CUED/C-EDC/ TR-8, 1991 (Oct).

5(P)

FUNCTIONAL SYNTHESIS OF SINGLE INPUT SINGLE OUTPUT SYSTEMS IN MECHANICAL CONCEPTUAL DESIGN

by

Amaresh Chakrabarti
Cambridge University Engineering department
Trumpington Street, Cambridge CB2 1PZ, U.K.

ABSTRACT

The conceptual design phase involves the derivation of solution concepts from the knowledge of functional requirements and constraints. In mechanical transmission design, the requirements are usually expressed as a transformation between the input and output characteristics. In the present paper, known solution concepts are analyzed to identify a set of primitive functional elements. These are then regarded as the building blocks, which can be combined to form more complicated elements or synthesized into solution concepts to satisfy specified instantaneous functional requirements, within the restriction of single input single output orthogonal systems.

1 INTRODUCTION

In this paper, engineering design is considered as the activity of deriving the physical description of an artefact to satisfy specified design requirements and constraints. According to one school of thought [Pahl & Beitz, 1984], in design, starting from the objective of a design problem we conceive the ideas (termed solution concepts) that would serve the specified objective. This phase is called the conceptual design phase. We then expand these concepts, using the knowledge of physics, engineering, etc., first,to a rough dimensioned layout, and eventually, to a concrete, optimized description. These two phases are respectively called the embodiment design and the detailed design phases. As we go down the design phases, information handling grows larger and succeeding design phases take substantially more time. However, it is important to ensure during the conceptual phase itself that the concepts achieve the required functionality, as the detailing makes it only physically achievable. Any error incurred in the concept derivation phase, in general, has a widespread effect on the rest of the design.

In most cases specified requirements and constraints can be satisfied by more than one artefact. In other words, there can be more than one possible solution. Commercial success of a design depends on four essential parameters : function, form, cost and time. Human designers prefer to use or fit previously known designs (they came across or designed before) as solution to their design problems. This, along with the constraint of meeting deadlines allow him to explore only a few of the feasible design solutions. This can lead to a just satisfactory design, rather than the optimum design.

Therefore, we need some way of deriving solution concepts, given the objective, that is able to:

1. generate more solution concepts than presently possible;
2. ensure that the objective is satisfied; and
3. do so in a comparable time to, if not faster than, that required by conventional methods.

This repertoire of solutions can then be checked against various criteria to choose the most promising solutions for further detailing. The present paper investigates the problem of synthesizing the instantaneous configurations of solution concepts to single input single output mechanical transmission design problems. The next section investigates the problem in greater detail.

2 PROBLEM

In mechanical transmission design, devices are designed to amplify forces, transmit torques, rotate wheels etc. In other words, they transmit and transform forces and motion, which technically is the act of transforming (a set of) input variable characteristics into (another set of) output variable characteristics. This, in essence, is the design objective.

As an example, suppose we require a device, as shown in fig.1, that would give a constant output force Fo at positions po1, po2, po3 and po4 respectively for a constant input force Fi at positions pi1, pi2, pi3 and pi4, at time instants t1, t2, t3 and t4 respectively. One of the ways it can be achieved is by a device consisting of an input rack, an intermediate pinion and an output rack, as in fig.2. At an instant, the input force and its position constitute the input characteristics. Similarly, the output force and position constitute the output characteristics. The temporal variation of the input characteristics forms the input function, while that of the output characteristics forms the output function. At a time instant, the input-output pair of characteristics is the instantaneous functional requirement or objective. The set of all such pairs forms the overall functional requirement or objective.

In this paper, we will consider only the instantaneous functional requirement of a transmission design problem and seek out means of deriving solution concepts to fulfil that objective. However, the same method could be used to generate concepts for each distinct instantaneous requirement of the overall function and the concepts which satisfy all such instantaneous requirements would be considered to have fulfilled the overall objective. We will limit ourselves to single input single output systems. So the problem can be stated as: how do we generate and configure solution concepts from the knowledge of specified requirements and constraints, in terms of instantaneous input-output characteristics? Is there any organised way of doing this, and if there is, is it reproducible?

3 APPROACH

How do we generate and configure solution concepts from the knowledge of requirements and constraints?

A solution concept is an idea of a physical structure, that may or may not be made of sub-structures, that is able to serve the specified requirement. In order to be able to generate a concept, we need to link the functional requirement, subjected to certain constraints, to the known principles, materials, structures etc.

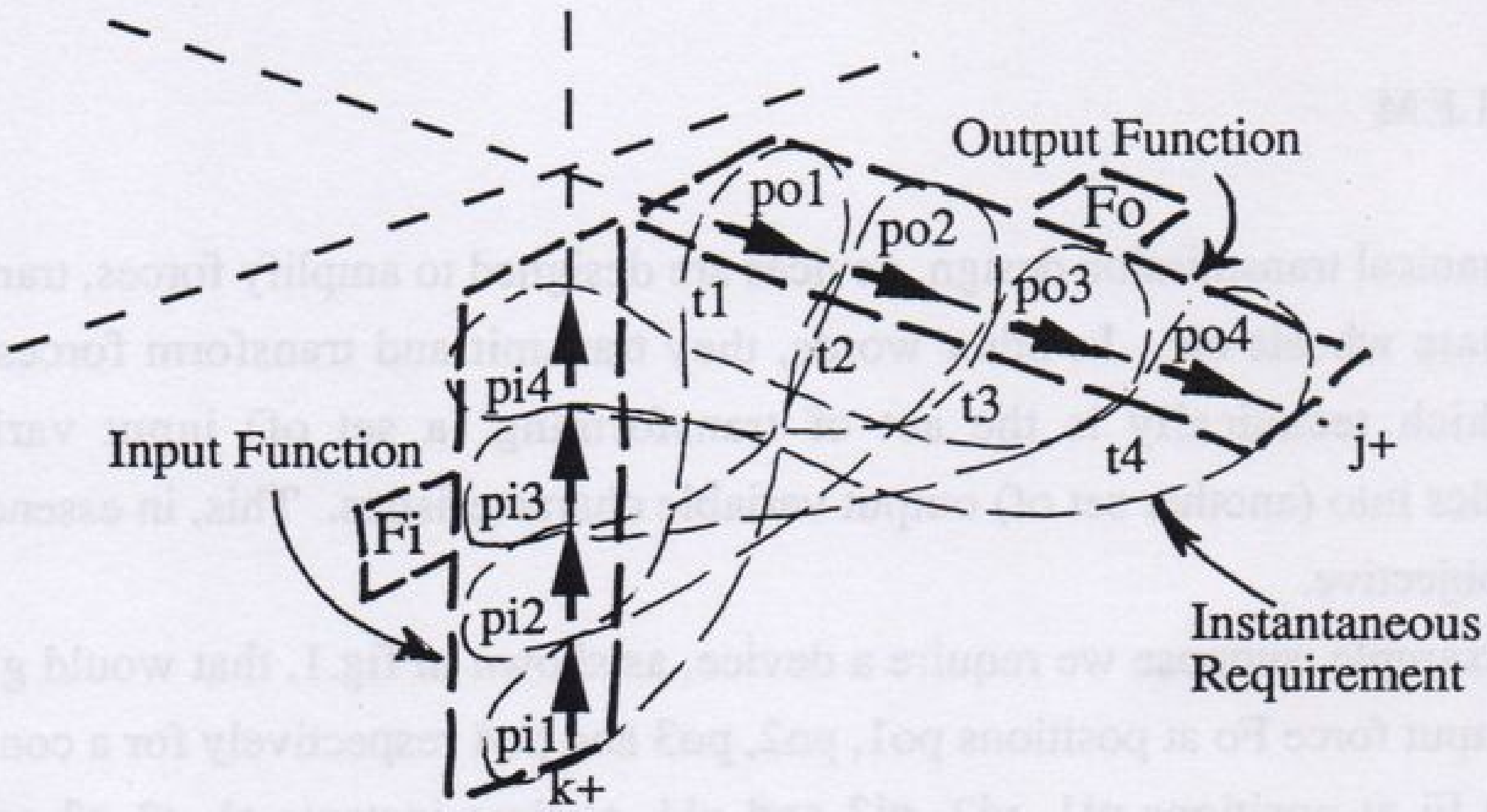


Fig.1 A Definition of.an Overall Transmission Function

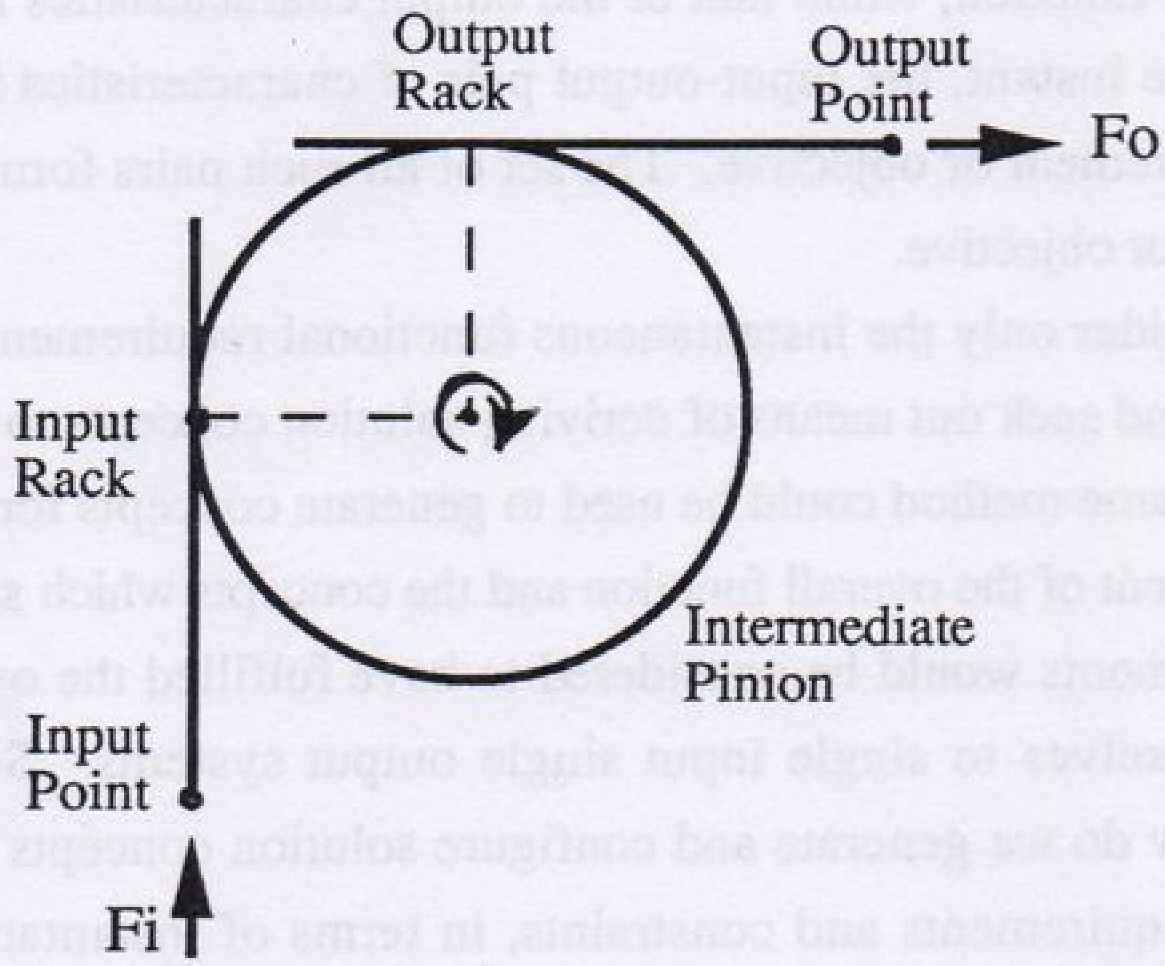


Fig.2 A solution Concept to Satisfy the Function shown in Fig.1

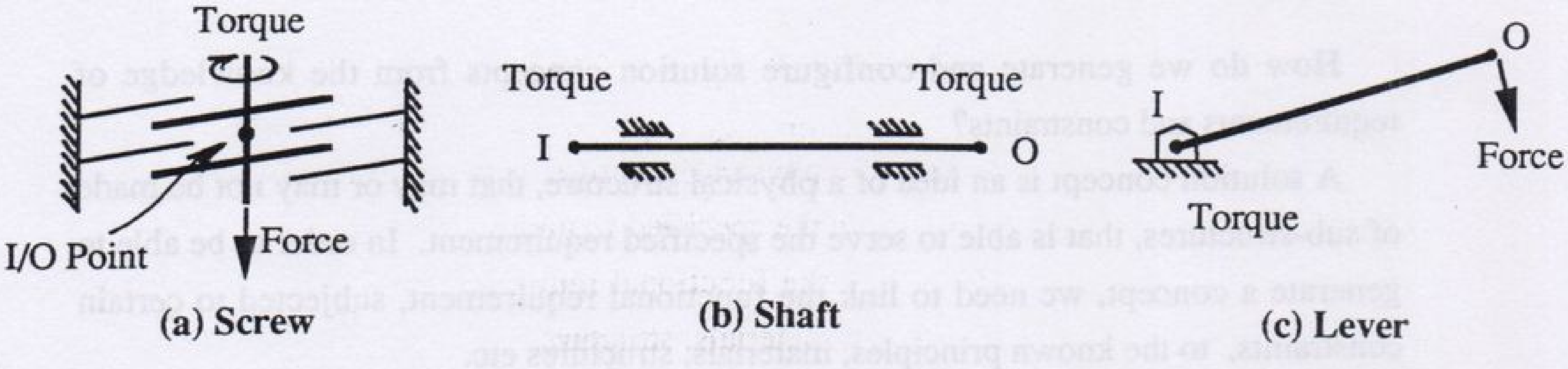


Fig.3 Examples of Primitive Functional Elements

One way of doing this would be to link each known transmission design solution to its served function and use it when exactly the same design problem is faced. This is totally restrictive because no new situation could be dealt with. However, if it were possible to identify and store the decision-making process undergone in each design, and partially modify this process when essentially similar designs with different parameter values are encountered, it might be an effective way of carrying out adaptive design [Mostow *et al*, 1989].

Research in the area of systematic design indicated that consideration of functional requirements as the criteria of identifying basic structural elements might lead to a consistent flow and growth of information in design [Pahl & Beitz, 1984; Grabowski & Benz]. Freeman & Newell [1971] suggested a functional model, primarily in the field of computer science, in which each structural element would be identified by the set of functions it could provide and the set of functions it would require in order to provide them. A super-structure could then be defined as an assembly of these structures. General characteristics and requirements of functional models have been further discussed in Schmekel [1989].

The field of mechanical design, however, is quite different from that of computer science due to the inseparability of function and structure. This is particularly prominent in transmission design domain. The task of linking function to structure therefore, probably because of the implicit constraint of separating function from structure, remains a major problem.

There have been very few attempts to understand the conceptual design process, and fewer to formalize it. There have been two notable exceptions. The first one is the classical kinematic approach, where a set of joints having specified types and number of degrees of freedom has been identified as basic structural elements. These elements are then combined to form assemblies, or kinematic chains, as they are called, for providing specified types and number of degrees of freedom. These kinematic chains can then be checked, after assumptions are made about input and output points and the distances between the joints, against required functional characteristics [Hoeltzel *et al*, 1987; Freudenstein & Maki, 1983]. In the other approach, a set of general physical characteristics in dynamic systems domain, called bond graph elements [Rosenberg & Karnopp, 1983], are used as the basic structural elements and their constitutive relations are combined, compatibly with specified input and output environments, to provide schematic solutions to transducer design problems [Ulrich & Seering, 1989].

However, before a representation qualifies as a basic evolutionary framework for mechanical conceptual design, it needs to satisfy two criteria.

The first criterion is, the representation should be abstract enough to represent the qualitative and imprecise character of the conceptual design phase, while still being able to capture the knowledge of geometry, a knowledge essential for mechanical design in general, and particularly so in transmission design. The bond graph approach seems to be too abstract to capture geometry [Shapiro & Voelcker, 1989], whereas the kinematic approach appears overly analytical to capture the desired qualitative nature of a conceptual design phase.

The second criterion is, the desired representation should be naturally extendible to support the detailed design activity. Moreover, research in detail design has already produced knowledge of how to detail the generated concepts [Shigley & Mitchell, 1983]. Any successful representation must be able to provide perceptible link between the knowledge generated in the conceptual design and that available for detailing. None of the above approaches seem to ensure this.

However, the above works influence the formation of the present approach in various ways. Firstly, the systematic design approach provides a criterion of identifying and classifying design knowledge, ie, the criterion of functionality. The classical kinematic approach provides a useful base of knowledge which acts both as an appropriate foundation to begin with and as a reasonable standard with which to compare present findings. The bond graph approach provides the idea of effort and flow as the essential input-output parameters for analysing energy-transformation systems. Finally, the work of Ulrich and Seering [1989] provides the important concept of power spine in SISO system synthesis.

The approach we take is to analyze known design solutions for the following:

1. We go on resolving the solutions into simpler sub-structures until we find the simplest identifiable sub-structures which cannot be resolved further.
2. We then look for functional similarity between the above sub-structures, and if it exists, try to identify the simplest possible sub-structures that are distinct from each other in terms of the input-output characteristics. In other words, we seek to identify those basic structures which can be combined to describe all the analysed solutions.
3. Once the above is found, we can combine these basic structures to synthesize a more complex structure or a new solution concept, and deduce the function(s) of the structure or concept so formed by combining the functions of its constituent structures. It is then possible to synthesize and configure new concepts from the library of primitive structures.

When this classifying criterion (input-output characteristics, or functionality) is applied, it is found that there exists only three kinds of sub-structures. Depending on whether the input and output vectors differ in terms of kind, magnitude, orientation or direction, without any change in position, the function is called conversion, and the structures able to do it, converters. A screw, for example, converts a torque vector into a force vector or vice-versa, without changing their position (fig.3a), and hence is a converter. If the input vector differs from the output vector in terms of position only, the function is called transmission, and the physical structures able to do it, transmitters. A shaft, for instance, simply changes the position of torque vector and hence is a transmitter (fig.3b). If the conversion of a vector happens with an inevitable change in position or vice-versa, the function can be called conversion-transmission, and the structures able to do it, converter-transmitters. A lever, for instance, converts a torque vector into a force vector and vice-versa, with an inevitable change in position, and hence is a converter-transmitter (fig.3c).

Let us take an example device to see if the functions of the sub-structures constituting the device can be combined to derive the function of the device. Fig.4 shows a clamping device. The head of the screw, when turned, makes the screw to go down and press the lever rod below. This causes the lever rod to rotate with respect to the hinge joint and thereby causes the head of the cam to press against the workpiece (shown with the dotted line). Using the terminology coined in the above discussion, we can say that the head of the screw AB is a converter-transmitter, which changes the hand force at point A into a torque at point B. The transmitter BC takes this torque down from point B to point C where it is converted to an axial force by the converter C (the screw), without any change in position. The force is carried down to point D by the transmitter CD and is taken up by the converter-transmitter DE (another lever) to convert it into a torque at point E. Finally, this torque is converted back to a force at point F by another converter-transmitter EF (a cam). It is to be noted that the conversion of the vectors in the process could occur in terms of either singly or in a combination of their magnitude, direction and kind.

So far we have found that the known transmission design solutions can be split into a set of distinct substructures, the functions of which can be combined to derive the function of the concept. The question is, can we do the reverse? Can we, starting from the desired function, combine the known primitive sub-structures to construct the solution concepts that would be able to serve the required function? We discuss this issue in detail in the following section.

4 SOLUTION TECHNIQUE

In the previous section, we discussed how mechanical transmission devices could be viewed as assemblages of converters, transmitters and converter-transmitters, and thereby how the functions of these devices could be derived from the functions of their constituent substructures. In this section, we would like to classify these primitive elements as functional elements and investigate the issue of constructing solution

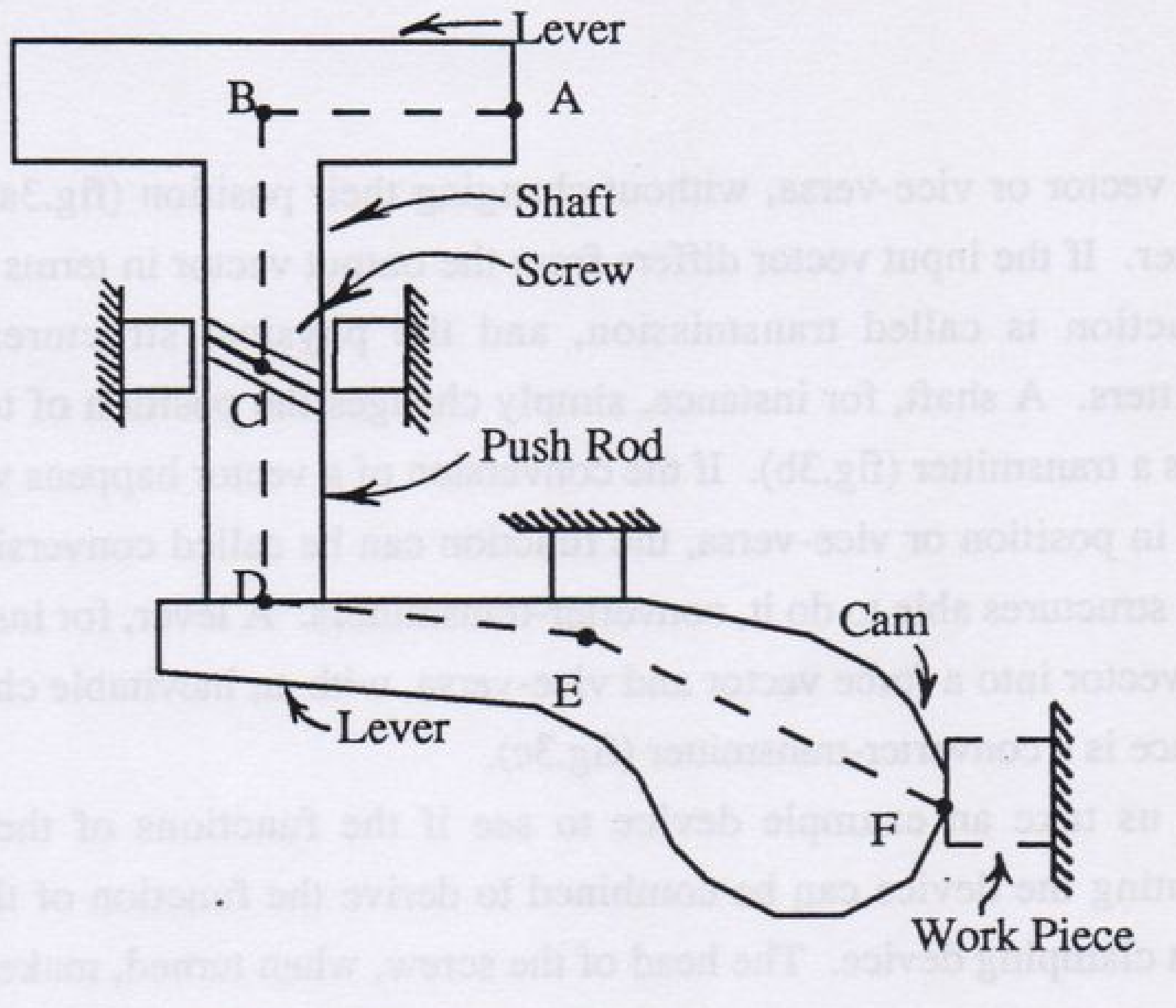
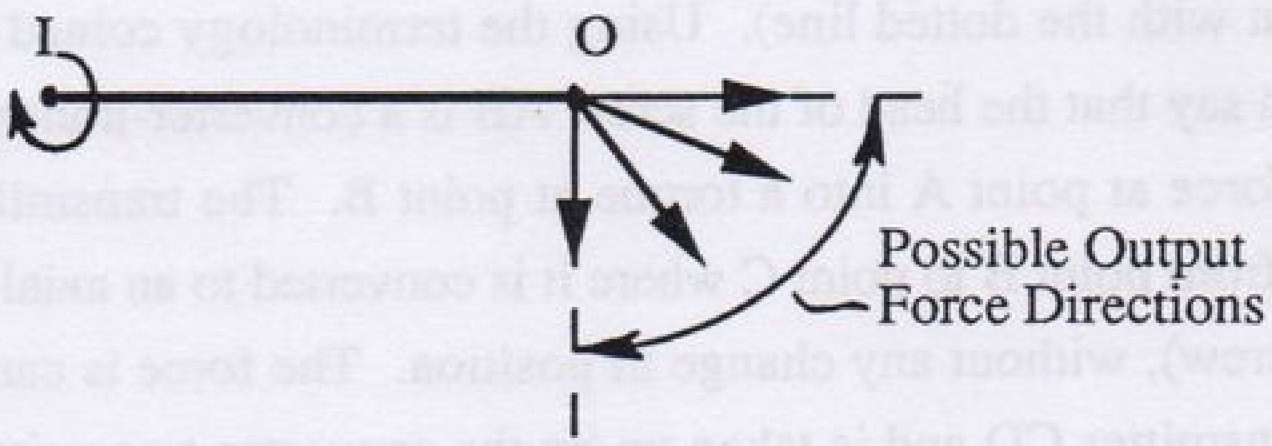
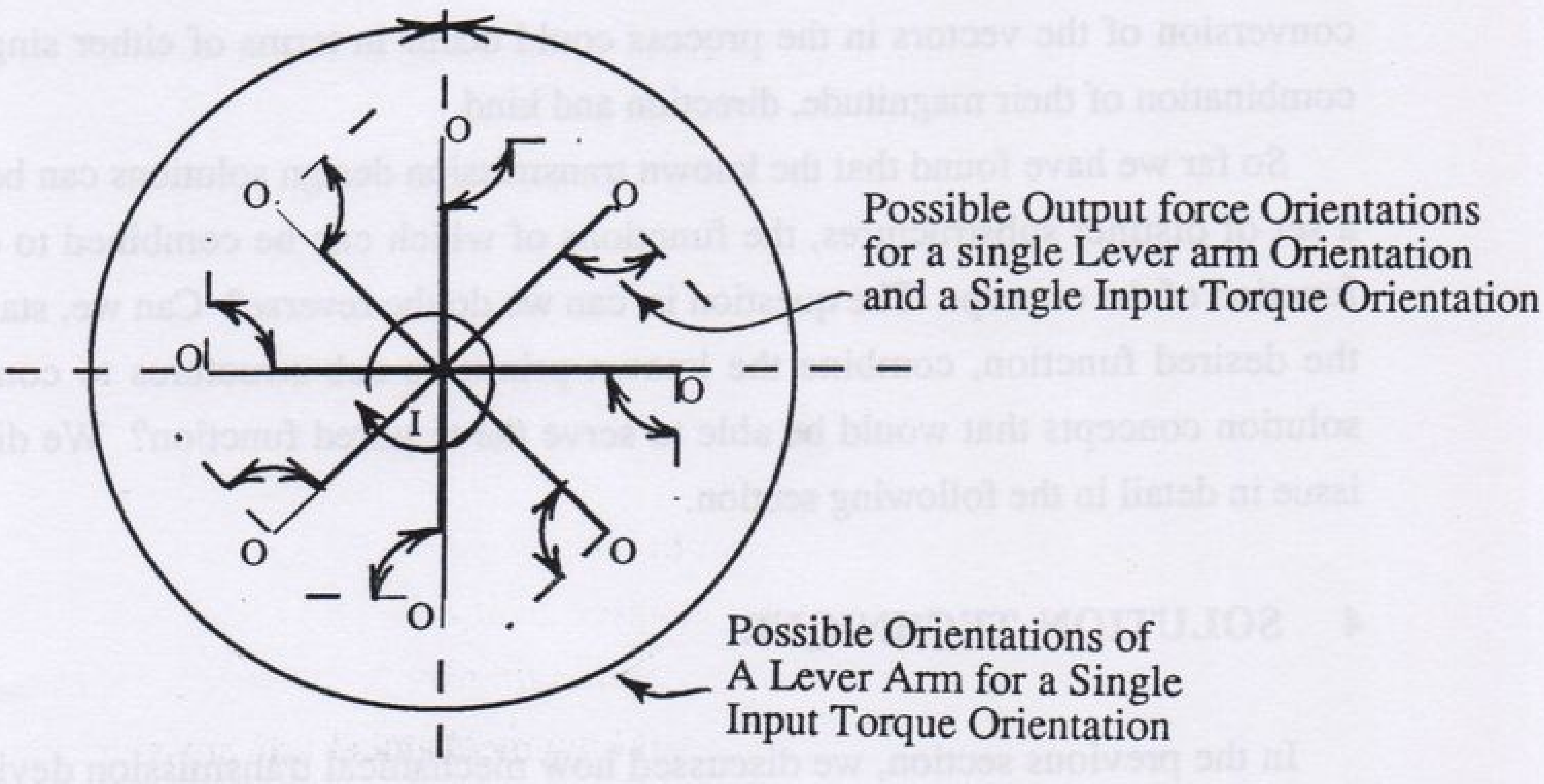


Fig.4 A Clamping Device



(a) A Single Orientation of A Lever Arm



(b) Possible Orientations of a Lever Arm

Fig.5 Functions as Orientation-Tranformation can be Infinite Even for a Single Element!

concepts, out of them, for serving specified functions. In order to do so, we need to be unambiguous about the functions of these elements. Otherwise, the ambiguities present in their functional definition would produce a greater order of ambiguity in that of the constructed concept, and hence make any prediction about the function of the constructed concept uncertain. In order to do this, we need to look at the definition of function in the context of transmission design.

A function is defined as a transformation between the characteristics (ie, kind, magnitude, orientation, direction and position) of the input and output vectors. Therefore, for each primitive functional element, its function will consist of a change in one, some or all the above characteristics. This necessitates the storage of an enormous number of possible vector transformations as the function-set for each element. A lever, for example, can have an infinite number of directions of force as output for a specific direction of input torque (fig.5a). And this is for but one of the infinite possible orientations of the lever arm in space (fig.5b)! These are not the only possible transformations. For each element, there can be a transformation associated with its input and output efforts, its input and output flows, its input flow and output effort, and its input effort and output flow. These define the kind transformations that a given element can be used for. Moreover, there can be an infinity of input-output magnitudes that each such element is functionally capable of handling.

How do we handle such an enormous number of transformations? Clearly, we can not handle it. We, therefore, need to limit the function-space to a tractable size. We have two ways. One is to look for natural constraints that forge some dependence between the input and output vectors. The other way is to make some assumptions to limit the function-space. To start with, we make the following two restrictions:

1. Input and output vectors for each primitive element can either be parallel or perpendicular to each other.
2. Input and output vectors for each primitive element can either be parallel or perpendicular to the length vector, the length vector being defined as the vector having the distance between the input and output point as the magnitude, and, the unit vector directed from the input point towards the output point as the direction.

In other words, we limit ourselves to those concepts that can have orthogonal orientations. This indeed limits the function space, but not yet to a tractable number. We, therefore, seek out, in the following sub-section, natural constraints for each separate parameters of the input-output characteristics, and based on that, try to derive some ways of constructing concepts out of the primitive elements.

4.1 Function Parameters

4.1.1 *Kind Synthesis*

This problem is to generate assemblies, out of the known set of primitive elements, that would take input of given a kind and deliver output of a desired kind. We want, for instance, to convert a force into a torque. Kind synthesis should try to form assemblies that would take force as input and deliver torque as output.There are two questions. One is, how do we choose the first element, and the next, and so on? The second is, where do we stop this process?

The answer to the first question is straightforward. We start by choosing any element from the known set that can have as input the specified system input. This will limit the possible output kinds, and the next element will be chosen from the elements that can take as input the output of the previously chosen element.

One obvious answer to the second question is that the process should stop whenever the output of the presently chosen element matches with the specified system output. But then we might miss out in the process some of the other requirements like position change etc. We therefore constrain the problem by specifying the maximum number of primitive elements we would allow to be used in constructing the concepts.

4.1.2 *Orientation Synthesis*

For each solution concept derived through the kind synthesis, there can be a number of orientation possibilities. How do we derive them?

The first natural constraint to the input and output vector for any primitive element is that they must be located on the length vector of the element. Now there can be only four ways by which the input and output vectors can be spatially related. They can be parallel and intersecting, parallel but non-intersecting, non-parallel but intersecting, and, non-parallel and non-intersecting. Within the scope of the assumptions made in the previous section, only the configurations given in fig.6 are possible for a single element.

Mathematically speaking, it is possible to have, for each possible combination of input-output kinds, each of the configurations shown in fig.6. The laws of motion, however, restrict some of them from existing physically and thereby limit the function-space as well as the consequent solution-space. For instance, though it is mathematically possible for a torque (and rotation) to get transformed into another torque (and rotation) by any of the configurations shown in fig.6 (a, b, c, d, e), none other than the configuration in fig.6a can physically realize this. These restrictions give

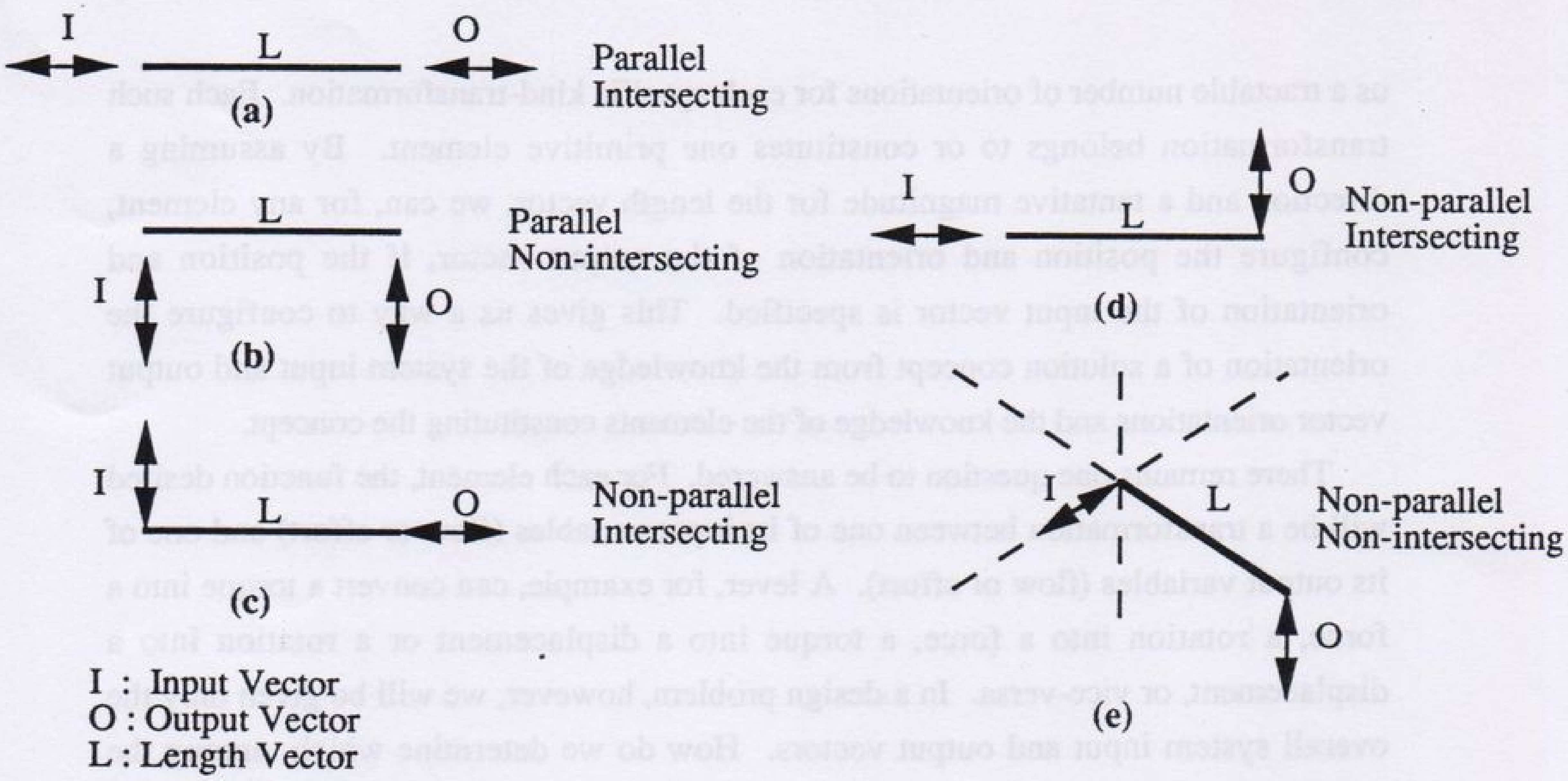


Fig.6 Basic Functional Elements

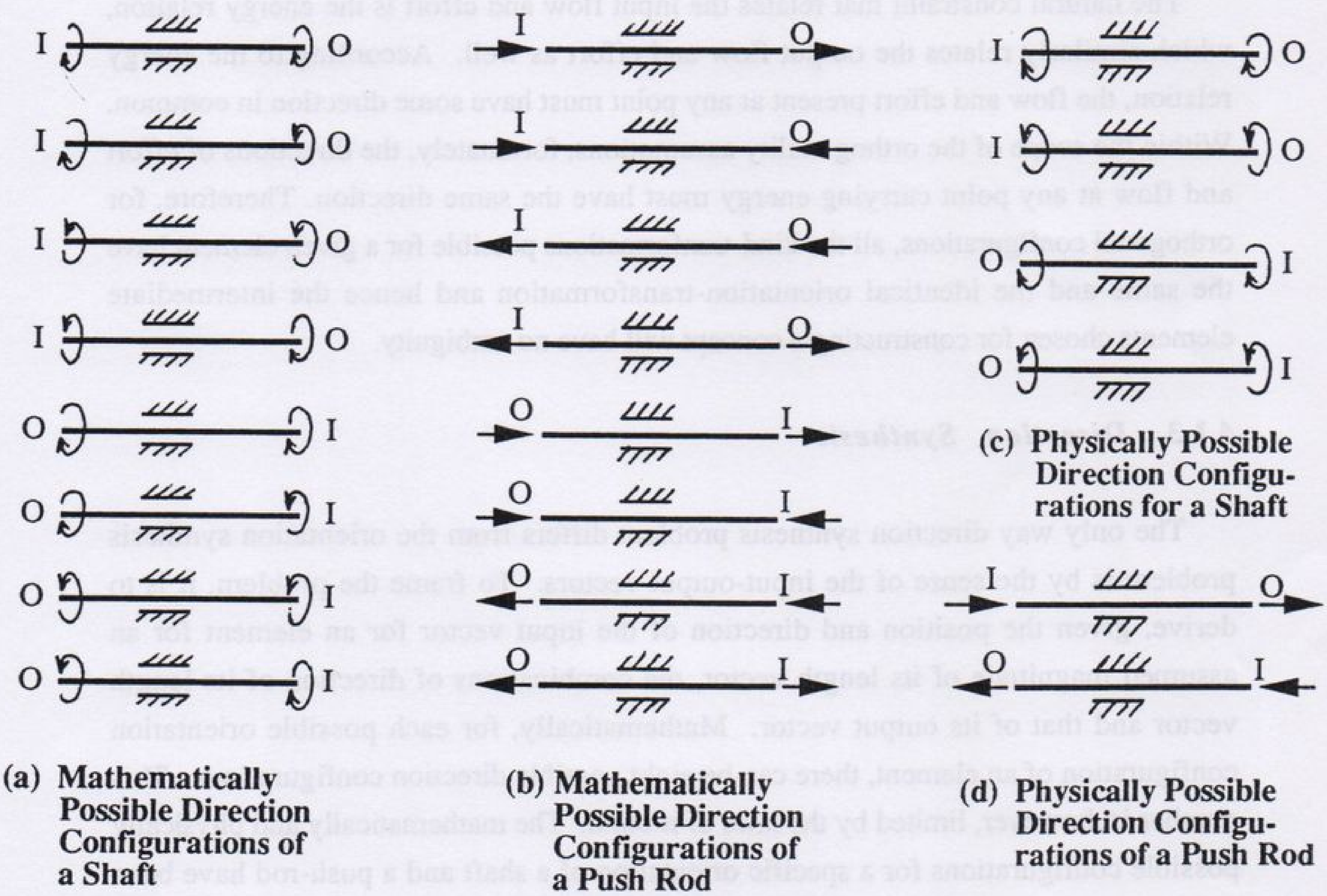


Fig.7 The Effect of Natural Constraint on Direction Configurations

us a tractable number of orientations for each specific kind-transformation. Each such transformation belongs to or constitutes one primitive element. By assuming a direction and a tentative magnitude for the length vector, we can, for any element, configure the position and orientation of the output vector, if the position and orientation of the input vector is specified. This gives us a way to configure the orientation of a solution concept from the knowledge of the system input and output vector orientations and the knowledge of the elements constituting the concept.

There remains one question to be answered. For each element, the function desired will be a transformation between one of its input variables (flow or effort) and one of its output variables (flow or effort). A lever, for example, can convert a torque into a force, a rotation into a force, a torque into a displacement or a rotation into a displacement, or vice-versa. In a design problem, however, we will be given only the overall system input and output vectors. How do we determine which, among the above four transformations, to choose for each intermediate element used in constructing the possible concepts?

The natural constraint that relates the input flow and effort is the energy relation, which similarly relates the output flow and effort as well. According to the energy relation, the flow and effort present at any point must have some direction in common. Within the scope of the orthogonality assumptions, fortunately, the directions of effort and flow at any point carrying energy must have the same direction. Therefore, for orthogonal configurations, all the kind-tranformations possible for a given element have the same and the identical orientation-transformation and hence the intermediate elements chosen for constructing a concept will have no ambiguity.

4.1.3 *Direction Synthesis*

The only way direction synthesis problem differs from the orientation synthesis problem is by the sense of the input-output vectors. To frame the problem, it is to derive, given the position and direction of the input vector for an element for an assumed magnitude of its length vector, the combinations of direction of its length vector and that of its output vector. Mathematically, for each possible orientation configuration of an element, there can be eight possible direction configurations. This number is, however, limited by the laws of motion. The mathematically and physically possible configurations for a specific orientation of a shaft and a push-rod have been shown in fig.7.

Once the possible direction-transformations for each primitive element are derived, direction synthesis becomes a mere extension of orientation synthesis.

4.1.4 *Magnitude Constraint*

In conceptual the design phase, there is no apparent way of synthesizing a solution concept based on specified magnitude requirements, neither is there any way of ensuring that a specific concept will or will not satisfy a specified magnitude requirement. Functionally speaking, each primitive element can be made to satisfy any specified magnitude requirement by changing the values of some of its physical parameters. A lever, for instance, can be made to change a force of any magnitude to a torque of any magnitude by adjusting its arm length. It is, however, possible to use the magnitude requirement as a constraint and link it to the design parameters of the concept constituents. This can be done by, first, constructing a magnitude transformation factor for each constituent element as a ratio between the element's input and output magnitudes (expressed in terms of its design parameters), and then, multiplying the magnitude transformation factors of all the constituent elements for the concept. We can constitute the magnitude constraint for a compound lever , shown in the fig.8, by multiplying the magnitude transformation factors L1 of lever1 and 1/L2 of lever2, giving a value of L1/L2. This constraint can be used for deriving or analysing the physical description in the later stages of design.

4.1.5 *Position Constraint*

For some designs, it may be required that the output vector of the desired concept be at a specific position when the input is given at some specified position. Consequently, we need to know, at the earliest possible phase of design, if the above position constraint is satisfied by the candidate concepts. If the magnitudes of the length vectors of the concept's constituents were known, the problem would be as simple as checking if the right hand side of the following equation, which gives the position change achieved, matched with its left hand side, which expresses the position change required (fig.9) :

$$P(O) - P(I) = \sum L(i)$$

where P(O) is the output position

P(I) is the input position

L(i) is the length vector of the ith constituent of the concept.

During the conceptual design stage, however, we do not usually know the magnitude of the length vectors. Therefore, all we can do is to keep the above equation as a constraint to be checked against in the later and better informed phases of design.

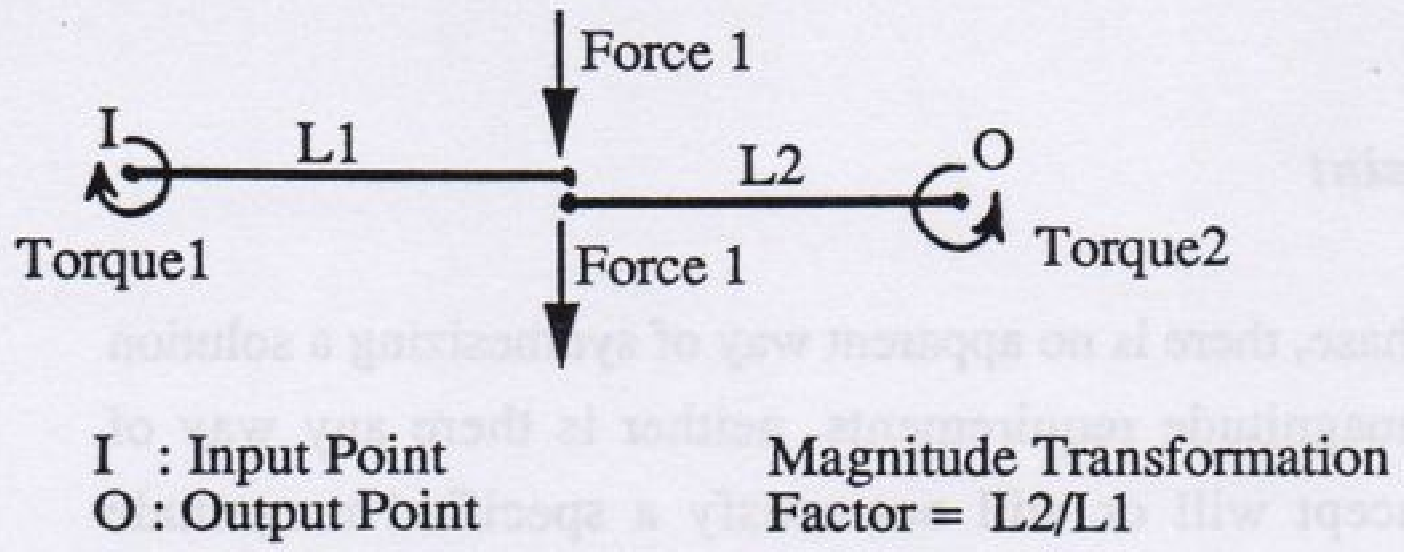
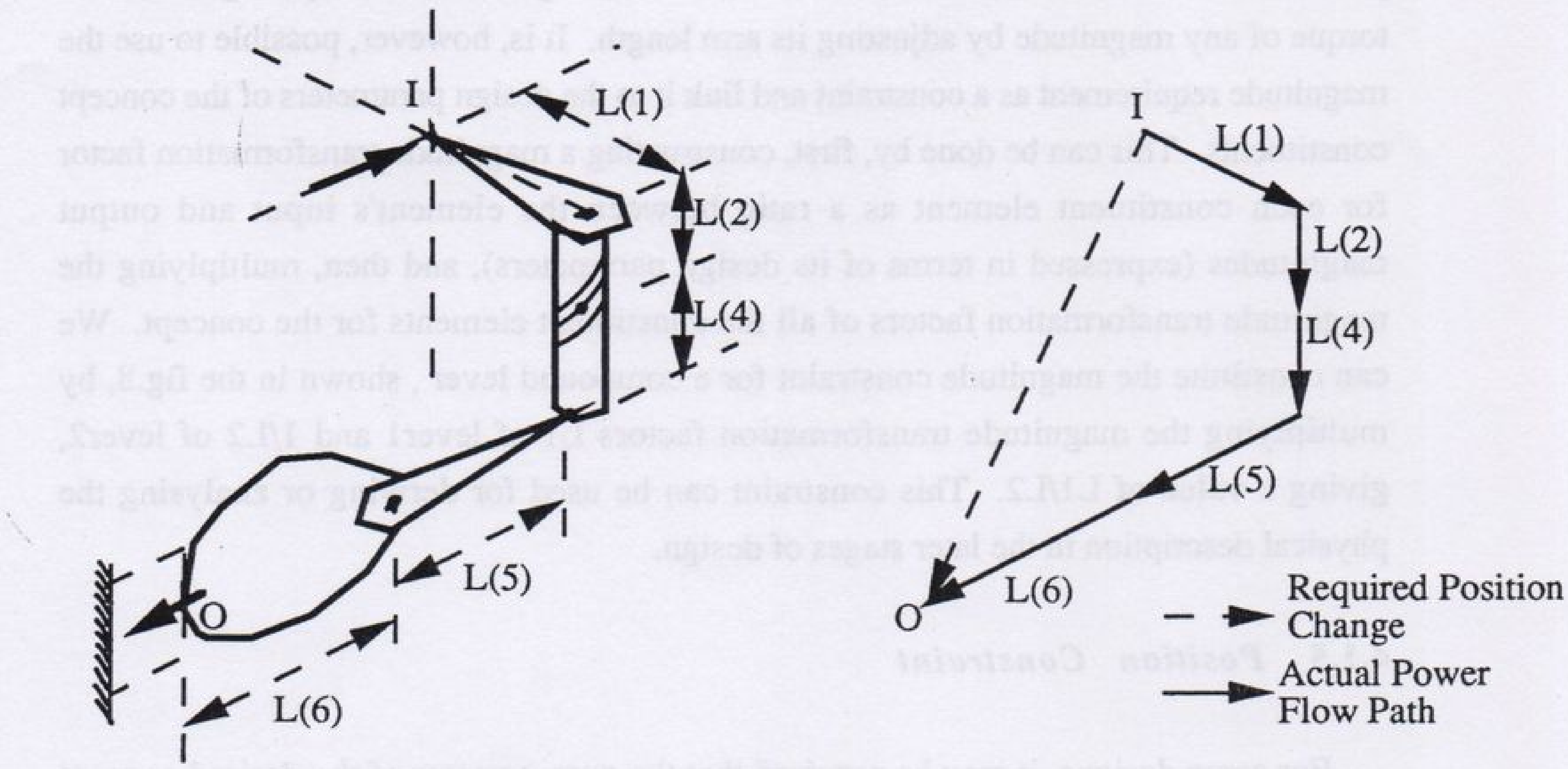


Fig.8 Magnitude Transformation Factor of a Compound Lever



(a) A Solution Concept for Converting a Force into another Force **(b) Position Change of the Energy Vector**

Fig.9 Position Constraint

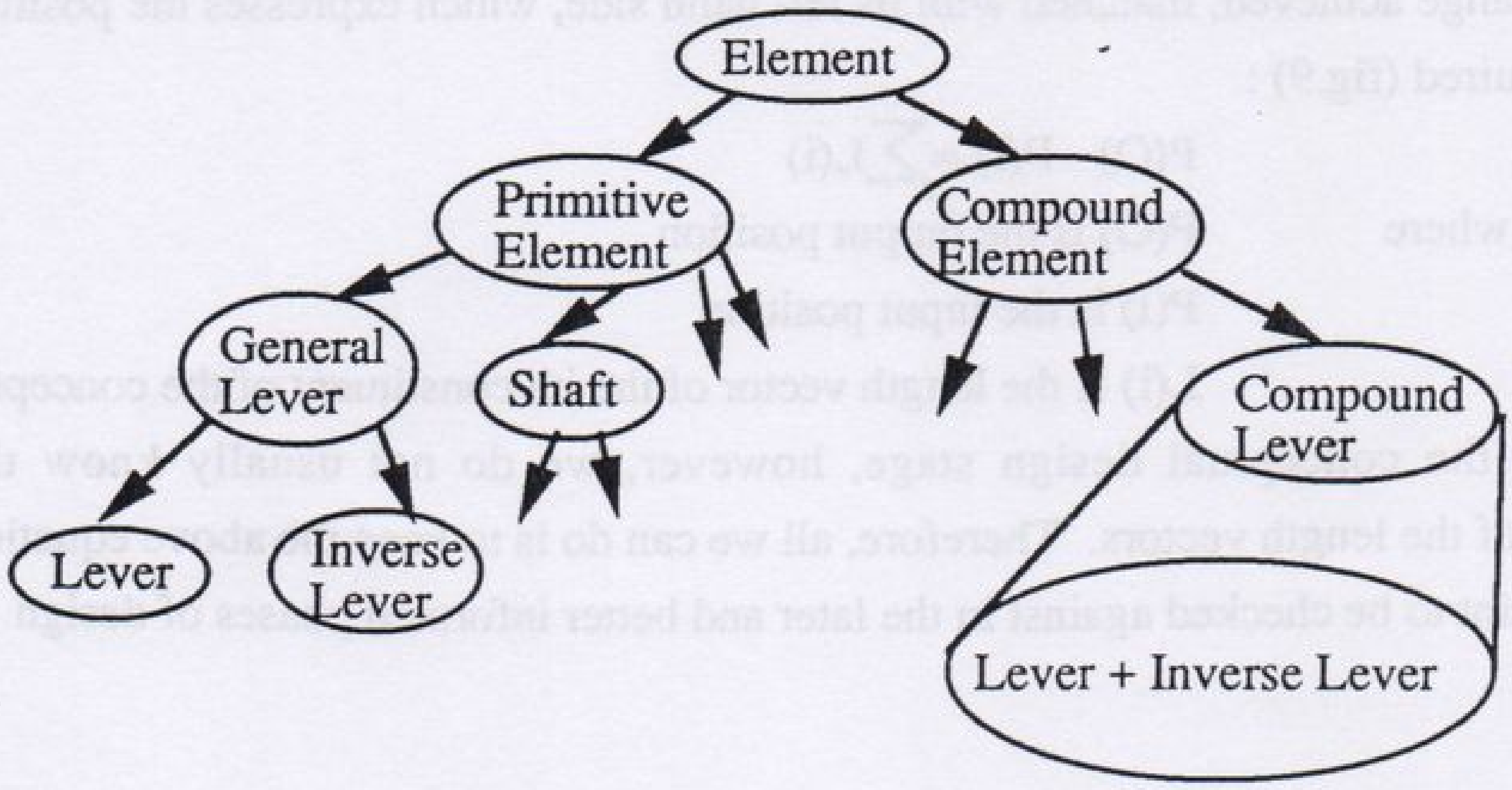


Fig.10 A Part of the Knowledge Base of Elements

5 FORMALISM

The foregoing discussion brings us to a stage where we can identify the pieces of knowledge we need to have in our knowledge base, and, the kinds of procedures we need to have for carrying out the conceptual design. The following subsections elaborate them.

5.1 Knowledge Base

The knowledge base primarily consists of a set of primitive elements that contain information about their permissible input-output kinds, permissible orientation and direction transformations, magnitude transformation factors and input-output feature requirements.

There are three kinds of effort: force, torque and pressure. Correspondingly, there are three kinds of flows: displacement, rotation and volume. These constitute the possible kinds of input and output. Within the scope of the restrictions imposed, all the vectors, whether length vectors or input/output vectors, can have three possible orientations, i.e., along i, j or k directions. There can be two possible senses for each such orientation, i.e., positive (+) or negative (-). There can be four possible orientation transformations between a pair of vectors: parallel and intersecting (PI), parallel and non-intersecting (PNI), non-parallel and intersecting (NPI), and, non-parallel and non-intersecting (NPNI). These are expressed as relations among the orientations of input vector (I), output vector (O) and length vector (L). A PI transformation for input orientation i, for instance, is expressed as (i i i), where the elements of the list respectively represent orientations of I, L and O vectors. The direction transformations are similar expressions, except that their elements represent the senses of the I, O and L vectors as well. The direction transformations for a push-rod will, therefore, be (i+ i+ i+) & (i- i- i-), for an input vector directed respectively in the positive and negative i directions. The Magnitude transformation factors usually come from the element constitutive relations and are stored as functions of the design parameters forming them. The input-output feature requirements are stored as the feature or effect required to input-output transfer, like teeth, friction etc.

Each element can also perform a function that is inverse of its usual function, except in some rare instances. A lever, for instance, can transform a force into a torque just as well as it can transform a torque into a force. This may bring ambiguity. To avoid that, we use them as two functionally distinct but physically identical elements in our knowledge base (fig.10).

5.2 Procedure Base

The procedure base essentially contains three procedures. One procedure allows the designer to describe his problem at various levels of detail, and, stores and brings results to him. This is basically the interface procedure. Another procedure, the central procedure for concept derivation, constructs solution concepts for serving specified function, from the repertoire of elements. The remaining procedure enables the designer to construct his own compound elements by combining elements from the repertoire. The latter two procedures are discussed in the rest of this subsection.

The designer can construct his own compound elements by applying two restrictions. One, he can choose a series of elements ,compatible to each other, to suit his required input and output kind requirements. The other is, he can use orientation and direction constraints on the compound-element input and output as well as between consecutive constituent elements.

Consequently, the above information about the constituent elements and constraints of a compound element remain stored in its knowledge base. The procedure for deriving this consists of a couple of simple procedures that take designer's intent, check the compatibility among the chosen element and create schemata to store them. When a compound element is used in a design, this information is to be used to generate its 'legal' configurations. This is done by, first generating all its possible configurations (by the application of the central procedure to be discussed next), and then, applying on them the constraints to eliminate the 'illegal' configurations.

The function of the central procedure is to construct, by combining elements available in the knowledge base, all possible solution concepts, within the scope of the restrictions made, that would satisfy the input-output requirements specified by the designer. One way to do this is to derive all possible unconstrained conceptual solutions, and then apply on them all the constraints to identify the 'legal' ones. This procedure, however, has two main drawbacks. One is, for a specific input-output kind requirement, there will be, depending on the maximum permitted number of constituents, a number of possible solution concepts. Each of these solutions can have a number of possible orientation configurations. Each such configuration will have a number of direction configurations. If we were to generate the final solutions in one step, the number of solutions could be intractably large. The other problem is, this one step procedure would not give designer the chance to observe, compare and select designs at the intermediate levels of detail, thereby eliminating designer's judgement in the process.

A better alternative would be to split the procedure into a set of independent sub-procedures, each of which would be devoted to synthesizing or analyzing solutions for a single characteristic of input-output. Clearly, there exists a hierarchy of which characteristic is to precede the others in the analysis. For example, we need to synthesize first those concepts that satisfy the design requirement only to the extent of the input and output kinds. The designer can then be given the chance to scrutinize them, and call another procedure, preferably the orientation synthesis procedure to generate the possible orientation configurations for one of the above kind-synthesized solutions. The results can then be left to the designer again, for him to choose one or some of the orientation configurations and feed into some other procedure, say direction synthesis, for producing the direction configurations, and so on.

The procedures for synthesizing or configuring concepts are essentially same, and is equivalent to that described in the sub-section 4.1.1. The difference is, in synthesizing, we look for elements that satisfy the given input-output requirements, whereas in configuring, we look for those specific configurations of the chosen elements which conform with the required input-output configurations. The next section illustrates the procedures through examples.

6 EXAMPLES

The first example illustrates the configuration process of a specified compound element and the second example illustrates the synthesis and configuration process for a specified conceptual design problem.

6.1 First Example

Suppose we have already used the procedure for the construction of compound elements to create a compound element *compound-lever*. The primitive elements used were a lever *lever-1*, which converts a force into a torque and an inverse lever *lever-2*, which converts a torque into a force. The knowledge base for these constituent elements looks like:

name: *lever-1*
input kinds: force, displacement
output kinds : torque, rotation
kind of: converter-transmitter, general lever, primitive element, element
has inverse: inverse lever
orientation transformation: NPNI
direction transformation: [(i+ j+ k+) (i+ j- k-)...]
input feature requirement: .....
output feature requirement: .....

flow magnitude transformation factor: 1/L1
effort magnitude transformation factor: L1

name: *lever-2*
input kinds: torque, rotation
output kinds : force, displacement
kind of: converter-transmitter, general lever, primitive element, element
has inverse: lever
orientation transformation: NPNI
direction transformation: [(i+ j+ k+) (i+ j- k-)...]
input feature requirement: ......
output feature requirement: .....
flow magnitude transformation factor: L2
effort magnitude transformation factor: 1/L2

The knowledge base for the compound lever looks like:

name: *compound-lever*
input kinds: force, displacement
output kinds : force, displacement
kind of: general lever, compound element, element
has inverse: inverse compound lever
orientation transformation: NPNI, NPNI
direction transformation: [(i+ j+ k+) (i+ j- k-)...], [(i+ j+ k+) (i+ j- k-)...]
input feature requirement: ......
output feature requirement: .....
flow magnitude transformation factor: 1/L1, L2
effort magnitude transformation factor: L1, 1/L2
list of elements: *lever-1*, *lever-2*
orientation constraint: perpendicular
direction constraint: positive

There are two points to be explained about the knowledge base of a compound element. The information *orientation constraint* is a list of constraints about how the designer likes each consecutive pair of length vectors to be oriented with respect to each other. Within the scope of this discussion, they can either be perpendicular or parallel. The information *direction constraint* provides a list of constraints on how the directions of each pair of consecutive length vectors relate to each other. The constraint is positive when one length vector oriented in the positive direction along one of the axes constrains the other vector to be oriented along the positive direction of its axis and vice-versa. It will be negative when the direction of one length vector being positive constrains the direction of the other to be negative and vice-versa.

The configurations of the above compound element are shown in fig.11, for an input in the i+ direction.

6.2 Second Example

Suppose we want to progressively derive and configure conceptual solutions for converting a specified input force in the i+ direction to a specified output force in the j+ direction. We limit the repertoire of elements to the following: lever (force -> toque), inverse lever (torque -> force), screw (torque -> force), inverse screw (force -> torque), shaft (torque -> torque) cam (torque -> force) and push rod (force -> force). The solutions yielded by the kind synthesis are:

solution 1: push rod
solution 2: push rod, push rod
solution 3: lever, inverse lever
solution 4: lever, screw
solution 5: lever, cam
solution 6: inverse screw, inverse lever
solution 7: inverse screw, screw
solution 8: inverse screw, cam

Let us choose one of the above solutions and configure its orientations. We choose solution 4, i.e., a device consisting of a lever a screw. Starting with the input orientation i, assuming that all the length vectors are oriented in the positive axis directions, assuming unit length vector magnitudes for the purposes of drawing, and, applying the output orientation constraint j, we get the orientation configuration given in fig.12a. We can now expand the above configuration, starting with the input direction i+ and applying output the direction constraint j+ to yield the direction configurations given in fig.12b.

The program has been implemented on a symbolics computer using symbolics-COMMON LISP functions and the schema system of the package ART (Automated Reasoning Tool).

7 CONCLUSIONS

In this paper we investigate why well defined techniques for carrying out conceptual design are needed and discuss, within the confines of mechanical transmission design, issues of how it can be achieved. We consider the conceptual

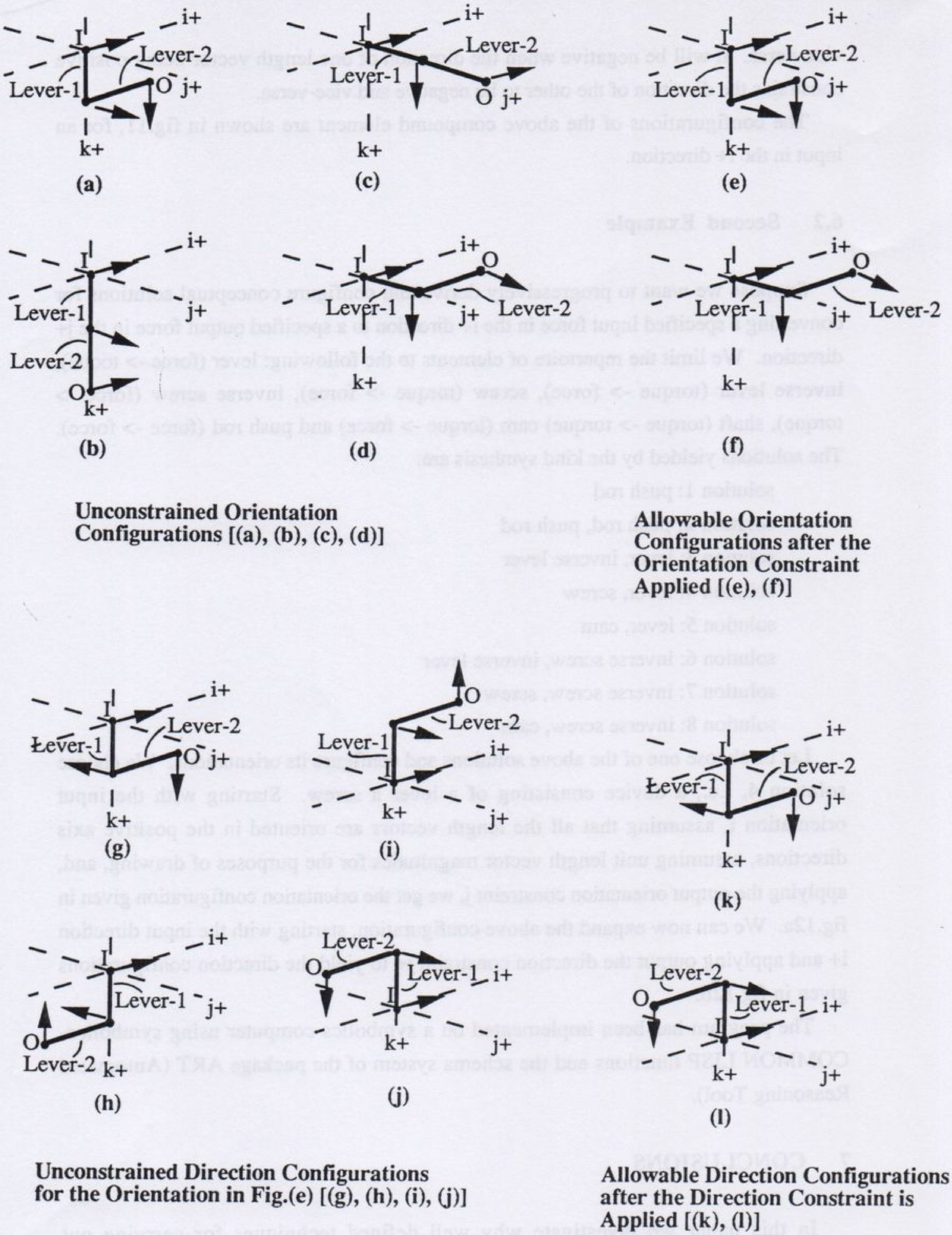
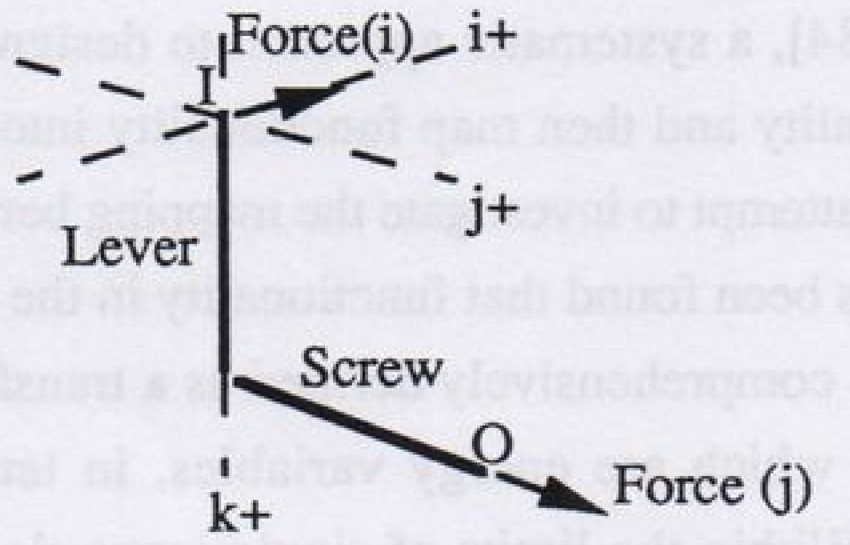
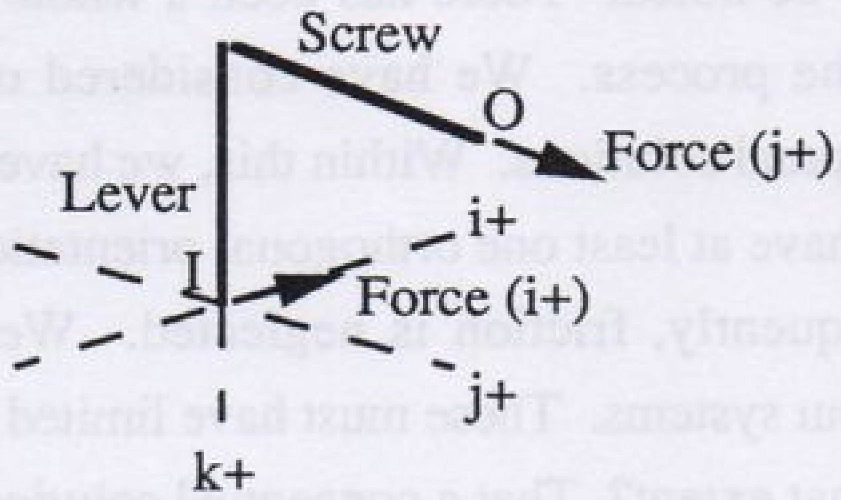


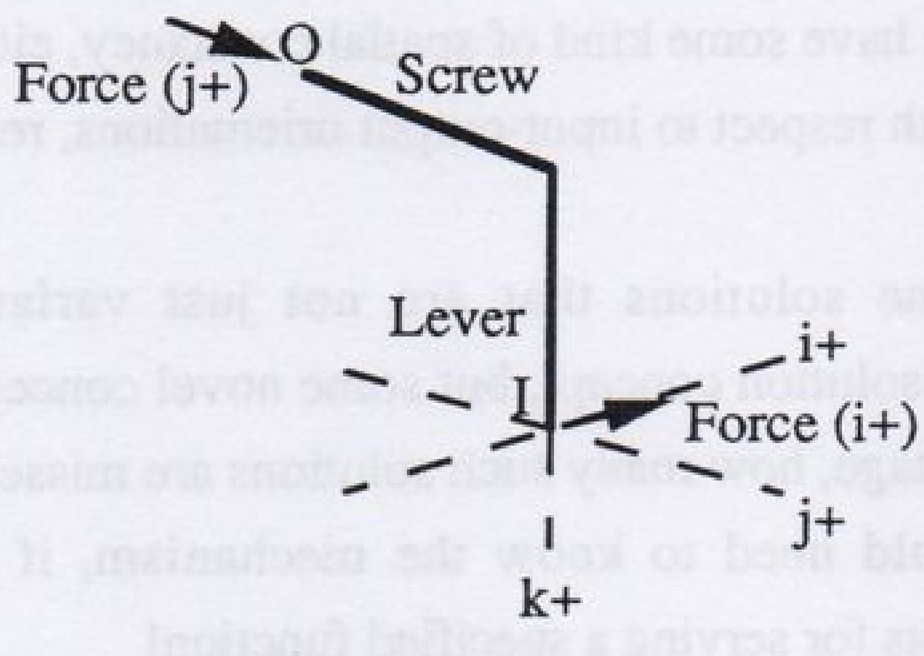
Fig.11 Orientation and Direction Configurations the Compound Lever of the First Example



(a) Orientation Configuration of Solution 4 in the Second Example



(b)



(c)

(b), (c) : Direction Configurations for Input (i+) and Output (j+) of Solution 4, Second Example

Fig.12 Configuration a Solution Concept for force-force Conversion [Solution 4, Second Example]

design phase as a mapping between the requirements of the design problem and the physical concepts that are able to fulfil those requirements. According to one school of thought [Pahl & Beitz, 1984], a systematic approach to design would be to map the requirements into functionality and then map functionality into conceptual solutions. This paper is a preliminary attempt to investigate the mapping between functionality and conceptual solutions. It has been found that functionality in the context of mechanical transmission design can be comprehensively defined as a transformation between the input and output vectors, which are energy variables, in terms of their position, characteristics and time. Within the limits of single input single output orthogonal systems, it has been possible to identify a reasonable number of primitive structures that can be used as basic elements for configuring conceptual solutions from the input-output specification. It has also been possible to construct new composite functional elements by combining the basic elements, and use them in constructing more complex solution concepts.

A few points should to be noted. There has been a whole series of assumptions and restrictions made in the process. We have considered only the instantaneous configurations of the conceptual solutions. Within this, we have configured only those solutions that have or may have at least one orthogonal orientation. Energy dissipation is not considered. Consequently, friction is neglected. We have not considered multiple input multiple output systems. These must have limited the solution-space.

It indeed has. But to what extent? That a conceptual solution may temporally vary in its configuration brings an important point that there would be a great chance of discovering but different configurations of the same concept, were we able to devise a mechanism for configuring all possible solution configurations. The orthogonality assumptions, along with the specification of input-output positions, and, the fortunate fact that most of the physical elements have some kind of spatial constancy, either with respect to input-output positions or with respect to input-output orientations, reduce the possibility of such repetition.

However, we do miss out some solutions that are not just variations in configuration of an already identified solution concept, but some novel concepts. We cannot estimate, unfortunately at this stage, how many such solutions are missed out by our restrictions, as for that we would need to know the mechanism, if any, of identifying all possible solution concepts for serving a specified function!

As far as the issue of further work is concerned, it is sufficient to say that each assumption/restriction made in this paper indicates an avenue for future work. The present work at the Cambridge University Engineering Department involves an extension of the work presented here to incorporate temporal reasoning.

REFERENCES

Beggs, J.S. 1955. *Mechanism*, McGraw-Hill.

Clayton, B.D. 1985. *Inference ART programming manuals*, Inference Corporation, California.

Freeman, P. and Newell, A. 1971. *A model for functional reasoning in design*, Proc. Second IJCAI, London, 621-640.

Freudenstein, F. and Maki, E.R. 1983. *Development of an optimum variable-stroke internal combustion engine mechanism from the viewpoint of kinematic structure*, Journal of Mechanisms, Transmissions and Automation in Design, 105, 259-266.

Grabowski, H. and Benz, T. 1989. *Imlementing the design methodology*, Preprints of the IFIP W.G. 5.2 Workshop on Intelligent CAD, Osaka, Japan.

Hoeltzel, D.A., Chieng, W. and Zissimides, J. 1987. *Knowledge representation and planning control in an expert system for the creative design of mechanisms*, AI EDAM, 1(2), 119-137.

Mostow, J., Barley, M. and Weinrich, T. 1989. *Automated reuse of design plans*, Application of AI in Engineering, 4(4), 181-196.

Pahl, G. and Beitz, W. 1984. *Engineering design*, The Design Council, London.

Rosenberg, R.C. and Karnopp, D.C. 1983. *Introduction to physical system dynamics*, McGraw-Hill.

Scmekel, H. 1989. *Functional models and design solutions*, Annals of the CIRP, 38/1/1989, 129-132.

Shapiro, V. and Voelcker, H. 1989. *On the role of geometry in mechanical design*, Research in Engineering Design, 1(1), 69-73.

Shigley, J.E. and Mitchell, L.D. 1983. *Mechanical engineering design*, 4th Ed., McGraw-Hill.

Ulrich, K.T. and Seering, W.P. 1989. *Synthesis of schematic descriptions in mechanical design*, Research in Engineering Design, 1(1), 3-18.