D, F) were deemed singular modules because each of them only appears in the respective configuration.

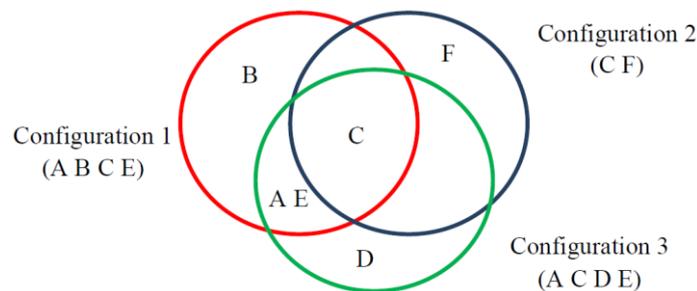


Figure 4. Determination of minimal set modules; where A—*x* axis slide, B—*y* axis slide, C—*z* axis slide, D—*x* axis rotation table, E—*z* axis spindle, F—*y* axis spindle.

The basic idea of a reconfigurable machine tool design is to retain all of the common modules as a base, while trying to reduce the number of singular and/or similar modules that will be required for different tasks. One may argue that by simply combining all six modules, the resulting machine tool will be able to machine all the features without reconfiguration. However, this is the conventional design approach of multifunction machines at the cost of extra machine tool axes that may not be needed when machining each part within the part family. The essence of a reconfigurable machine tool design is to find a minimum number of machine tool axes that can machine all the features of a part family. Back to the case under study, if by reconfiguring singular module E (*z*-axis spindle) to F (*y*-axis spindle), the total number of modules will be reduced to five. However, the other two singular modules (B, D) cannot be simply replaced through reconfiguration. Hence, (A, B, C, D, E)

represents a minimal set of modules for designing this MRMT. To summarize, common modules serve as a base, similar modules are used for reconfiguration design, and singular modules generally cannot be designed for reconfiguration due to their single usages.

4. Layout Synthesis

4.1. Layout Generation

After modules are selected, layout will be generated to form a desired reconfigurable machine tool. For this purpose, tree graph enumeration algorithms are adopted and the interested reader can refer to [23–28] for details. Steps are provided here: the first step is to enumerate all the generalized layouts by defining enumeration constraints, performing enumeration, and checking isomorphism. The enumeration constraints for generalized layout generation are: (1) number of vertices; (2) number of pendant vertices; and (3) maximum number of solutions. The number of vertices is set equal to the number of selected module components. The number of pendant vertices is set according to machining requirements.

The layout enumeration algorithm adopted is a growing algorithm for generating rooted-tree type layouts [28]. The algorithm is initialized by assigning vertex numbers, initial values for variables, and constraint constants, such as number of vertices, number of pendant vertices, and maximum number of solutions. At this point, there is only one vertex (the root vertex) in the tree graph. Then, a new vertex is loaded into the tree graph and solutions are generated by connecting the new vertex with the existing vertices. For example, when the second vertex is loaded into the tree graph, one solution can be created by connecting the second vertex with the root vertex. Loading vertices continue until all the possible solutions are generated or the number of solutions exceeds the maximum value.

When a new layout is generated, it is possible that the layout is isomorphic to one of the existing layouts. Therefore, isomorphism detection is performed during solution generation. There are several methods proposed in the literature for isomorphism detection [23–28]. The isomorphism detection algorithm presented in this paper is based on the tree method [23], which requires examining if the tree is rooted or not first. If the tree is not rooted, the center of the tree is determined and is taken as rooted. One way to find the center of a tree is to repeatedly remove all the vertices with connectivity 1 until either a single vertex or two vertices connected by an edge are left. A vertex is the center of the tree if it is the only one left. If there are two vertices left, there will be two separate centers. Once the trees are rooted, the isomorphism detection algorithm is applied. Either Breadth-First Search (BFS) or Depth-First Search (DFS) methods can be used for this purpose [26].

For the given example, there are five selected module components; hence, the number of vertices is 5, and two pendant module components are required, i.e., one for the spindle (E) and the other for the module component that holds the part to be machined, which can be any one of (A, B, C, D). Hence, the number of pendant vertices is 2. The result of enumeration is shown in Table 4.

Table 4. Results of generalized layout generation.

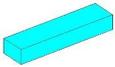
Solution	Tree Graph	Enumeration Constraints
1		Number of vertices = 5
2		Number of Pendant vertices = 2
3		Maximum number of solutions = 50

4.2. Module Assignment

After the generalized layouts are created, the next step is to convert generalized layouts into specialized layouts by replacing the vertices in the generalized layouts with the module components.

First, the module components are simplified for the purpose of illustration. As shown in Table 5, for example, the module component “x slide” can be simplified into a box. The function of this module component is “x-Translation”. There are two connecting ports on this module component. One of them is a prismatic port and the other is a fixed port. In this work, there are three different types of connecting ports considered: prismatic port, revolute port, and fixed port. Prismatic port refers to a connecting port whose coordinate system is able to move along its own z-axis for a distance with a specific range. Revolute port refers to a connecting port whose coordinate system is able to rotate about its own z-axis for an angle with a specific range. Fixed port refers to a connecting port whose coordinate system is fixed.

Table 5. Results of module component simplification.

Name	Original CAD Model	Simplified CAD Model	Function	Connecting Ports		
				Type	Location ($x, y, z, \theta_x, \theta_y, \theta_z$)	Parameters
xslide_			x-Translation	P	[1000, 250, 0, 0, 90, 0]	[-550, 550]
				F	[100, 250, 300, 90, 0, -90]	N/A
yslide_			y-Translation	P	[250, 1500, 500, -90, 0, -90]	[-300, 300]
				F	[250, 250, 500, 0, 90, 0]	N/A
zslide_			z-Translation	P	[250, 500, 1200, 0, 0, 90]	[-300, 300]
				F	[250, 250, 0, 90, 0, 90]	N/A
rtable1_			y-Rotation	R	[0, 0, 200, 0, 0, 0]	[-180, 180]
				F	[0, 0, 0, 90, 0, 90]	N/A
spindle_			Spindle	F	[250, 0, 250, 180, 0, -90]	N/A

P—“Prismatic Port”; R—“Revolute Port”; F—“Fixed Port”; Linear Unit: mm; Angular Unit: degree.

After the module components of the new machine tool are idealized and the key attributes of the module components are collected, the next step is to define module assignment constraints. As mentioned early, the module generation method proposed in this paper is based on enumeration. There may be some layout solutions that are not feasible according to the machining functional requirements or computational efforts. Therefore, proper constraints need to be set so as to reduce or eliminate the infeasible solutions. Listed below are some typical constraints for module assignment:

- Specify the modules that should be assigned to the pendant vertices;
- Specify a module that should be assigned to the root vertex;
- Specify some module components that should be connected with each other directly;
- Specify allowed port connecting patterns, such as Prismatic-Fixed (P-F), Revolute-Fixed (R-F), and Fixed-Fixed (F-F);
- Specify mobility requirements;
- Specify maximum number of module assignment solutions.

For the given example, the module assignment constraints are set as follows. The spindle module “spindle” is assigned to a pendant vertex. The module component “y-slide” is assigned to the root vertex. The module components “spindle” and “z-slide” are connected together directly. Only three port connecting patterns are allowed: P-F, R-F, and F-F. Other connecting patterns like P-P and R-R are not considered as they will result in mobility loss. The degree of freedom between the two pendant vertices is four. The maximum number of module assignment solutions is 100.

To generate specialized layouts, a module assignment procedure is developed that consists of two steps: (1) enumerate sequences of the module components (Module Component Sequences

Enumeration); and (2) enumerate port combinations between each pair of module components (Port Combinations Enumeration), as shown in Figure 5. The first step is to enumerate the connecting sequences of the target module components. The module components can be classified into groups according to some of the constraints to reduce the complexity of the problem. In this case, the module components are classified into: pendant module components, module components within the same module, and other module components. After module component classification, module sequence solutions are generated by applying sorting algorithms. There are many sorting algorithms available. In this paper, we use the sorting function provided in MATLAB. The second step is to enumerate the port combinations between each pair of the neighboring module components. Changing the port combination may result in mobility loss. Therefore, mobility check is performed to screen out those infeasible solutions. The module assignment produces a total of 15 solutions as listed in Table 6.

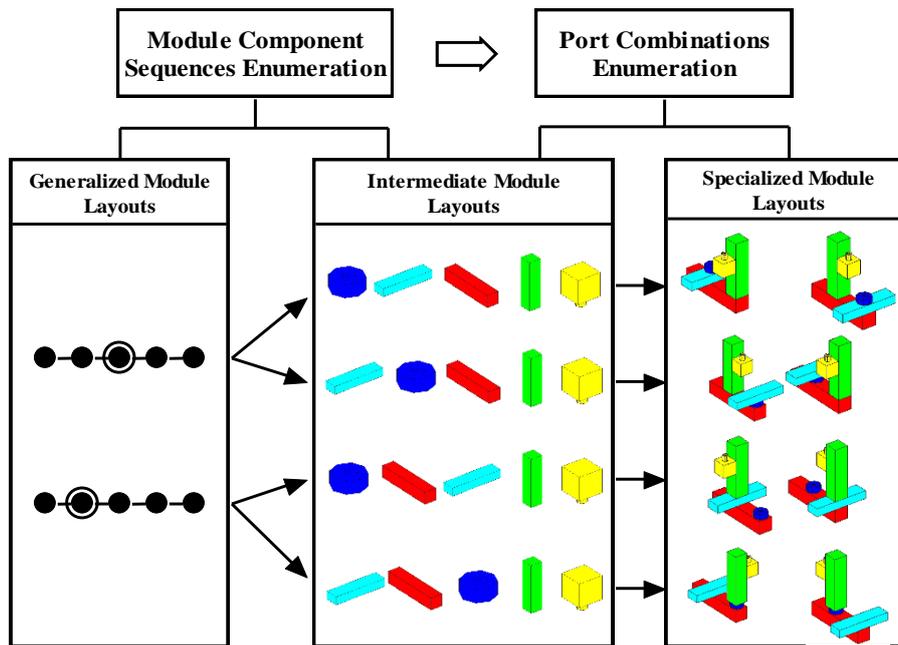
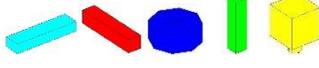
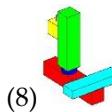
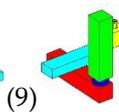
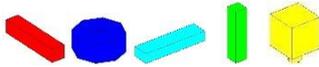
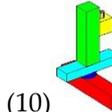
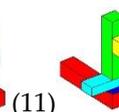
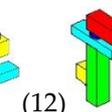
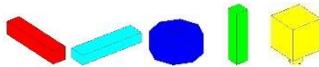
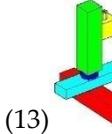
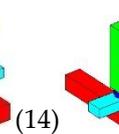
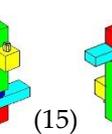


Figure 5. Flowcharts of the module assignment approach.

Table 6. Results of module assignment.

Generalized Layouts	Intermediate Layouts	Specialized Layouts
		(1) (2)
		(3) (4) (5)
		(6) (7)

Table 6. Cont.

Generalized Layouts	Intermediate Layouts	Specialized Layouts
		(8)  (9) 
		(10)  (11)  (12) 
		(13)  (14)  (15) 

4.3. Layout Evaluation

Layout evaluation is to find a best solution from the aforementioned 15 solutions. Three indices are used for this purpose.

4.3.1. Reconfigurability Analysis

First, reconfigurability analysis is to measure how easily a machine layout can be reconfigured. The reconfigurability index is put forward as follows:

$$R_k = \frac{1}{\sum_{b=1}^d N_b}, \tag{1}$$

where N_b is the number of steps required to change from configuration 1 to configuration b ; d is the total number of configurations required; and k means the k th layout solution. We choose the first layout solution as an example. Figure 6 demonstrates the calculation of the reconfigurability index for the first layout solution. A total of five adding/removing steps are required to change from the initial configuration to the last configuration. Therefore, the reconfigurability index is $1/5 = 0.2$. The complete result of reconfigurability analysis is listed in Table 7 for all 15 solutions.

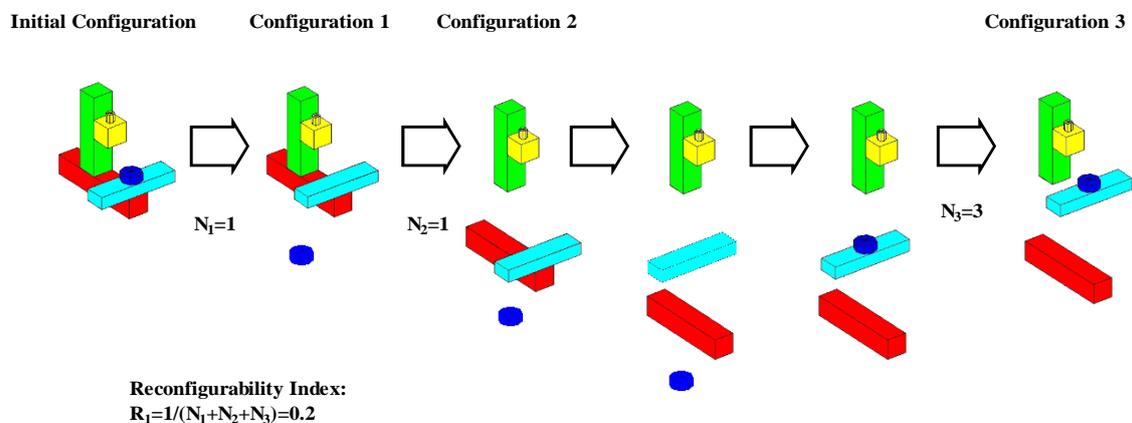
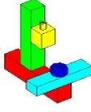
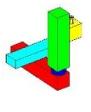
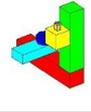
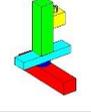
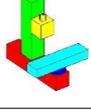
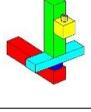
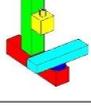
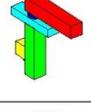
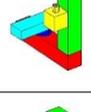
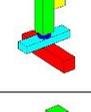
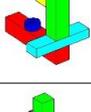
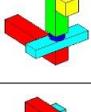
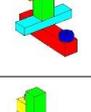
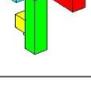
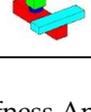


Figure 6. An example of the reconfigurability analysis of a layout.

Table 7. Results of reconfigurability analysis.

S/N	CAD Model	N	R	S/N	CAD Model	N	R
1		5	0.2000	9		7	0.1429
2		5	0.2000	10		9	0.1111
3		7	0.1429	11		9	0.1111
4		7	0.1429	12		9	0.1111
5		7	0.1429	13		7	0.1429
6		7	0.1429	14		7	0.1429
7		7	0.1429	15		7	0.1429
8		7	0.1429				

4.3.2. Structural Stiffness Analysis

Second, structural stiffness analysis is to measure how strong a layout can be for the generated MRMT. The stiffness matrix is expressed as

$$K^j = (J^j \bar{K}^{-1} J^{jT})^{-1}, \quad (2)$$

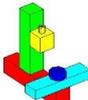
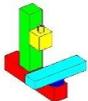
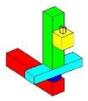
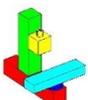
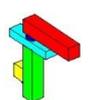
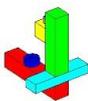
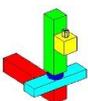
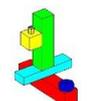
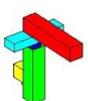
where \bar{K} is the local stiffness matrix of the module, and J is Jacobian. Since there are translational and revolute modules, each layout has a workspace. In order to measure the stiffness of the layout over the workspace, the following stiffness index is introduced

$$\mu = \frac{1}{N} \sum_{j=1}^N (tr(K^j)), \quad (3)$$

where N represents the total number of sampling points, and $tr()$ indicates the matrix trace operation. The index μ measures the average stiffness of a layout over the workspace. By assuming that all the local compliance indices are equal to 10^5 , and picking five sampling points in the workspace, the result of structural stiffness analysis is listed in Table 8 for all 15 solutions. A similar idea was presented in [29] for multitasking machines. Generally, the system stiffness should fall under the standard value

of machine tools, i.e., $>0.1\text{N/nm}$, depending on the required accuracy. It can be seen from Table 8 that solution 5 has the highest stiffness.

Table 8. Results of structural stiffness analysis.

S/N	CAD Model	μ (N/nm)	S/N	CAD Model	μ (N/nm)
1		0.0710	9		1.5642
2		0.5593	10		0.0216
3		0.0765	11		0.0413
4		0.0671	12		0.0611
5		2.4138	13		0.0201
6		0.0458	14		0.0349
7		0.0265	15		0.4845
8		0.0521			

4.3.3. Error Sensitivity Analysis

The purpose of error sensitivity analysis is to assess the effect of the module error to the tool tip error. The relationship between two errors is given as

$$\begin{bmatrix} \delta\theta_x^j \\ \delta\theta_y^j \\ \delta\theta_z^j \\ \delta x^j \\ \delta y^j \\ \delta z^j \end{bmatrix} = \begin{bmatrix} \Phi_B^j & 0 \\ 0 & I_3 \end{bmatrix}^{-1} J^j \begin{bmatrix} \delta q_1 \\ \delta q_2 \\ \delta q_3 \\ \vdots \\ \delta q_n \end{bmatrix}. \quad (4)$$

In the above equation, $\delta\theta_x^j$, $\delta\theta_y^j$, and $\delta\theta_z^j$ represent the angular displacements of the tool tip in terms of Bryan angle; Φ_B^j is the transformation matrix between the angular displacements in the global

coordinate system and the Bryan angular displacements; I_3 represents a 3 by 3 identity matrix; and δq_i represents the i th module error. The linear and angular errors of the tip can be calculated by

$$\delta h^j = \sqrt{(\delta x^j)^2 + (\delta y^j)^2 + (\delta z^j)^2}, \quad (5a)$$

$$\delta \theta^j = \cos^{-1}((\cos \delta \theta_x^j \cos \delta \theta_y^j + \cos \delta \theta_y^j \cos \delta \theta_z^j + \cos \delta \theta_z^j \cos \delta \theta_x^j - \sin \delta \theta_x^j \sin \delta \theta_y^j \sin \delta \theta_z^j - 1)/2). \quad (5b)$$

In this context, two evaluation indices, S_p and S_o , are introduced for position error sensitivity and orientation error sensitivity, respectively, as follows:

$$S_p = \frac{1}{N} \sum_{j=1}^N \left(\frac{1}{n} \sum_{i=1}^n \frac{\delta h^j}{\delta q_i} \right), \quad \delta q_i = c, \delta q_t|_{t \neq i} = 0, \quad t = 1, 2, \dots, n, \quad (6a)$$

$$S_o = \frac{1}{N} \sum_{j=1}^N \left(\frac{1}{n} \sum_{i=1}^n \frac{\delta \theta^j}{\delta q_i} \right), \quad (6b)$$

where c represents module error. For the given example, c is set 0.1 and the total number of sampling points as 5. The result of error sensitivity analysis is listed in Table 9 for all 15 solutions. It can be seen that solution 5 has the least position error sensitivity, and is in the 2nd least category of the orientation error.

Table 9. Results of error sensitivity analysis.

S/N	CAD Model	S_p	S_o	S/N	CAD Model	S_p	S_o
1		250.3333	0.6667	9		42.1667	0.5000
2		79.3979	0.5000	10		476.2317	0.5714
3		200.5000	0.5000	11		280.7751	0.4286
4		255.3333	0.6667	12		226.2857	0.4286
5		33.8333	0.5000	13		488.7837	0.5714
6		254.0173	0.5000	14		280.7427	0.4286
7		463.2140	0.6667	15		119.3801	0.4286
8		276.8262	0.6667				

4.4. Layout Selection

A final layout is decided by considering all the afore-mentioned four indices, based on which a combined index is proposed

$$L_k = \mu^r \mu_k^r + \mu^s \mu_k^s - \mu^e (\mu^p \mu_{k,p}^e + \mu^o \mu_{k,o}^e). \quad (7)$$

In the above equation, r , s , and e represent reconfigurability, structural stiffness, and error sensitivity, respectively; k refers to the k th layout solution; μ^r , μ^s , and μ^e are the importance weights of the three criteria; μ^p and μ^o are the importance weights of position error sensitivity and orientation error sensitivity, respectively; and μ_k^r , μ_k^s , $\mu_{k,p}^e$ and $\mu_{k,o}^e$ are the normalized reconfigurability index, structural stiffness index, position error sensitivity index, and orientation error sensitivity index, respectively.

In this example, we assume that the three criteria are of equal importance and thus their importance weights can be assigned as: $[\mu^r, \mu^s, \mu^e] = [0.3333, 0.3333, 0.3333]$. We also assume that the position error sensitivity and the orientation error sensitivity are of equal importance: $[\mu^p, \mu^o] = [0.5, 0.5]$. The overall indices of all the layout solutions are listed in Table 10. The layout solutions with a higher index value are considered as more suitable for the target machine tool design. In view of Table 10, layout #5 is selected as the final design because this layout has the highest combined index value, with a very high structural stiffness, low error sensitivity and average reconfigurability.

Table 10. Overall indices of the layout solutions.

S/N	Normalized Indices				L
	μ^r	μ^s	μ_p^e	μ_o^e	
1	0.0925	0.0128	0.0672	0.0831	0.0100
2	0.0925	0.1010	0.0213	0.0623	0.0506
3	0.0661	0.0138	0.0538	0.0623	0.0073
4	0.0661	0.0121	0.0685	0.0831	0.0008
5	0.0661	0.4357	0.0091	0.0623	0.1554
6	0.0661	0.0083	0.0681	0.0623	0.0031
7	0.0661	0.0048	0.1243	0.0831	−0.0109
8	0.0661	0.0094	0.0743	0.0831	−0.0011
9	0.0661	0.2824	0.0113	0.0623	0.1039
10	0.0514	0.0039	0.1278	0.0712	−0.0147
11	0.0514	0.0075	0.0753	0.0534	−0.0018
12	0.0514	0.0110	0.0607	0.0534	0.0018
13	0.0661	0.0036	0.1311	0.0712	−0.0105
14	0.0661	0.0063	0.0753	0.0534	0.0027
15	0.0661	0.0875	0.0320	0.0534	0.0370

5. Conclusions

It is demonstrated through this paper that the proposed method can effectively design a modular reconfigurable machine tool. The following conclusions can be drawn:

- For a given family of parts, by extracting their machining features, the corresponding module sets can be properly selected, from which a number of machine configurations can be grouped. By combining singular and/or similar modules through reconfiguration, a minimal set of modules can be determined.
- A layout synthesis process can then be carried out to produce generalized layouts based on a tree-graph algorithm and reduce them to create specific layouts by examining the selected modules.

- Finally, a layout evaluation process is conducted to find a final design based on a combined index considering three important issues for design of modular machine tool, i.e., reconfigurability, stiffness and error sensitivity.
- The proposed design system has been developed into a design software using MATLAB. The method presented can be extended to micro-machine design as described in [30].

Author Contributions: All authors contributed equally to this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References and Note

1. Yim, M.; Shen, W.; Salemi, B.; Rus, D.; Moll, M.; Lipson, H.; Klavins, E.; Chirikjian, G. Modular Self-Reconfigurable Robot Systems—Challenges and Opportunities for the Future. *IEEE Robot. Autom. Mag.* **2007**, *14*, 43–52. [[CrossRef](#)]
2. Arbuckle, D.; Requicha, A.A.G. Active Self-Assembly. In Proceedings of the 2004 IEEE International Conference on Robotics and Automation, Barcelona, Spain, 18–22 April 2005; pp. 896–901.
3. Yoshida, E.; Matura, S.; Kamimura, A.; Tomita, K.; Kurokawa, H.; Kokaji, S. A Self-Reconfigurable Modular Robot: Reconfiguration Planning and Experiments. *Int. J. Robot. Res.* **2002**, *21*, 903–915. [[CrossRef](#)]
4. Chen, I.-M. Rapid Response Manufacturing through Reconfigurable Robotic Workcells. *J. Robot. Comput. Integr. Manuf.* **2001**, *17*, 199–213. [[CrossRef](#)]
5. Moon, Y.; Kota, S. Design of Reconfigurable Machine Tools. *ASME J. Manuf. Sci. Eng.* **2002**, *124*, 480–483. [[CrossRef](#)]
6. Kota, S. A Methodology for Automated Design of Reconfigurable Machine Tools. *ERC/RMS Annu. Rep.* **1999**, 35–40.
7. Tilbury, D.M.; Kota, S. Integrated Machine and Control Design for Reconfigurable Machine Tools. In Proceedings of the 1999 IEEE/ASME, International Conference on Advanced Intelligent Mechatronics, Atlanta, GA, USA, 19–23 September 1999.
8. Moon, Y.M.; Kota, S. Generalized Kinematic Modeling of Reconfigurable Machine Tools. *J. Mech. Des.* **2002**, *124*, 47–51. [[CrossRef](#)]
9. Shinno, H.; Ito, Y. A Proposed Generating Method for the Structural Configuration of Machine Tools. *ASME Pap.* **1984**, *84*, 1–8.
10. Garro, O.; Martin, P. Towards New Architectures of Machine Tools. *Int. J. Prod. Res.* **1993**, *31*, 2403–2414. [[CrossRef](#)]
11. Ouyang, M.-A.; Yi, C.; Li, C.; Zhou, J. Intelligent Layout for Modular Design of Machine Tools. In Proceedings of the International Conference on Intelligent Manufacturing, Wuhan, China, 14–17 June 1995; Volume 2620, pp. 547–552.
12. Zatarain, M.; Lejardi, E.; Egana, F. Modular Synthesis of Machine Tools. *CIRP Ann. Manuf. Technol.* **1998**, *47*, 333–336. [[CrossRef](#)]
13. Otto, K.N.; Wood, K.L. *Product Design—Techniques in Reverse Engineering and New Product Development*; Prentice Hall: Upper Saddle River, NJ, USA, 2001.
14. Slocum, A. *Precision Machine Design*; Society of Manufacturing Engineer: Dearborn, MI, USA, 1998.
15. Ito, Y. *Modular Design for Machine Tools*; McGraw Hill: New York, NY, USA, 2008.
16. Chen, L.; Xi, F.; Macwan, A. Optimal Module Selection for Preliminary Design of Reconfigurable Machine Tools. *J. Manuf. Sci. Eng.* **2005**, *127*, 104–115. [[CrossRef](#)]
17. De Lacalle, L.N.L.; Lamikiz, A.; Sanchez, J.A.; de Bustos, I.F. Simultaneous Measurement of Forces and Machine Tool Position for Diagnostic of Machining Tests. *IEEE Trans. Instrum. Meas.* **2005**, *54*, 2329–2335.
18. ISO Technical Committee 184, Sub-Committee 4, STEP-ISO 10303, 1998.
19. Satty, T.L. *The Analytic Hierarchy Process*; McGraw-Hill: New York, NY, USA, 1980.
20. Regli, W.C.; Gupta, S.K.; Nau, D.S. Extracting Alternative Machining Features: An Algorithmic Approach. *Res. Eng. Des.* **1995**, *7*, 173–192. [[CrossRef](#)]
21. Gao, S.; Shah, J. Automatic Recognition of Interacting Machining Features Based on Minimal Condition Subgraph. *Comput. Aided Des.* **1998**, *30*, 727–739. [[CrossRef](#)]

22. Han, J. Manufacturing Feature Recognition from Solid Models: A Status Report. *IEEE Trans. Robot. Autom.* **2000**, *16*, 782–796.
23. Tsai, L. *Mechanism Design—Enumeration of Kinematic Structures According to Function*; CRC Press: Boca Raton, FL, USA, 2001.
24. Schmidt, L.C.; Shetty, H.; Chase, S.C. A Graph Grammar Approach for Structure Synthesis of Mechanisms. *ASME J. Mech. Des.* **2000**, *122*, 371–376. [[CrossRef](#)]
25. Yan, H.-S.; Chen, F.-C. Configuration Synthesis of Machining Centers without Tool Change Arms. *Mech. Mach. Theory* **1998**, *33*, 197–212. [[CrossRef](#)]
26. Chen, I.-M.; Burdick, J.W. Enumerating the Non-Isomorphic Assembly Configurations of Modular Robotic Systems. *Int. J. Robot. Res.* **1998**, *17*, 702–719. [[CrossRef](#)]
27. Rao, A.C. A Genetic Algorithm for Topological Characteristics of Kinematic Chains. *J. Mech. Des.* **2000**, *122*, 228–231. [[CrossRef](#)]
28. Valiente, G. *Algorithms on Trees and Graphs*; Springer: Heidelberg, Germany, 2002.
29. Diaz-Tena, E.; Ugalde, U. Propagation of Assembly Errors in Multitasking Machines by the Homogenous Matrix Method. *Int. J. Adv. Manuf. Technol.* **2013**, *68*, 149–164. [[CrossRef](#)]
30. Huo, D.; Cheng, K.; Wardle, F. A Holistic Integrated Dynamic Design and Modeling Approach Applied to the Development of Ultra-Precision Micro-Milling Machines. *Int. J. Mach. Tools Manuf.* **2010**, *50*, 335–343. [[CrossRef](#)]



© 2017 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).