DEVELOPMENT OF A DESIGN SUPPORT TOOL FOR SYNTHESISING MULTI-STATE MECHANICAL DEVICE CONCEPTS

Majumder, Anubhab;
Chakrabarti, Amaresh

Indian Institute of Science, Bangalore, India

ABSTRACT

Conceptual design synthesis, which focuses on generating solution alternatives, has a significant impact on the cost and quality of the final product. The development of radically new and significantly better solutions requires the generation and exploration of a large solution space. This work deals with the conceptual design synthesis of multi-state mechanical devices (MSMD). A scheme for representing a MSMD design task is described. Empirical studies have been carried out to develop a common understanding of the MSMD design synthesis process and use this knowledge for developing a prescriptive model. In order to support the effective and efficient use of the proposed prescriptive model, a web-based computational tool is developed.

Keywords: Conceptual design, Design methodology, Computational design methods, Design support tool, Multi-state mechanical devices

Contact:
Majumder, Anubhab
Indian Institute of Science
India
anubhabm@iisc.ac.in

Cite this article: Majumder, A., Chakrabarti, A. (2023) ‘Development of a Design Support Tool for Synthesising Multi-State Mechanical Device Concepts’, in *Proceedings of the International Conference on Engineering Design (ICED23),* Bordeaux, France, 24-28 July 2023. DOI:10.1017/pds.2023.146

1 INTRODUCTION

The conceptual design synthesis of mechanical devices can be defined as transforming a perceived need into a solution concept that incorporates mechanical engineering principles to satisfy the need. The conceptual design significantly influences the cost and quality of the final product (Hoover & Rinderle, 1989). In order to develop a radically better solution, it is essential to generate and explore a wide range of solutions (Fricke, 1996; Langdon & Chakrabarti, 1999; Liu et al., 2003). Traditionally, the synthesis of mechanical devices is classified into three categories: *Type*, *Number*, and *Dimensional* synthesis (Jiménez et al., 1997). This particular work specifically focuses on the *Type* synthesis of multiple-state mechanical devices. *Type* synthesis includes all those considerations concerning the selection of the type of mechanism that satisfies the design requirements. The kind of links or constructional units, such as bodies, joints, cams, gears, etc., are decided at this phase of conceptual design. In general, the input and output elements of mechanical devices have a single input-output relationship. However, for specific applications, a significant number of mechanical devices include multiple input-output relations, each of which is enabled by a set of functions present in the device's relevant operating state. In literature, these devices are termed multi-state mechanical devices (MSMD) (Li et al., 1999; Todeti, 2015), mechanisms with variable topology (MVT) (Yan & Kuo, 2006), multi-modal devices (Liu et al., 2015) and metamorphic mechanisms (Zhang et al., 2011). Several methods for automating the conceptual design synthesis process with computer support have been proposed in the literature. However, the majority of the existing works have focused on design problems with single operating states. Existing design synthesis processes use the following methods: function-based (Chakrabarti & Bligh, 1996), grammar-based (Starling & Shea, 2003), configuration-space (Subramanian & Wang, 1995; Joskowicz & Sacks, 1994), and matrix-based (Chiou & Kota, 1999) synthesis. Research reported on the conceptual design synthesis of MSMD has been sparse. Li et al. (1999) developed ADCS (**A**utomatic **D**esign by **C**onfiguration **S**pace), a computational tool for generating solutions for a given multi-state design task. ADCS uses the method of combinatorial retrieval of building blocks, which is simply a hierarchical search from requirement space to solution space, and the solution generated by ADCS is a network of building blocks. ADCS generates only one solution for a given task. ADCS does not consider modification of building blocks, and there is no guarantee that a compatible building block would exist in the ADCS database, in which case no solution could be generated. Zhang et al. (2011) proposed a morphological synthesis approach for synthesising MSMD design tasks. This approach was illustrated by a mechanism design example, in which the input effort was fixed to a particular driving element in all operating states, which is not necessarily true for every MSMD. Therefore, new support systems or enhancements to existing design synthesis tools are required to assist designers in generating a substantial variety of feasible alternative solutions for the conceptual design synthesis of MSMD.

2 RESEARCH OBJECTIVE AND METHODOLOGY

The overall objective of this research is to automatically or interactively synthesise a variety of possible mechanical device concepts that satisfy a given task with multiple operating states. Thus, it is essential to understand how designers carry out a MSMD design synthesis task. In this work, first, a scheme for representing a MSMD design task is described. Then, empirical studies are conducted with 12 designers, and a MSMD design task is given to them. All the synthesis processes are video recorded and analysed to develop a shared understanding of the synthesis process and outcomes. Based on this empirical understanding, a prescriptive model of MSMD design synthesis is developed. Finally, a web-based computational tool is implemented, aiming to support the effective and efficient use of the prescriptive model.

3 TASK REPRESENTATION

This section describes the representation scheme of the MSMD synthesis task. The scheme is primarily adopted from the existing literature (Li et al., 1999; Todeti, 2015), where elemental functions (EF) and specification graphs (SG) are used to specify the qualitative kinematics of the desired MSMD.

https://doi.org/10.1017/pds.2023.146 Published online by Cambridge University Press

3.1 Elemental functions

The relation from an input set to an output set in a MSMD synthesis task is called 'Elemental Function' (EF). For example, consider the following three EFs:

- f_1: When Element-A is at configuration a_1, and Element-B is at configuration b_1, if a torque is applied to Element-A in the anti-clockwise direction, Element-A should rotate in the anti-clockwise direction to configuration a_2 and simultaneously Element-B should also rotate in the anti-clockwise direction to configuration b_2.
- f_2: When Element-A is at configuration a_2, and Element-B is at configuration b_2, if a torque is applied to Element-A in the anti-clockwise direction, Element-A should not rotate any further, and Element-B should remain at configuration b_2.
- f_3: When Element-A is at configuration a_2, and Element-B is at configuration b_2, if a torque is applied to Element-A in the clockwise direction, Element-A should rotate in the clockwise direction to configuration a_1 but Element-B should not rotate.

In the above three EFs, there are two elements, A and B, both of which can act as input or output elements. The element to which effort (force or torque) is applied is called the input element. The output element is the element on which motion is desired. Here in this example, the set of configuration parameters of elements A and B is the configuration of the MSMD. The changes in the configuration of the MSMD in each of the above three EFs are: f_1: from $C_1 = (a_1, b_1)$ to $C_2 = (a_2, b_2)$ due to applying an effort on Element-A in the anti-clockwise direction, f_2: remains at $C_2 = (a_2, b_2)$ even if an effort is applied to Element-A in the anti-clockwise direction, f_3: from $C_2 = (a_2, b_2)$ to $C_3 = (a_1, b_2)$ due to applying an effort on Element-A in the clockwise direction.

The effort-motion relation of an element $'A'$ with motion axis X_A can be denoted as $\{E(X_A), M(X_A)\}$, where E and M represent the effort and motion kinds, respectively. The effort kind can be 'force (F)', 'torque (T)', or 'null (0)'. Likewise, the motion kind can be 'translation (Tr)', 'Rotation (Rt)', or 'null (0)'. The motion axis X_A specifies the intended direction of the input or output effort/motion. Six directions are considered here: $i+$, $j+$, $k+$, $i-$, $j-$, and $k-$. The EF of a MSMD with $'n'$ elements with motion axes $X_1, X_2, \dots X_n$ can be written as, $<\{E(X_1), M(X_1)\}, \{E(X_2), M(X_2)\}, \dots \{E(X_n), M(X_n)\}>$. Using this representation, the above EFs are represented as: $f_1: <\{T(k+), Rt(k+)\}, \{0, Rt(k+)\}>, f_2: <\{T(k+), 0\}, \{0,0\}>, f_3: <\{T(k-), Rt(k-)\}, \{0,0\}>$.

Effort (E) and motion (M) can also be represented using qualitative values of $+$, $-$, 0. In this case, effort or motion kind are not explicitly mentioned. As per this representation, the above EFs are written as: $f_1: <(+,+), (0,+)>, f_2: <(+,0), (0,0)>, f_3: <(-,-), (0,0)>$.

3.2 Specification graph

Each operating state of a MSMD involves one or more EFs, each corresponding to a particular kinematic configuration of the device or a transition from one configuration to another. This can be represented as a digraph called the specification graph (SG). Each node (C_i) of the SG denotes a kinematic configuration of the MSMD. An EF can be represented with an arc directed from the node C_j to C_k. If an arc starts and ends at the same node, it implies that there has been no change in the configuration of the device. Figure 1 shows the SG of the three EFs mentioned above. In all three EFs, Element-A acts as the input element. However, there are no fixed input or output elements for a MSMD in general.

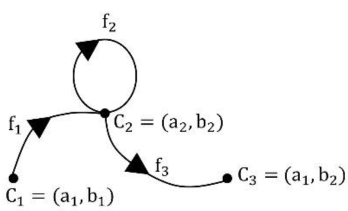


Figure 1. Specification graph of an example design task

3.3 Classification of elemental functions

From the above discussion, it can be observed that an EF may or may not lead to a change in kinematic configuration. It is also important to note that the input to the device can be either an external effort (e.g., a user input) or an internal effort (e.g., an energised spring present within the system enabling a specific change in configuration in the absence of an external effort). The latter can be considered a 'null (0)' effort in the representation of an EF. Considering all logically feasible EFs, the EFs of a MSMD can be broadly classified into four categories:

- Type-1: An external effort is applied, and the configuration changes.
- Type-2: An external effort is applied, but the configuration doesn't change.
- Type-3: The configuration changes in the absence of an external effort.
- Type-4: No external effort is applied, and no change in configuration occurs.

4 EMPIRICAL STUDIES

To obtain a better understanding of the MSMD synthesis process, empirical studies were conducted with 12 designers. All designers were postgraduate students (Masters' or Ph.D.) with a bachelor's degree in mechanical engineering and proficient with the theory of machines/ mechanisms. The design task given to the designers is formulated from a door-latch device with a handle and block as input/output elements. No time limit was imposed for completing the synthesis task. The designers were asked to think-aloud and draw schematic diagrams of the proposed solution concepts on paper. The synthesis processes were video recorded. The task consists of two operating states (opening and closing) with five EFs ($f_i, i = 1 \dots 5$), where f_1, f_2, f_3 belong to the opening state, and f_4, f_5 belong to the closing state. The configuration parameters for the handle and block are θ and x, respectively with respect to the world coordinate system, as shown in Figure 2. The five EFs are described as follows:

- f_1: When the handle is at $\theta = \theta_1$ and the block is at $x = x_1$, if a torque is applied to handle along $k-$ direction, it should rotate from $\theta = \theta_1$ to $\theta = \theta_2$ and simultaneously, the block should translate from $x = x_1$ to $x = x_2$ along $i+$ direction.
- f_2: When the handle is at $\theta = \theta_2$ and the block is at $x = x_2$, if the torque is still applied to the handle along $k-$ direction, the handle should not move any further from $\theta = \theta_2$ and the block should also not move from $x = x_2$.
- f_3: When the handle is at $\theta = \theta_2$ and the block is at $x = x_2$, if the torque is released from the handle, it should rotate along $k+$ direction from $\theta = \theta_2$ to $\theta = \theta_1$ and simultaneously, the block should translate along $i-$ direction from $x = x_2$ to $x = x_1$.
- f_4: When the handle is at $\theta = \theta_1$ and the block is at $x = x_1$, if a force is applied to the block along $i+$ direction, it should translate from $x = x_1$ to $x = x_3$ but the handle should remain at $\theta = \theta_1$.
- f_5: When the handle is at $\theta = \theta_1$ and the block is at $x = x_3$, if the force is released from the block, it should translate along $i-$ direction from $x = x_3$ to $x = x_1$ while the handle is at $\theta = \theta_1$.

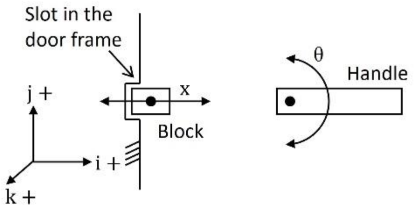


Figure 2. Elements of the MSMD design task used in the empirical studies

All recorded videos were analysed to develop a shared understanding of the designers' synthesis approach. The observations from one of the designers' synthesis processes in the empirical studies are described below:

After analysing the given EFs, the designer generated a rack and pinion mechanism as an initial solution proposal for f_1 (a Type-1 EF) as shown in Figure 3(a), where the handle was attached to the pinion with a fixed joint, and the rack acted as the block of the door-latch device. The designer modified the current solution to satisfy the directional specifications of f_1 by adding another gear, as shown in Figure 3(b). To satisfy f_2 (a Type-2 EF), a motion constraint was added to the rack, as shown in Figure 3(c). Next, a spring was added between the rack and the ground to satisfy f_3 (a Type-3 EF) as shown in Figure 3(d). To satisfy f_4, the designer observed a contradiction between f_1 and f_4 (both of these are Type-1 EFs) as the handle should remain detached from the rack during this operating state. This contradiction was solved by replacing the fixed joint between the handle and the gear with a

https://doi.org/10.1017/pds.2023.146 Published online by Cambridge University Press

variable constraint joint, as shown in Figure 3(e). The joint allows the transfer of torque from handle to gear while the handle is rotated along k − direction, but the joint disengages the gear and handle when the gear rotates along k − direction caused by the motion of the rack along i + direction. In order to stop the rotation of the handle due to its self-weight, a torsion spring was added between the handle and the ground (see Figure 3(f)). Finally, the designer analysed f_5 (a Type-3 EF) and realised that the current solution was already satisfying the requirement for f_5.

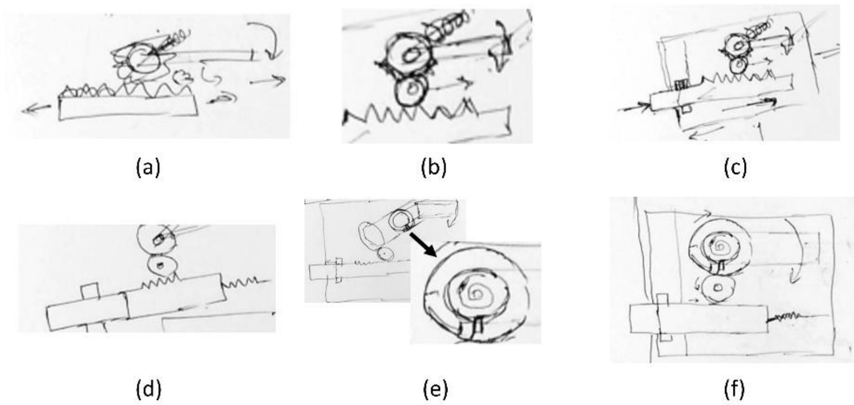


Figure 3. Initial solution generation and modifications done by one of the designers

5 A PRESCRIPTIVE MODEL

Based on the empirical studies, a prescriptive model is proposed for synthesising multiple solutions for a given MSMD synthesis task. It has been found from the empirical studies that while synthesising the MSMD, the designers typically select one of the Type-1 EFs for which a fully or partially satisfying solution proposal is generated. Further, this solution is modified until all of the EFs in the design task are satisfied. In the context of MSMD design synthesis, the approach observed in the empirical studies differs significantly from the morphological matrix approach employed by Li et al. (1999) and Zhang et al. (2011). In the case of a morphological matrix, an appropriate kinematic pair needs to be selected for each EF from a library of mechanisms realising different motion transformations. A final working solution can be achieved by combining those kinematic pairs as a kinematic chain. In contrast, the proposed prescriptive model intends to start with a semi-working initial solution and encourage the designer to modify it until it becomes a fully working solution. The steps involved in the proposed prescriptive model of the MSMD design synthesis process are given below:

1. Develop the SG of the given MSMD design synthesis task.
2. Analyse all the EFs in the SG to identify the types of EFs.
3. Select one Type-1 EF (an arc is SG directed from initial configuration C_i to final configuration C_f).
4. Generate an initial solution proposal for the selected Type-1 EF.
5. Modify the solution for the selected EF if it does not completely satisfy the EF.
6. Select the next EF, which starts at configuration C_f, as specified by the path in the SG.
7. Modify the solution if it does not satisfy the current EF till the current EF is completely satisfied.
8. Repeat step 6, followed by step 7 with other remaining EFs (if any), resulting in a solution satisfying all the EFs.

The above eight steps should lead to one working solution for the given MSMD synthesis task. In order to synthesise multiple solutions, there are two possible ways, as follows:

9. Repeat steps 6 to 8 with a different set of modifications.
10. Repeat steps 4 to 8 with a different initial solution proposal.

5.1 The need for developing a design support tool

The proposed prescriptive model addresses the approach of a MSMD synthesis task in order to generate multiple alternative solutions. In the empirical studies, the average time spent on the given design task to arrive at one final design solution that satisfied all the EFs was 31 minutes. The average percentage of total time spent on solving mismatches between f_1 and f_4 was found to be 70.5 %, which

https://doi.org/10.1017/pds.2023.146 Published online by Cambridge University Press

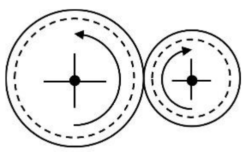
was much higher compared to that on any other EF. This could be caused by a lack of prior knowledge of metamorphic or variable-constraint kinematic joints. Some of the other issues identified from the empirical studies that prevented the designers from being able to produce a feasible design solution in a time-efficient manner include: 1. the fixation to explore more than one initial solution proposal for the chosen Type-1 EF; and 2. not exploring possible ways of modifying the geometric features in the elements of a solution proposal to satisfy multiple EFs. Therefore, there is a need to develop a design support tool to ensure the effective and efficient use of the proposed prescriptive model.

5.2 Generating initial solutions

As mentioned in the prescriptive model, the initial outcome of the synthesis process is a solution proposal that satisfies an EF of Type-1. For a single-input single-output MSMD design synthesis task, an initial solution proposal can be generated in the form of a primitive mechanism (e.g., slider-crank, double-slider, rack-and-pinion, etc.) or a combination of multiple primitive mechanisms. These primitive mechanisms are called building blocks. A building block comprises multiple physical elements connected by joints, which serve as an interface for energy flows from one element to the next. An effort can be applied to one of the building block's elements (input element) in either the positive or negative direction along or around its axis of motion. The effort causes motion in the output element based on the kinematic structure of the building block. Let us consider an EF of a MSMD design synthesis task with two elements: $\mathrm{e}_1$and e_2, which act as input and output elements, respectively. Considering the direction of effort applied to the input element along or around its axis of motion, four possible Type-1 EFs (without considering the effort/motion kinds) can be listed: $<(+,+),(0,+)>$, $<(+,+),(0,-)>$, $<(-,-),(0,+)>$, and $<(-,-),(0,-)>$. If the effort/motion kinds are also considered for representing the EFs, four possible input-output (I-O) motion relationships can be found: $\mathrm{Rt} \leftrightarrow \mathrm{Rt}$, $\mathrm{Rt} \leftrightarrow \mathrm{Tr}$, $\mathrm{Tr} \leftrightarrow \mathrm{Rt}$, $\mathrm{Tr} \leftrightarrow \mathrm{Tr}$. Now, if all the possible combinations of directions (i.e., $\mathrm{i}+$, $\mathrm{j}+$, $\mathrm{k}+$, $\mathrm{i}-$, $\mathrm{j}-$, and $\mathrm{k}-$) along or around the I-O axis of motion are considered, $6 \times 6 = 36$ possible EFs can be listed for a specific I-O motion relation. Thus, considering all 4 I-O motion relations, in total $36 \times 4 = 144$ possible Type-1 EFs can exist in a MSMD design synthesis task while generating an initial solution proposal.

The above analysis implies that in order to support the generation of an initial solution proposal, the design support tool should contain a database of building blocks, with each building block tagged with all possible EFs that it can satisfy. For example, Type-1 EFs that can be satisfied by a spur gear-pair are listed in Table 1. Likewise, multiple such building blocks and the corresponding EFs can be stored in the database. This can help designers retrieve multiple initial solution proposals for a specific Type-1 EF.

Table 1. Example of a building block and corresponding EFs

Building block	Elemental Functions
 Spur gear-pair	$<\{\mathrm{T(i+)}, \mathrm{Rt(i+)}\}, \{0, \mathrm{Rt(i-)}\}>$
	$<\{\mathrm{T(j+)}, \mathrm{Rt(j+)}\}, \{0, \mathrm{Rt(j-)}\}>$
	$<\{\mathrm{T(k+)}, \mathrm{Rt(k+)}\}, \{0, \mathrm{Rt(k-)}\}>$
	$<\{\mathrm{T(i-)}, \mathrm{Rt(i-)}\}, \{0, \mathrm{Rt(i+)}\}>$
	$<\{\mathrm{T(j-)}, \mathrm{Rt(j-)}\}, \{0, \mathrm{Rt(j+)}\}>$
	$<\{\mathrm{T(k-)}, \mathrm{Rt(k-)}\}, \{0, \mathrm{Rt(k+)}\}>$

5.3 The modification rules

Modifications are required when a proposed initial solution does not satisfy the subsequent EFs. After modifying the solution, it must also retain all the previously satisfied EFs. As modification of solution proposals is critical in the MSMD synthesis process, prescriptive knowledge about modification of solution proposals that can help designers generate alternative solutions has been developed in this work. Each modification to a solution proposal is intended to eliminate a mismatch between the required EF and the existing Type-1 EF exhibited by the initial solution proposal at a specific kinematic configuration. The mismatches between the existing Type-1 EF and the required EF are classified into three kinds (see Figure 4). In each kind, logically possible cases of mismatches have been identified. For each case of mismatch, various rules of modification are also proposed. A designer can modify a solution proposal for an identified case of mismatch using the proposed modification rules. The kinds of mismatches, various cases of mismatches in each kind, and associated modification rules for eliminating these mismatches are discussed below.

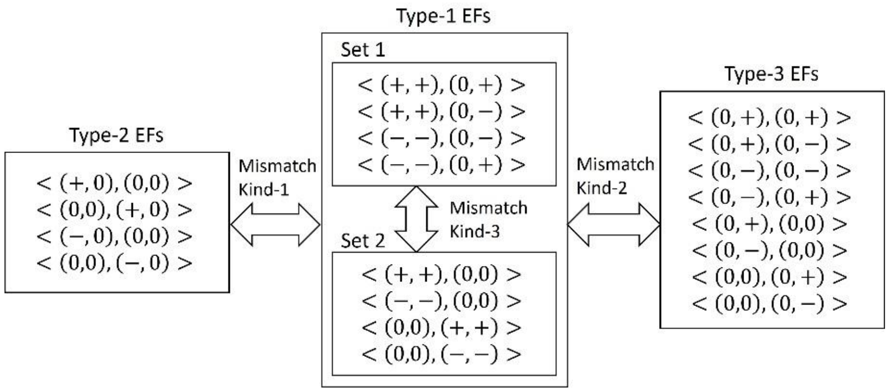


Figure 4. A comprehensive list of EFs and three kinds of mismatches that can occur while modifying the initial solution proposal

5.3.1 Mismatch Kind-1

This mismatch can occur when an initial solution generated for a Type-1 EF needs to satisfy a Type-2 EF. The characteristic of a Type-2 EF is to keep the configuration unchanged in the presence of an external effort. This implies that the motion of the I/O elements needs to be constrained at a specific configuration. There are two possible ways a constraint can be imposed on the elements of a building block:

- Case 1: The input motion axis directions are the same for both the Type-1 and Type-2 EFs (e.g., $<(+,+),(0,+)>$ and $<(+,0),(0,0)>$.
  - Rule 1: Impose geometric constraint (see Figure 5(a)).
- Case 2: The input motion axis direction of the Type-2 EF is opposite to the Type-1 EF (e.g., $<(+,+),(0,+)>$ and $<(-,0),(0,0)>$.
  - Rule 2: Impose spring controlled geometric constraint (see Figure 5(b)).

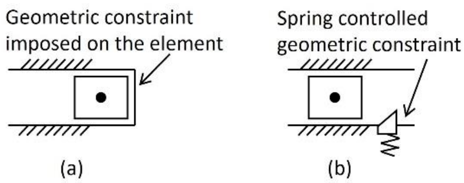


Figure 5. Modification examples of Mismatch Kind-1

5.3.2 Mismatch Kind-2

This mismatch can arise when an initial solution generated for a Type-1 EF requires satisfying a Type-3 EF where the configuration changes in the absence of an external effort. Here, the required effort to achieve the desired change in configuration has to come from the mechanical energy stored within the device. One of the possible ways to eliminate this mismatch is to add a spring element to the existing solution.

- Case 1: The motion axis direction of the Type-3 EF is opposite to the Type-1 EF (e.g., $<(+,+),(0,+)>$ and $<(0,-),(0,-)>$.
  - Rule 3: Add a spring element between the I/O element and ground such that the spring axis coincides with the motion axis of the I/O element (see Figure 6(a)).
- Case 2: The motion axis directions of the Type-3 and Type-1 EFs are the same (e.g., $<(+,+),(0,+)>$ and $<(0,+),(0,+)>$. This case implies snap-action bistable motion characteristics of a MSMD (e.g., an ON-OFF electrical switch).
  - Rule 4: Position a spring element between the I/O element in such a way that it supports snap-action motion (see Figure 6(b)).

https://doi.org/10.1017/pds.2023.146 Published online by Cambridge University Press

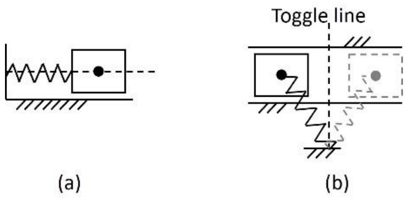


Figure 6. Modification examples of Mismatch Kind-2

5.3.3 Mismatch Kind-3

In Figure 4, it can be observed that the Type-1 EFs are divided into two distinct sets. Set 1 consists of those Type-1 EFs which cause changes in configuration parameters for both elements (i.e., $e_1$and e_2). These Type-1 EFs are considered while generating an initial solution proposal, as discussed in Section 5.2. Whereas, in Set 2, the Type-1 EFs cause a change in configuration parameters for one of the elements (i.e., either $e_1$or e_2). Mismatch Kind-3 can take place when an initial solution generated for a Type-1 EF that belongs to Set 1 is required to satisfy a Type-1 EF from Set 2.

- Case 1: The motion axis direction of the Type-1 EF from Set 2 is opposite to the Type-1 EF from Set 1 (e.g., $< (+,+),(0,+) >$ and $< (-,-),(0,0) >$.
  - Rule 5: Incorporate metamorphic or variable constraint kinematic joints (Zhang et al., 2011) in the existing initial solution. Figure 7(a) shows an example of a metamorphic joint.
- Case 2: The motion axis directions of the Type-1 EF from Set 1 and the Type-1 EF from Set 2 are the same (e.g., $< (+,+),(0,+) >$ and $< (+,+),(0,0) >$.
  - Rule 6: Incorporate metamorphic joints, along with force and geometric constraints, in the existing initial solution (see Figure 7(b)).

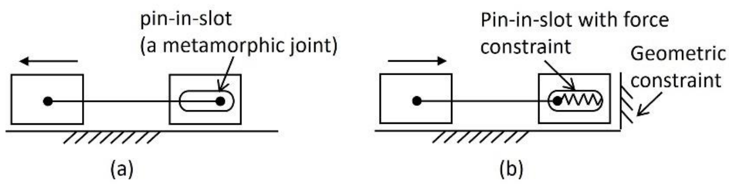


Figure 7. Modification examples of Mismatch Kind-3

6 DEVELOPMENT OF THE COMPUTATIONAL TOOL

A web-based application called CoDe SyMM (**Co**nceptual **De**sign **Sy**nthesis of **M**ulti-state **M**echanical Devices) has been developed to support designers in executing the design synthesis process for MSMD design tasks. The objective is to provide a knowledge base to the designers with which they can search through an existing database of building blocks (mechanisms/ kinematic pairs) and modification rules. The user can also contribute to the database by adding new building blocks or modification rules. CoDe SyMM has been coded with HTML, CSS, PHP, JavaScript and VPython. A MySQL server has been used for maintaining the database. The front-end user interface of the web application is coded with HTML, CSS, and JavaScript, which are rendered in the web browser. PHP is used as a server-side (back-end) scripting language. It helps in fetching the data from the database and sends the results to the web browser based on the user's request. VPython and JavaScript libraries are used for creating 3D animations in order to enhance the visual representation of modification rules, which is required to help designers understand the workings of the modification rules (Park et al., 2015). Figure 8(a) shows a screenshot taken from the "Search initial solution" tab of the tool. The user can enter the desired input/output motion kind and direction in this tab and search for relevant building blocks. In the current version of the tool, the building blocks are represented in the form of static 2D schematic diagrams. However, the modification rules can be retrieved in the form of 3D animations (see Figure 8(b)) using the "Modification Rules" tab. In the case of a specific Type-1 EF, if the tool doesn't contain a corresponding building block, the "Combine building blocks" tab can be used, where the tool can help in decomposing the entered Type-1 EF into two or more EFs for which a building block is available in the database. The user can combine those building blocks in order to generate an initial solution proposal. For the door-latch design task described in Section 4, three alternative solutions are created with the help of the tool, as shown in Figure 9. First, three initial solutions (slider-crank, rack-pinion and cam-follower) are selected from the suggested building blocks. Then the final working solutions are achieved by applying the modification examples provided by the tool.

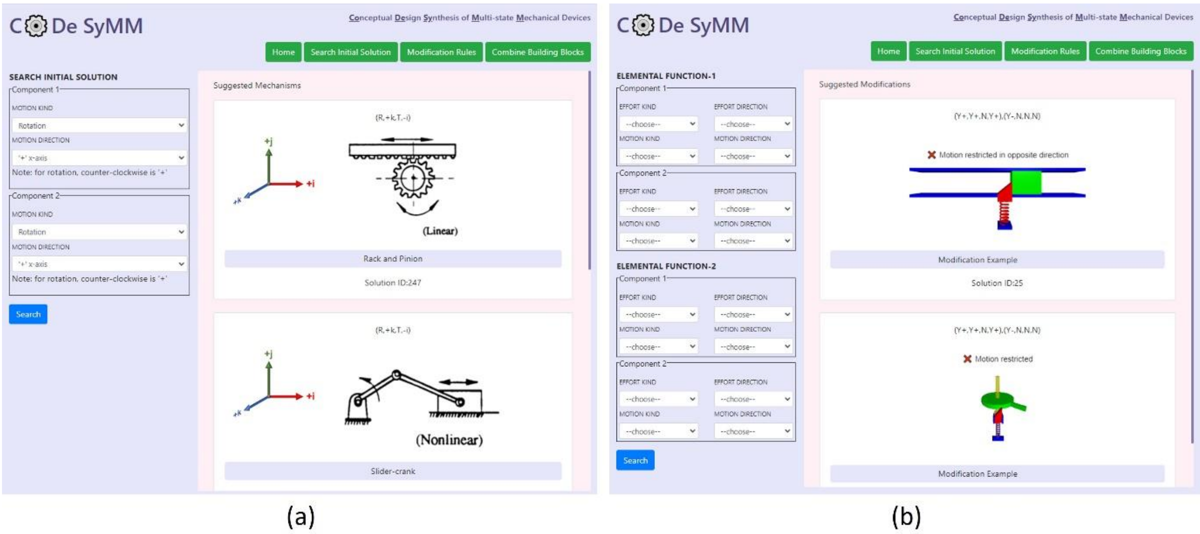


Figure 8. Screenshots from the CoDe SyMM application

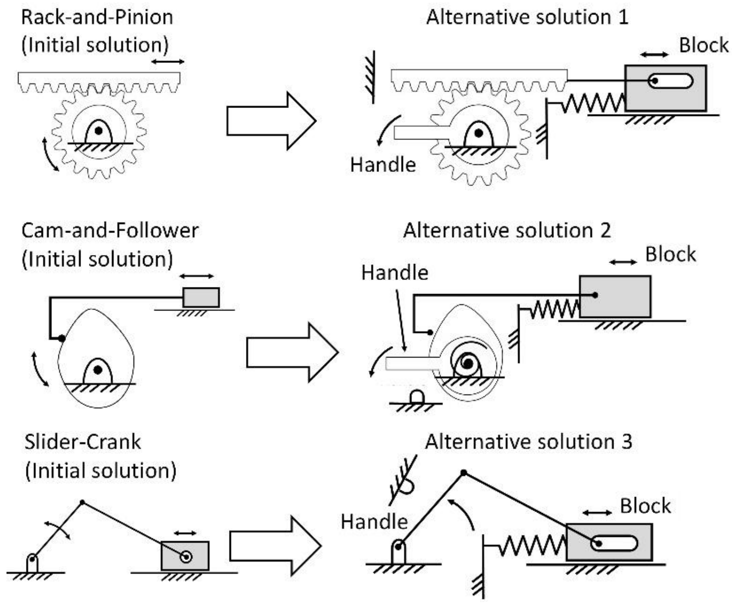


Figure 9. Three alternative solutions are generated with support from the tool for the door-latch design task

7 SUMMARY AND FUTURE WORK

In summary, this paper reports a prescriptive model for supporting the synthesis of multiple alternative solution concepts that fully satisfy a given MSMD design task. Empirical studies have been undertaken to understand how designers handle MSMD design tasks. It has been observed from these empirical studies that modification of semi-working initial solution proposals is the crucial, common process followed by all the designers. A method is proposed to create a database of building blocks that can support the generation of initial solutions. Mismatches between an existing EF and a required EF that can occur during the synthesis process are classified into three kinds. For each kind of mismatch, various cases of mismatch are developed through logical projections. For resolving each case of mismatch, a number of modification rules are proposed. The knowledge so developed about generating an initial solution and modifying initial solution proposals is further utilised in developing a

https://doi.org/10.1017/pds.2023.146 Published online by Cambridge University Press

web-based design support tool called CoDe SyMM to help designers generate a larger solution space for a given MSMD design task. Future work involves the following:

- A comprehensive evaluation of the prescriptive model involving potential users—designers—in solving different MSMD design tasks needs to be done.
- Currently, the CoDe SyMM database contains building blocks taken from Chiou & Kota (1999). This database can be further improved with other existing mechanism catalogues.
- More modification examples can be added to the modification rule database.
- The efficacy of the tool needs to be evaluated in terms of both fluency and variety of solution space generated by using the tool for a MSMD design task.

REFERENCES

Chakrabarti, A. and Bligh, T. P. (1996), "An approach to functional synthesis of mechanical design Concepts: Theory, applications, and emerging research issues", *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* Vol. 10 No. 4, pp. 313–331. https://doi.org/10.1017/s0890060400001645.

Chiou, S. J. and Sridhar, K. (1999), "Automated conceptual design of mechanisms", *Mechanism and Machine Theory*, Vol. 34 No. 3, pp. 467–495. https://doi.org/10.1016/s0094-114x(98)00037-8.

Fricke, G. (1996), "Successful individual approaches in engineering design", *Research in Engineering Design*, Vol. 8 No. 3, pp. 151–165. https://doi.org/10.1007/bf01608350.

Hoover, S.P. and Rinderle, J.R. (1989), "A synthesis strategy for mechanical devices", *Research in Engineering Design*, Vol. 1 No. 2, pp.87-103. https://doi.org/10.1007/bf01580203.

Jiménez, J.M., Alvarez, G., Cardenal, J. and Cuadrado, J., (1997), "A simple and general method for kinematic synthesis of spatial mechanisms", *Mechanism and Machine Theory*, Vol. 32 No. 3, pp. 323-341. https://doi.org/10.1016/S0094-114X(96)00017-1.

Joskowicz, L. and Sacks, E., (1994), "Configuration Space Computation for Mechanism Design", *IEEE International Conference on Robotics and Automation*, 1994, pp. 1080-1087. https://doi.org/10.1109/robot.1994.351215.

Langdon, P. and Chakrabarti A. (1999), "Browsing a Large Solution Space in Breadth and Depth", *International Conference in Engineering Design (ICED99)*, Munich, August 24-26,1999, pp. 24–26.

Li, C.L, Chan, K.W. and Tan, S.T. (1999), "A configuration space approach to the automatic design of multiple-state mechanical devices", *Computer-Aided Design*, Vol. 31 No. 10, pp. 621–653. https://doi.org/10.1016/S0010-4485(99)00058-5.

Liu, Y.C., Chakrabarti, A. and Bligh T. (2003), "Towards an 'ideal' approach for concept generation", *Design Studies*, Vol. 24 No. 4, pp. 341–355. https://doi.org/10.1016/s0142-694x(03)00003-6.

Liu, C., Hildre, H. P., Zhang, H., and Rølvåg, T. (2015), "Conceptual design of multi-modal products", *Research in Engineering Design*, Vol. 26 No. 3, pp. 219–234. https://doi.org/10.1007/s00163-015-0193-0.

Park, Y.S., Narsale, S.S., Mani, P.K. and Shah, J.J. (2015), "Multi-Modal Knowledge Bases to Facilitate Conceptual Mechanical Design", *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Boston, Massachusetts, USA, August 2-5, 2015, American Society of Mechanical Engineers. https://doi.org/10.1115%2Fdetc2015-46372

Starling, A.C. and Shea, K. (2003), "A Grammatical Approach to Computational Generation of Mechanical Clock Designs", *International Conference on Engineering Design ICED03*, Stockholm, August 19-21, 2003, pp. 445-446.

Subramanian, D. and Wang, C.S.E. (1995), "Kinematic synthesis with configuration spaces", *Research in Engineering Design*, Vol. 7 No. 3, pp.193-213. https://doi.org/10.1007%2Fbf01638099.

Todeti, S.R. (2015), *Understanding and Supporting Conceptual Design Synthesis of Multiple State Mechanical Devices*, Indian Institute of Science.

Yan, H.S. and Kuo, C.H., (2006), "Representations and identifications of structural and motion state characteristics of mechanisms with variable topologies", *Transactions of the Canadian Society for Mechanical Engineering*, Vol. 30 No.1, pp.19-40. https://doi.org/10.1139/tcsme-2006-0003.

Zhang, W. X., Ding, X.L. and Dai, J.S. (2011), "Morphological synthesis of metamorphic mechanisms based on constraint variation", *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, Vol. 225 No.12, pp. 2997–3010. https://doi.org/10.1177%2F0954406211408953.