CONF-961105--18 SAND96-1949C

WHY MECHANICAL SUBSYSTEMS ARE DIFFICULT TO INTEGRATE

Daniel J. Segalman and Keith Ortiz
Sandia National Laboratories, Albuquerque, NM
John J. Wesner
Lucent Technologies, Middletown, NJ

ABSTRACT

Though the theme of System Engineering is integration, and it is normal to attempt in integration to ignore the lines between disciplines, there are distinct characteristics of the mechanical design portion of any major system design project that make this difficult. How these characteristics compound the difficulty of integration is discussed and means to minimize the associated obstacles are suggested.

INTRODUCTION

The process of systems engineering is often thought of as having two phases: decomposition and integration. In the decomposition phase, written specifications allocate the system's functions to subsystems and components. In the integration phase, the components are assembled and the proper functioning of the system is assured.

It is common for system functions to be partitioned along disciplinary lines so that mechanical functions, such as, structural support, are assigned to their own subsystems. Mechanical subsystems have a reputation for being difficult to systems engineer, especially in the integration phase. Whether this is deserved, it is true that just about every component has a mechanical interface, which means that whenever there is a mechanical problem the impact of design changes can ripple throughout the entire system.

It is our contention that these difficulties are due to the nature of mechanical system problems. This paper identifies three common problems and their causes, and it offers some solutions.

SURFEIT OF CHOICES

Mechanical subsystems have tremendous "opportunity distance" between the functional design and the final design. Having told a mechanical engineer what the component needs to do, one often has no idea what the solution will look like.

- For instance, if the required function is to actuate a control, this can be achieved by linkages, hydraulics, pneumatics, or electro-mechanical methods. One might even choose among exotic methods involving "smart materials," such as, shape-memory alloys or polyelectrolyte gels.
- Similarly, if the required function is to convey material from one location to another, this can be achieved by rollers, wheels, liquid flotation, air cushions, or magnetic fields.

Ordinarily, one would think that having lots of choices is a good thing. However, unconstrained freedom can lead to problems. For example, the mechanical designer might inappropriately select an electric motor for use in an environment that is occasionally filled with a potentially explosive gas. This can happen because many systems engineers are unfamiliar with the scope of the mechanical design space and assume the mechanical solution will come from an implicit "hotbox" that is much smaller. ("Of course, the mechanical solution will be non-electrical....") Meanwhile, many mechanical engineers are unaware of the broader system context of their designs beyond nominal operating conditions. This combination can be deadly.

The basic problem here is that the system engineer and the mechanical engineer may not be thinking of the same design space when the mechanical subsystem specification is written. Hence, there is no meeting of the minds.

One solution to this problem is for the system engineer to over-specify the mechanical solution. That is, to eliminate surprises, the systems engineer might over-constrain the design space, for instance, by specifying hydraulic actuators, leaving only the sizing of the

The portion of this work performed at Sandia National Laboratories was supported by the United States Department of Energy under Contract Number DE-AC04-AL85000.

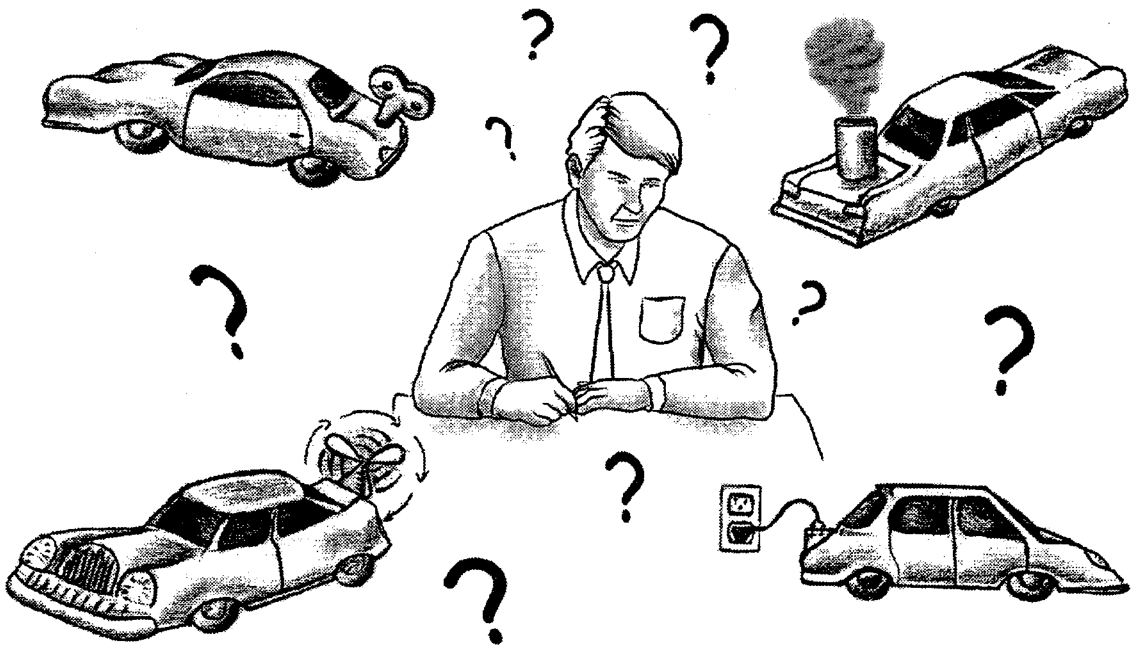

Figure 1. The tools available for mechanical design, with often overlapping functionality, include electromechanical (such as motors and solenoids), pneumatic, hydraulic, and purely mechanical (such as springs, dampers, rollers, and deforming bodies.

components to the designer. However, this is a poor systems solution because it can easily exclude much better designs.

A better solution is for the systems engineer and the mechanical engineer to negotiate the specification before it is written. The mechanical engineer must educate the systems engineer about potential solutions in the design space; and the systems engineer must in turn educate the mechanical engineer about the true needs of the system, including the mission and operational environments. Prototypes, mock-ups, and computer models are useful tools in this negotiation.

UBIQUITY OF NON-LINEARITY

Mechanical subsystems are almost always non-linear. The non-linearity may be in kinematics, in dynamics, or in material properties. Most mechanical assumptions of linearity hold only for small perturbations in the design space. The behavior of the subsystem away from a particular point design is often difficult to predict without creating a new model.

An illustration is the problem of predicting vehicle crash-worthiness. It is known that adding metal under the hood can increase crash-worthiness and some computer codes can predict the relative contribution of one design feature over another. However, crash-worthiness is typically determined by an expensive series of crash tests of a prototype into a rigid wall. Significantly altering the design requires more such experiments to calibrate and verify the models. Thus, solving design problems of this sort requires several iterations of design, analysis, and experiment. This forces the other vehicle components to be integrated iteratively also. Other examples can be drawn from vibration, manufacturing processes, heat transfer, and so on. Even the routing of pipes is non-linear.

Other common examples of non-linearity are:

- Forming metal parts by forging or bending requires deforming the original parts far beyond the linear "elastic" region of deformation, yet there will be some elastic "spring back". It is difficult to determine how far to deform the material to achieve the desired final shape - if it can be done at all (as with drawing a deep and narrow seamless can).
- If a spring-loaded component bounces away from the spring at any time during its travel, or impacts some fixed part of the system, linear assumptions no longer apply.
- A seemingly simple mechanism composed of links ceases to be so simple (and linear) if the loads are sufficiently high that the links deform more than an infinitesimal amount. Making the links more substantial to reduce the deformations may increase inertia loads enough to affect other performance parameters of the mechanism - creating other non-linearities!

The problem here is that non-linearity can force mechanical design changes to impact other system components resulting in new layouts. This can potentially change the environments (vibration, for example) experienced by the other components and subsequently force their

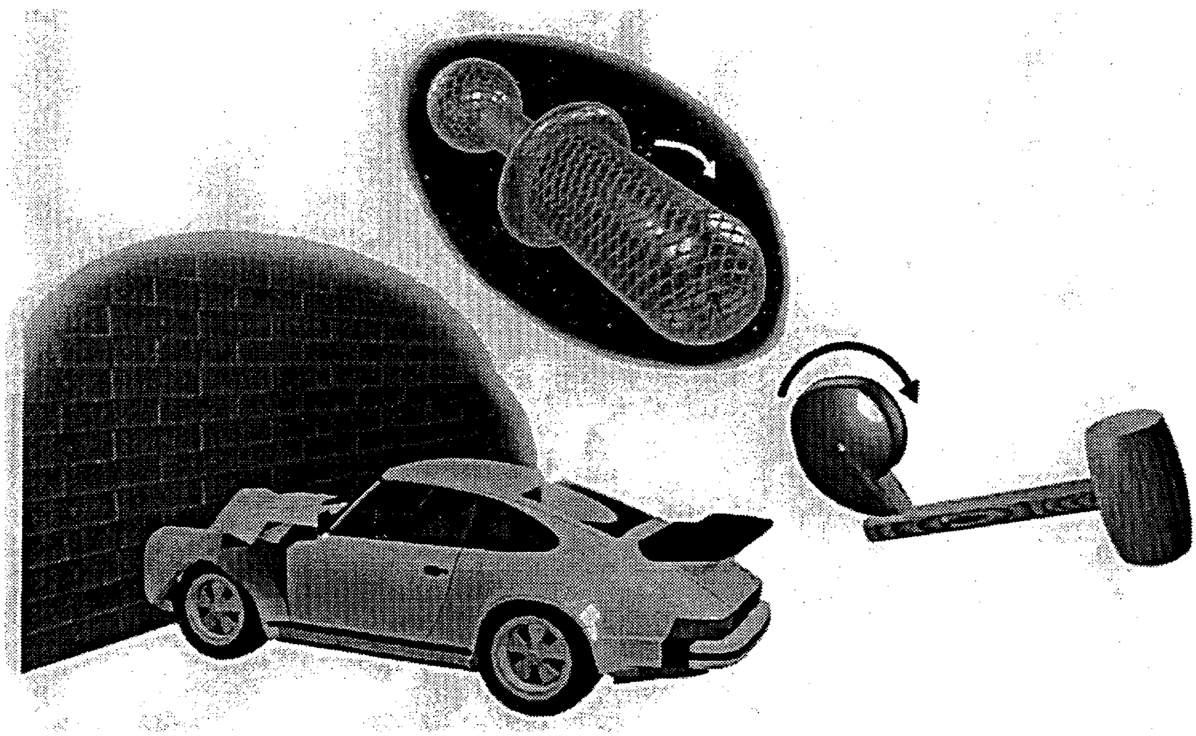

Figure 2. Nonlinearity is ubiquitous in mechanical systems and subsystems. Material nonlinearities dominate problems where impact tolerance is a design criterion; dynamics of rotating rigid bodies is governed by the (nonlinear) Euler equations; and even simple linkages such as the trip hammer shown are governed by nonlinear kinematics.

redesign and re-qualification.

One solution to this problem is to get closer to the final design with the first iteration. Where possible, this can be advantageous in many dimensions, including complexity, time, cost, quality, and reliability. The tried and true way to do this is to start from an existing design that works, and slowly evolve the design. Of course, there will be cases where there is no adequate existing design, or where innovation is clearly called for.

Another way to accommodate the nonlinearity of mechanical subsystems is to provide plenty of "room" to allow the mechanical design to change--a luxury that is not always available; system design requirements or resource limitations may make this impractical. Perhaps the best way of all is "concurrent engineering,' where the mechanical elements of the system design are developed along with the rest.

Better solutions are becoming available. Non-linear computer modeling is within the reach of many designers, even for the difficult crash-worthiness problem. We are learning to use rapid prototyping and testing using plastic models.

HIGH DEGREE OF COUPLING

Mechanical subsystems are much more likely to be coupled in non-functional ways than subsystems of other types (except perhaps for radio-frequency electronics). Among such couplings are shock and vibration transmission (including sound), mass coupling (i.e., center of gravity and moments of inertia), and thermal coupling.

An example is the mass coupling of a spinning space structure. Changing or moving the mass of a component requires other components to be moved or modified in order to maintain the same center of gravity and moments of inertia and to preserve stability. The mechanical engineer must manage the system as a whole.

Coupling also occurs with non-mechanical systems, but the same systems engineering solutions are not available to the mechanical engineer. For example, in radar operation, the pulse transmitted is often strong enough to overload the sensitive receiver electronics. The radar engineer can usually de-couple the problem by turning off the receiver from the time when the pulse is transmitted until the surrounding environment "rings down". Unfortunately, mechanical subsystems cannot be de-coupled so neatly. (For example, it is difficult to make mass appear and disappear on command.)

The solution is to learn to live with the coupling. In general, coupling must be managed at the systems level. This could imply that there should be a mechanical engineer on the systems engineering team.

CONCLUSION

The problems we have identified are not unique to mechanical

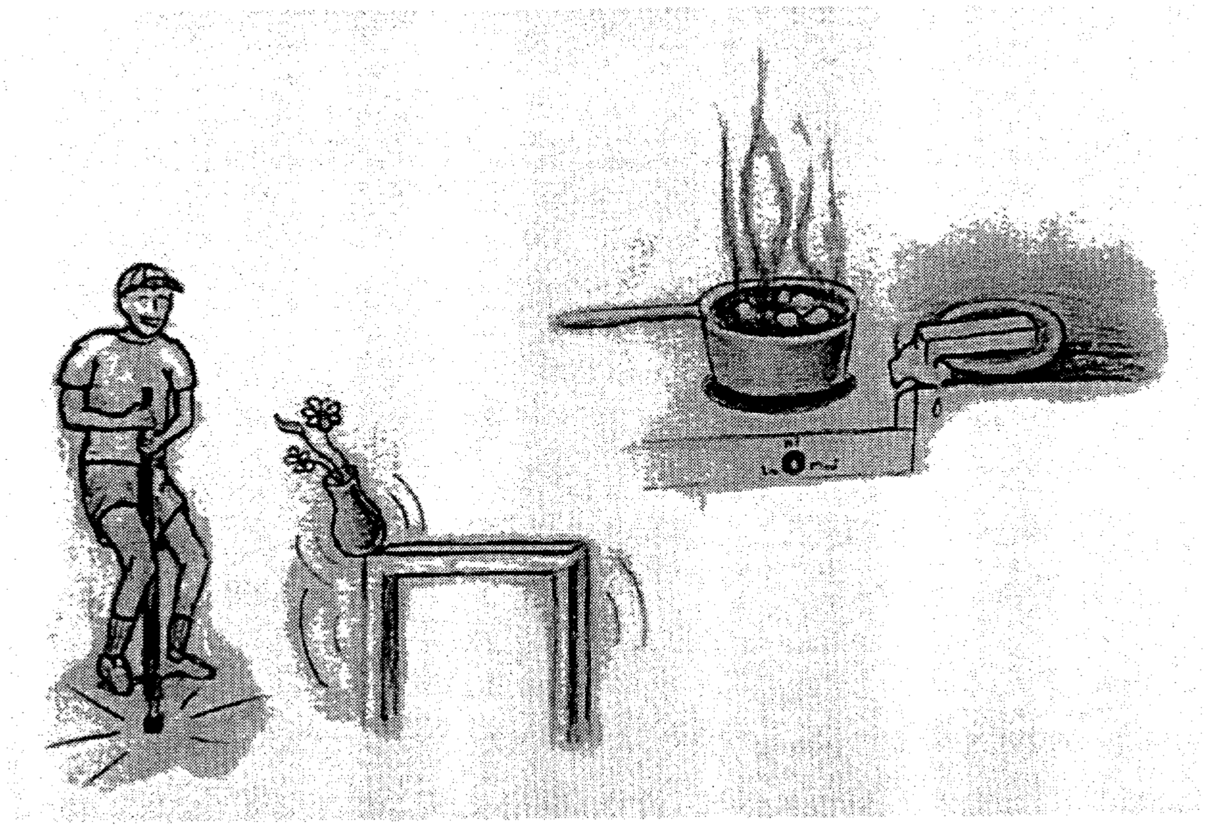

Figure 3. Coupling of one sort or another between mechanical components and subsystems is the natural state and specific isolation design is necessary to prevent that coupling. Among the most commonly encountered couplings are vibration, mass, and thermal coupling.

subsystems or components. They are in fact different versions of three of the biggest issues in systems engineering:

- How to negotiate subsystem or component requirements and how much detail to specify?
- How to select a baseline to minimize iteration?
- How to manage system functions or attributes when de-coupling is not feasible?

What makes mechanical subsystems more difficult is the fact that almost all of them have all these problems at the same time. Further, changes to mechanical subsystems tend to ripple throughout the whole system.

Systems engineers, please don't shoot the mechanical engineer! Work closer together, perhaps embrace the mechanical engineer on the core systems engineering team.

Mechanical engineers, learn the language of systems engineers and educate them.