Abstract: Flexure-based mechanisms, also referred to as flexures, are widely being used as motion-guidance, or bearing, elements in applications requiring multi-degree-of-freedom positioning and alignment. Unlike friction-bearing (such as sliding or rolling contact bearings), flexures can be designed to offer, to a large extent, reliable linear elastic motion with a high resolution (on the order of nanometers) over small ranges of motion (on order of micrometers). Example applications include positioning a probe or sample in atomic force microscopy, alignment of tool and sample in stamping processes, and fine-positioning of wafers and masks in semiconductor manufacturing. These applications are often required to satisfy critical functional requirements, such as load-capacity, bandwidth, resolution, and range. A systematic approach is needed to simultaneously address the design and control challenges involved, starting from the initial design concept generation stage to the final control implementation and testing. In this paper, we present an integrated design and control method for implementing flexure-based nanopositioning systems. We discuss the need for varying design topology and order of a controller in design and control optimization. An automation engine generates a set of flexure-based design topologies and also controllers of varying order in the optimization. A simple 1-DOF example is worked out to illustrate the steps involved in using this methodology. The outcome of the exercise is a novel design topology, with its shape and size optimized, and a controller synthesized such that a desired control bandwidth and design requirements of strength and modal separation are met.

Keywords: Flexure-based mechanisms, Nanopositioning, Topology Generation, Synthesis.

1. INTRODUCTION

Precision linear positioning and angular (rotary) alignment at nanoscale resolutions are often referred to as “nanopositioning.” Many applications for nanopositioning systems have emerged over the past few decades in various contexts, such as semiconductor manufacturing, metrology, x-ray crystallography, and biological imaging. From among the many different methods of implementing nanopositioning systems, those involving compliant flexure-based mechanisms have gained popularity over the years. Flexure-based mechanisms are composed of slender beam-like spring elements in their mechanical design; they are close to being ideal motion bearings with minimal friction, backlash, and other uncertainties. These advantages make flexure-based mechanisms, also referred to as flexures, ideal candidates for precision motion control implementations.

Prime performance requirements for nanopositioning applications include range, load-capacity, and bandwidth. Flexure-based nanopositioning systems have been around for many decades [8]. Kinematics [9], statics [7] and dynamics of flexure-based mechanisms have been extensively studied [11, 33]. However, few publications [12, 13] have appeared in the context of design for dynamic performance. A more challenging and critical requirement is achieving an overall desired closed-loop control performance [15] of a system assembled with the flexure-based mechanism, and suitable actuator and sensor subsystems. To the best of our knowledge, an integrated approach for the design and control of flexure-based nanopositioning systems is lacking in the existing literature. A common systems-based methodology can facilitate developing valuable synthesis tools for achieving the desired closed-loop control performance.

In this paper, we tackle the “co-design” problem, integrating design and control for achieving a desired closed-loop control performance of flexure-based nanopositioning systems. In Section 2 we provide a detailed review of rele-
vant literature that tackle the co-design problem, while (i) motivating the need for co-design from two practical servo-

hardware examples, and (ii) highlighting the deficiencies in current approaches in the field of nanopositioning sys-
tems. A novel method for integrated design and control is presented in Section 3, and tailored for flexure-based
mechanisms. A detailed set of steps needed to implement the method is analyzed in Section 4. The paper concludes
with a summary in Section 5. The reader is referred to Part II of this paper for an application case study of a
1-DOF alignment mechanism that is worked out in detail using the proposed method.

2. INTEGRATED DESIGN AND CONTROL

Integrated design and control has been an active area of re-
search spanning applications such as robotic manipulator
design and control [17]-[22], motion stages developed using
lead-screw drives [24], passive and active vibration isolation platforms [25, 26], and chemical process control [27].
In this section, we cover a detailed survey of relevant
methods in the literature.

2.1 Varying Design and/or Control Parameters

In what follows, we first review works reported on op-

timizing a design (plant) or controller so that a desired
performance metric (design or control) is met under phys-

ical (design) constraints, and state/output and control
constraints.

Optimal design and control of flexible structures has been studied for (i) improving a mass efficiency metric (defined as mass moved per unit work output) in [3], (ii) a quadratic control performance index in [6], and (iii) a weighted sum of structural mass and the energy of the controlled mechanism in [26]. The integrated design and control prob-

lem was formulated as a multi-objective optimization in-
volving design and proportional-integral-derivative (PID)
controller parameters in [28] for mechatronic systems. A
similar approach optimizing proportional-derivative (PD)
controller parameters and design parameters for four-bar
linkages was studied in [18]. A non-linear optimization
formulation including design costs and a robust perfor-
mance constraint on the weighted sum of sensitivity and
complementary sensitivity functions is considered for a
chemical distillation column in [27]. Decentralized control


techniques were used for the optimization of passive
(design parameters) and active (control parameters)
for vibration isolation platforms in [25]. Different
approaches for integrated design and control have been
studied from an optimization theory standpoint in [17]
and [15]. These approaches include (i) sequential optimiza-
tion with design optimization followed by control opti-
mization, (ii) simultaneous design and control optimiza-
tion, and (iii) an iterative combination where the design
is initially optimized without affecting the controller, then
the controller is optimized, and such a cycle is iterated
until performance requirements are met.

Optimal locations for embedded actuators and sensors in
a mechanism with distributed compliance are discussed
in [16] for satisfying controllability and observability condi-
tions. However, neither the design of the controller nor the
influence of a poor design choice on control performance is
addressed in this reference. A related critical issue is one
of lightly damped flexible modes of flexure-based mech-

anisms. Physical damping is low in flexures made from
metals such as aluminium (used in development stages
of the design process for ease of machining), or titanium
(used in the implementation and testing phase because of
its high fatigue strength and other material properties).
External damping such as squeeze film damping and foam-
damping have been suggested and explored for flexures in
the past. Active damping through appropriate selection
of control strategies needs to be addressed to tackle the
lightly damped resonances in these structures. Since the
level of damping in an assembled mechanism is hard to
predict before the fabricated product is available for test-
ing, it becomes necessary to iterate the design process with
thorough system identification and testing of hardware
mechanism implementations.

Motion stages developed using lead-screw drives were char-
acterized for their dynamics and controlled with classical
lead-lag compensators in [24]. In this reference, the design
and control performance space in terms of performance
requirements, such as (i) the positioning error and (ii)
control bandwidth of the drive and (iii) the maximum
acceleration of the carriage, were captured for the entire
range of geometry, material, and other parameters. Since
lightly damped harmonics hinder control performance,
achieving robust passive damping with foam-based ma-
terials is proposed by the same research group in [38]. An
integrated design and control methodology for high-speed
control of robotic manipulators is presented in [21, 22].
Since unmodeled dynamics in the control bandwidth can
adversely affect the performance, it is necessary to account
for model-truncation errors in the design and control op-
timization. In this context, a constraint condition on the
Hankel norm of the truncated modes is formulated in the
optimization problem [21].

2.2 Varying Design Topology

Unlike most of the methods reported above, few refer-

cences address changing the design structure or configu-
ration (referred to as the topology) itself, so that control
performance is enhanced. We examine here two specific
cases from the literature that illustrate the importance of
selecting an appropriate design topology before deploying
any optimization routine.

Consider the example of a robotic system shown in Fig. 1
addressing the end-point control of a flexible link. The
actuator is a rotary servomotor that generates a torque
required for moving the end-point of the link. The feedback
signal is the end-point position, which can be recorded
by a sensor such as an accelerometer. Since the actuator
and the sensor are not at the same location in space,
Fig. 1. Design for control example from [36]. (a) Moving the torque application point away from the hub and closer to the end point of the flexible manipulator results in minimum-phase dynamics, and hence allows for higher control bandwidths. (b) A belt transmission is used on a motor to vary the location of the torque application point. i.e. the system is non-collocated. For the non-collocated system, the flexibility of the link is known to cause non-minimum phase zeros in the transfer function between the voltage applied to the motor and the measured end-point displacement [36].

In order to avoid the occurrence of the non-minimum phase zero, the actuation point shown in Fig. 1 (a) can be moved away from the motor closer to the end-point, as shown in Fig. 1(b). With the actuation location moved closer to the end-point, the portion of the link from the new actuation point to the sensor location is shorter, and hence stiffer. It is shown in [36] that, under certain geometry conditions, this topology change results in moving the zeros from the real-axis on to the imaginary axis, making the system minimum-phase. The design topology change shown in Fig. 1(b) is implemented in Fig. 1(c) using a cable transmission from the motor.

Without this topology change, with the actuator just as the motor and sensor at the end-point, the system would be non-minimum phase and pose critical control challenges.\(^2\)

\(^2\) The constraint on control bandwidth imposed by non-minimum phase zeros is worked out for an example positioning system in Part II of this paper.

---

We next consider the example of a hard disk drive actuator subsystem in Fig. 2. As shown in Fig. 2 (a), this subsystem positions the read (or write) head at the end of an arm pivoting about a rotary bearing. A lorentz-force \(F_m\) generated by voice coil motor at an distance \(R_e\) causes the arm to rotate about the pivot. However, the applied force \(F_m\) also exerts a force \(F_r\) at the bearing, exciting its translation mode. The displacement at the read head is composed of the difference of modal responses arising from the rigid body rotation and the bearing translation mode.

The presence of the bearing translation mode is undesirable for two reasons: (i) the translation shows up in the displacement at the read head and (ii) the transfer function between the applied force and the measured displacement at read head can be non-minimum phase under certain geometry conditions [37]. A novel actuator (see Fig. 2(b)) based on a set of magnetic arrays called Hallbach arrays is designed in [35] to form a voice coil motor that generates only a torque and now net translational force. The new design topology is shown in Fig. 2(c) with the purely-torque motor mounted in the pivot itself, without the need for the linear force \(F_m\) applied at the arm distance \(R_e\).

Without this design topology change, the translation of the bearing and the non-minimum phase zero would limit the performance of the read head.

In summary, the two examples discussed above emphasize the need for developing suitable design topologies before any optimization is attempted. An interesting extension of this problem is one of identifying a set or library
of topologies from which we can select an appropriate topology.

In what follows, we discuss our integrated design and control method that is based on optimizing over a library of topologies, not just dimensional (and other) parameters within a given topology.

3. PROPOSED METHOD

Based on the examples of integrated design and control described in Section 2, we identify the four possible cases for integrated design and control in Table 1. As indicated in the table, in any design for control approach, the design (or plant) and the controller need to be judiciously chosen in the problem formulation step. The options listed in the table are based either on a fixed or a varying topology/order for a design/controller. Before we proceed any further, we present our definitions of these terms as relevant to the integrated design and control methodology we will propose shortly.

A fixed controller is one with a pre-specified order and parameters to be selected appropriately. A basic knowledge of the plant dynamics can facilitate a nominal choice for the controller order. The problem of maximizing performance reduces to selecting the best possible controller parameters. However, such a fixed structure for the controller limits in most cases the freedom in maximizing performance. On the other hand, we define a varying order controller as one in which the order is not pre-selected. Rather, the controller order evolves in the integrated design and control iterations.

Similarly, we define a fixed design topology as one in which the overall structure is initially chosen and the iterations are performed to tune the parameters. Tuning the parameters of a fixed design cannot alter the design structure at all. On the other hand, by a varying design topology option, we iterate with changes in design topology, or configuration, each time creating a different design altogether. For example, a design topology can be varied to go from a parallel kinematic design to a serial kinematic design, or from an exact-constraint design to an elastic-averaging configuration, each time creating a different design altogether.

Table 1. All possible cases for integrated design and control.

<table>
<thead>
<tr>
<th>Case</th>
<th>Design Topology</th>
<th>Controller Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Fixed</td>
<td>Fixed</td>
</tr>
<tr>
<td>II</td>
<td>Fixed</td>
<td>Allowed to Vary</td>
</tr>
<tr>
<td>III</td>
<td>Allowed to Vary</td>
<td>Fixed</td>
</tr>
<tr>
<td>IV</td>
<td>Allowed to Vary</td>
<td>Allowed to Vary</td>
</tr>
</tbody>
</table>

The cases presented in the table work out as follows. In Case I, the integrated approach optimizes performance over a combination of design and controller parameters. The final outcome after the iterations is a design and a controller of the same structure as at the beginning of the iterations, but with the selection of the most promising parameters. Hence, in this case, both design and control structure are fixed and cannot be altered. A poor choice of design topology or controller structure can leave critical performance requirements unfulfilled.

In Case II, for a fixed design topology, the controller is allowed to vary. Hence, in the iterations, the design parameters and controller order evolve to facilitate optimizing the performance.

In Case III, the design topology is varied (i.e. many possible design structures are tested) against a controller with a fixed structure. This case fundamentally limits the performance, since for every new plant we are confined to the same controller type. The potential performance of the system can be lost in thus fixing the control structure. The most intuitive and useful case is Case IV, which uses a varying the design topology and the order of the controller. However, since the number of possible design configurations in typical nanopositioning system applications are finite, the varying design topology problem can be broken down into a number of fixed design (each tested with a controller of varying order) problems. Hence, we formulate our methodology on Case II with a controller of varying order tested for each fixed design topology of all possible design topologies.

4. DETAILED STEPS

In this section, we detail the steps involved in implementing the integrated design and control approach based on Case IV of Table 1, in which both the design topology and the controller structure are allowed to vary. As discussed above, we simplify this to the one of Case II for a varying order controller tested for all possible fixed design topologies from a design library. The methodology is illustrated as a flow chart diagram in Figure 4.

Fig. 3. A performance-driven design library shown as constructed from building blocks prepared by performance-driven operations on a set of primitives. Novel designs synthesized with this method are schematically shown in the Design Library block.
Step 2: Design Topology Library Generation: The following actions are involved in automating generation of topology concepts that improve the specified performance requirements. (i) First, a library is set up with a set of primitives. (ii) These primitives are subjected to a finite number of operations dictated by performance requirements. These operations could be, for example, a parallel or serial replication, or a geometrical transformation, or adding a redundant constraint that imparts symmetry. (iii) The primitives are then subjected to these operations generate building blocks that meet the desired performance requirement. (iv) Once the building blocks are generated, a library of design topologies can be generated by using the building block as an implementation of the constraints (following a constraint-based synthesis approach [29]) for satisfying the necessary kinematics. In a nutshell, using the performance-tuned building block allows to meet a strength, or modal performance criterion, while the constraint-based arrangement allows for satisfying the required kinematics. This step is detailed for the example 1-DOF positioning system in Part II of this paper.

Step 3: Design Topology Selection/Screening: Every nominal design topology in the topology library is subject to a screening test to eliminate design topologies that obviously do not meet critical requirements. This screening test is necessary before (blindly) feeding the design topology to a shape and size optimization procedure.

Step 4: Controller Selection/Screening: On the controller side, an initial controller is selected as a nominal controller from the entire class of stabilizing controllers for the screened nominal design topology. For the screened nominal design and the nominal controller selected above, a screening test is used to weed out controller choices that do not allow for critical requirements to be met. It is important to perform this screening test before (blindly) feeding the design to an optimization procedure. The structure of the nominal controller is revised until it passes the screening test.

Step 5: Optimization: Given that the nominal design and the nominal controller have passed the screening test, we now feed them to an optimization procedure. This procedure collects the design and controller parameters and optimizes them for an objective function defined by the user. The design optimization may target shape and size optimization of the chosen topology. The controller optimization varies the order of the controller to meet the robust stability and performance specifications on sensitivity transfer function or complementary sensitivity transfer function. Many choices exist for implementing the design and control optimization. As we discussed in Section 2, different approaches for the optimization are discussed in [15]. These include (i) simultaneous design and control optimization (ii) sequential optimization, with design optimization followed by control optimization, and (iii) iterative design and control optimization. While each of these approaches could be applied in our methodology, we select the option (ii) of optimizing design first and then optimizing the controller for the optimized design since (i) it comes closest to what is done in practice when designing and controlling a hardware positioning system and (ii) it is computationally less intensive and feasible. The design optimization is formulated to select the design parameters that best allow minimizing or maximizing a desired cost function while meeting design constraints such as stress, fatigue limits. This part of the design optimization is often referred to as shape and size optimization. The optimized plant is fed to the controller optimization block. Once a nominal controller is chosen, it can be enhanced for imparting robustness. If the performance requirements are met by the outcome of the optimization procedure, the controller is tested on the hardware to see if the performance can be demonstrated. If the performance requirements are not met at the end of the optimization procedure and the maximum number of iterations has not been reached, the nominal design topology is revised. Unless the nominal design topology is revised it is impossible to achieve the desired performance. If the maximum number of iterations has been reached, the only way to proceed any further is by relaxing the performance requirements. The design intuition gained from the optimization or from the hardware application should be used to revise the performance requirements suitably, taking us back to Step 1 listed above.
5. SUMMARY

In this paper, we presented a method for iterating on design (plant) topologies and controller order to achieve a desired closed-loop system specification. It is emphasized that iterating a design is not just about fine-tuning shape and size of a particular design configuration. Instead, we need to iterate over design topologies and controller order. An automated topology generation engine is discussed. Further, a novel controller parameterization is used to vary the controller order while directly tuning the sensitivity function to a desired form. An example of a flexure-based 1-DOF positioning system is worked out in Part II of this paper as an application of the integrated design and control methodology.

Acknowledgements:
The first author is thankful to Xerox Foundation and MIT Dean School of Engineering for fellowship support. We are thankful to Dr. Pradeep Subrahmanyan of Data Physics Corporation for discussion of design and control interactions in precision machines.

REFERENCES


