From: Proceedings of the Twelfth International FLAIRS Conference. Copyright ' 1999, AAAI (www.aaai.org). All rights reserved.

Automating Mechanical FMEA Using Functional Models

Nigel Hughes, Enxi Chou, Chris Price, Mark Lee

Department of Computer Science
University of Wales, Aberystwyth
Ceredigion, SY23 3DB, United Kingdom
e-mail: nnh@aber.ac.uk, eec@aber.ac.uk, cjp@aber.ac.uk, mhl@aber.ac.uk

Abstract

This paper considers the application of functional modelling to the automated production of FMEAs for mechanical systems. We consider how a functional model can be generated algorithmically from the geometric and assembly data already present for a device in a CAD/CAM system. We present the functional model used for representing the mechanical system, and propose reasoning techniques that can be applied to the model in order to produce an FMEA.

Introduction

The GENMech project is an investigatory project considering the feasibility of automating FMEA production for mechanical systems using functional modelling techniques. The work is based on other successful work on automating FMEA production for electrical systems (Price et al. 1995; Pugh and Snooke 1996) using qualitative reasoning techniques. Traditional quantitative modelling of mechanical systems is inappropriate for this task for two reasons:

1. Complex mechanical systems are computationally very expensive to model, and suffer from a combinatorial explosion in model complexity.
2. The output from conventional modelling systems (for example the load, force, or torque on a particular component) are not suitable for the natural language descriptions of a system's behaviour required for FMEAs.

This paper discusses the requirements for automated FMEA production for mechanical systems. We also present a novel functional modelling scheme that has been designed to meet the particular requirements of FMEA production. GENMech attempts to minimize the effort required by an engineer when producing an FMEA. As a result, geometric and assembly data generated by AutoCAD and Mechanical Desktop is used to automate model construction activity.

Failure Modes and Effects Analysis

Before discussing the techniques used to automate Failure mode effect analysis (FMEA) production, it is important to consider how and why FMEA is currently performed (MIL-STD-1629A 1980; BS5760 1991; Ford Motors 1995; Chrysler, Ford and General Motors 1995). Producing an FMEA of a design involves the investigation and analysis of all possible failure modes of the system being designed. For mechanical design, this involves examining what would happen if any component in the system fails unexpectedly. This kind of analysis is of growing interest in the automotive (Ford Motors 1995; Chrysler, Ford and General Motors 1995), aerospace, and other advanced manufacturing industries (MIL-STD-1629A 1980; BS5760 1991), where increasingly complex electrical, electronic and mechanical systems are being combined.

Automation of FMEA is an attractive proposition, as it is a tedious and repetitive task, yet one that is time consuming and must be done by skilled engineers (Onodera 1997; Dale and Shaw 1990; Bowles and Bonnell 1994). Previous research on automating mechanical FMEA has generally focused on the housekeeping tasks associated with the FMEA production, recording previous completed FMEAs and presenting them for integration into a new FMEA. The work on the GENMech system, described in this paper, attempts to produce automatically an FMEA from original CAD data.

Mechanical FMEA is made more difficult by the wide range of domains covered, including Kinematics, Fluid Dynamics (including hydraulics, pneumatics and tribology), Dynamics, EM emission and field properties. The initial prototype software being developed for GENMech focuses on kinematics because the information that can be automatically extracted from a CAD/CAM application is richest in this domain area.

Why use Functional Modelling?

Although there is a wide range of software available (such as Working Model 3D) that supports quantitative modelling of mechanical systems, these models are not well suited to FMEA production. The produced FMEA must be in a natural language format, if additional re-training and effort

Component	Function	Failure Mode	Failure Effect	Remarks
Hydraulic Cylinder	Produce force to push inner drums	Leaking, cracked, or loose	No force or not enough force to push the inner drum. This reduces friction.	Leaked oil makes the friction surface greasy and reduces friction.

Figure 1: FMEA Sample

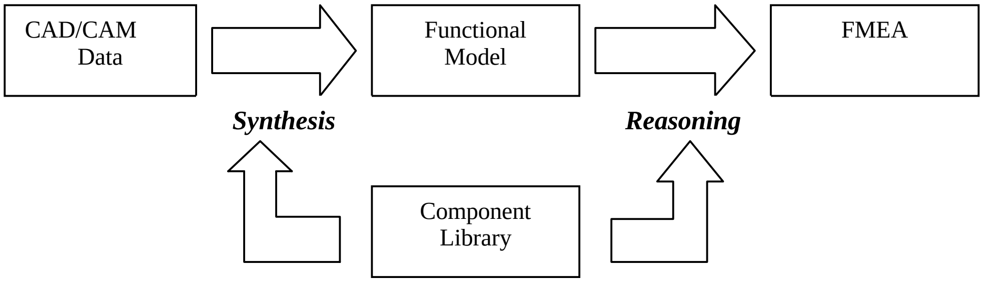

Figure 2: GENMech System Overview

is to be avoided. It is natural to describe a mechanical system in terms of its functions. Indeed, the effects of a failure are typically described in terms of which functions of the system are not achieved.

Figure 1 shows a row from an FMEA for a car drum-brake. Each row in an FMEA considers a single failure mode for a single component. As shown in the table, the failure's effect is described in terms of functions that it does not achieve, i.e. no force or reduced force on the inner drum. Functional descriptions of components are very similar to the syntax used in an FMEA. This tight coupling is desirable because any model building effort required of the engineer then utilizes a similar vocabulary.

The GENMech System

GENMech's approach to automated FMEA production can be broken down into a number of distinct stages and applications. The overall structure of the system as a whole is shown in Figure 2.

The system is broken down into a number of tasks, and the applications required in order to complete or apply them:

- *Synthesis* – The CAD data must be mapped into a functional model that can be reasoned about in order to produce a FMEA.
- *Reasoning* – The functional model is reasoned about in order to produce an FMEA. Component models described in a component library are used.
- *Component Library* – Maintains and manages a global library of components and their associated state transition tables. At present this application is entirely dependent on operator input for component model building. The operator is required to define state transition tables describing the normal behaviour of the component, and an additional table for each failure mode the component has.
- *Model Builder* – Although the functional model is automatically synthesized, an engineer must be able to alter or augment the generated model before it is used to produce an FMEA.

GENMech works by extracting CAD/CAM data from the associated application, and translated from a spatial/geometric form, in the functional mode. Then, using the component library to provide data about the nature of the components in the system, the functional model is automatically constructed. The resultant model is reasoned about in order to identify the correct behaviour of the device. Finally, component failure modes are introduced, and the effect on the functionality of the rest of the device is recorded in an FMEA.

The prototype system is being constructed around Autodesk's AutoCAD release 14 and Mechanical Desktop 2.0. Mechanical Desktop allows the components designed in AutoCAD to be integrated into a complete assembly. The current focus of the project is the synthesis of the CAD data into the chosen format for the functional model, using the assembly data built using Mechanical Desktop.

The Functional Model

GENMech uses a functional model that has been tailored to the requirements of FMEA production. The device is represented as a network of components. Each link between components is called a port. Ports are bi-directional, conveying information about the linked components' behaviour between them. A more detailed description of exactly what is passed through ports is given later in this section. A simple device network is shown in Figure 3.

Each component in the system has a state transition table. The transitions between states are annotated with information about the function that the component is realising when the transition is made.

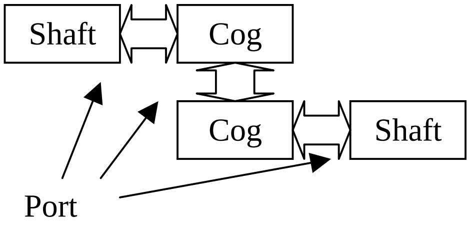

Figure 3: A Network Representing a Gear-mesh

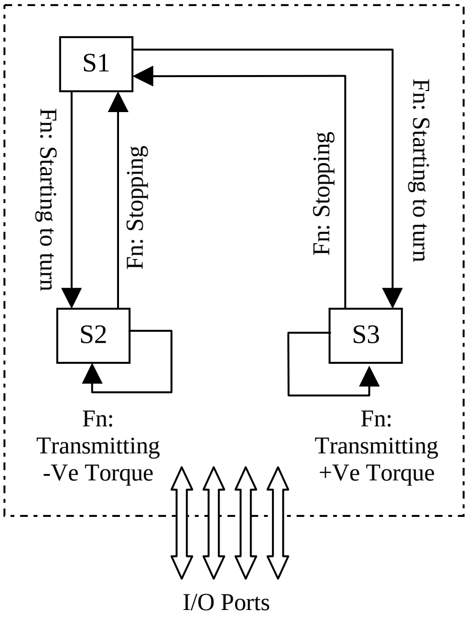

Figure 4: Transition Diagram for a Simple Shaft

In addition, any number of conditions may be specified for any state transition. These conditions may be on either the internal state of the component, or on external influences from other components within the system. Once a function is realised by a component, it is transmitted to all connected components through a network of ports. An example state transition diagram is shown in Figure 4.

The system is simulated by the propagation of the realised function through the system, as it becomes an input to some subsequent component's state transition table. FMEA is concerned primarily with how the system responds to the failure of one or more components. Consequently it is necessary to incorporate a description of a component's behaviour when it is failing. In fact, a number of these descriptions must be provided so that a range of failure conditions (Hudson 1994) are considered.

The state transition table for normal operation of the simple shaft is shown in Figure 5. The shaft responds to a torque by transferring that torque to a connected component. The components supplying, or being supplied, functions are denoted by the number in the On column. In this case, component 20.1.0 is the mechanical power supply sub-assembly for an adjustable car seat. If the shaft were to jam (possibly due to worn bearings), the component could not enter state S2 or S3. To model this failure the designer can define a new state-transition table for the failure mode (see figure 6). The "Transmit +Ve Torque" function is no longer realised, and the supplied torque will not now be propagated to connected components. By substituting the state transition table for normal behaviour (figure 5) with a named state transition table describing a particular failure mode (figure 6), the effects of the particular failure on the system can be simulated.

Automated Model Generation

In order to automate effectively the FMEA process, it is necessary to minimize both the overall effort required by the new practice, and any learning necessary to produce the required data. It is unlikely that an engineer will have any experience building functional models. Therefore, GENMech has focused on the automation of model building. This involves taking CAD/CAM data already produced for the device, and mapping it to the functional model.

State	Described As	On	Go To	Function Realised
S1	Inactive	(20.1.0 – Remain Inactive)	S1	Remaining Inactive
		(20.1.0 – Supplying +Ve Torque)	S2	Accelerating
		(20.1.0 – Supplying -Ve Torque)	S3	Accelerating
S2	Turning +Ve	(20.1.0 – Supplying +Ve Torque)	S2	Transmitting +Ve Torque
		(20.1.0 – Decelerating)	S1	Decelerating
S3	Turning -Ve	(20.1.0 – Supplying -Ve Torque)	S3	Transmitting –Ve Torque
		(20.1.0 – Decelerating)	S1	Decelerating

Figure 5: A Sample State-transition Table

State	Described As	On	Go To	Function Realised
S1	Inactive	(20.1.0 – Remain Inactive)	S1	Remaining Inactive
		(20.1.0 – Supplying +Ve Torque)	S2	-
		(20.1.0 – Supplying –Ve Torque)	S2	-
S2	Jammed	(20.1.0 – Supplying +Ve Torque)	S2	-
		(20.1.0 – Decelerating)	S1	Decelerating

Figure 6: Substitute Description of Failure Mode Behaviour

Connections between devices are represented through spatial location and one or more constraints placed on, or between, components. The constraints are between the geometric properties of the component, such as their surfaces or edges. Most usefully these include (the numbers in brackets are AutoCAD's identifiers for the constraint, as shown in italics in figure 7):

- Fixed distance between two planes (3),
- Fixed distance between two lines (4)
- Fixed distance between line and plane (117)
- Plane in Plane (118)

Synthesis is performed based on the assumption that constraints represent connections between the components that the constraints are between. The synthesis application integrates with Mechanical Desktop and extracts a list of components, and the constraints on each, from the current assembly. The initial data captured contains a significant amount of redundancy. A single constraint between two components is represented as two complementary constraints. GENMech parses this data and identifies those complementary constraints and produces a single constraint between the two components. Figure 7 shows an extract from the first parsing of the data for a brake's cylinder sub-assembly. For each component in the assembly a list is produced of the other components it is connected to (and the geometric constraints between them). Finally, each component's local coordinate space is provided as a homogenous matrix. This may be necessary to map the calculated degrees of freedom of each component into the co-ordinate space of other components.

The constraints (shown in italics) are used to establish the links in the graph that forms the basis of the functional model. The geometric constraints themselves are not directly useful for further model building, and must first be resolved into mechanical constraints. Mechanical constraints have been represented in terms of the degrees of freedom of each individual component. Knowing the degrees of freedom for each component allows the nature of any transferals of force to be determined. For example, if a component has a rotational degree of freedom it is able to transmit torque. If it has a translational degree of freedom it is able to transmit linear force.

```
CYLINDER_1
  1 PISTON1_2 CYLINDER_1 118 0
  1 PISTON1_1 CYLINDER_1 118 0
    1.00   0.00   0.00   176.16
    0.00   1.00   0.00   191.92
    0.00   0.00   1.00    14.00
    0.00   0.00   0.00     1.00
```

Figure 7: Extracted Assembly Data after Initial Parsing

By reasoning with this information and the state transition tables stored in the component library for the component, further functional links between component can be added. In addition, redundant links can be removed. If the component description only defines the component's behaviour being dependant on supplied linear force, and reasoning with the CAD data has resulted in the identification of only one rotational degree of freedom then any connection between that component and a torque supplier is removed.

The algorithm for producing the final state transition tables for the component in the device can be broken into four principle phases. At each phase the engineer can be presented with a "check-list" of the generated information to confirm or correct the results of the relevant phase.

- **Input Component Identification**: The engineer is requested indicate those components such as motors that represent inputs into the system, and to confirm the nature of their input (for example, torque).
- **Active Component Output Identification**: Although the functional model of each component may include a wide range of possible outputs, not all of these may be realisable in a particular assembly. By analysing the outputs of connected components in the system, outputs that will never be realised can be eliminated.
- **Identify Supportive Inputs to Components**: Where a component is receiving an input from another component that does not effect a possible state, the input can be considered to be important in all states.
- **Produce State Transition Tables**: Once the range of inputs and outputs of each component have been identified, a state transition table is produced for each component. The transitions are labeled with the low-level functions they realize. The connections between the components are established.

Limitations

Qualitative models have inherent limitations when modelling devices where component behaviour is dependent on the values of parameters (such as the amount centrifugal force). To overcome this it is necessary for an engineer to construct a separate (possibly complex) state-transition table for that part of the mechanism. That part of a device can then be treated as a single component. Consequently, the failures of the individual components that are grouped cannot be included in the produced FMEA. As already discussed, building state transition tables is a non-trivial task that engineers are not familiar with.

The types of constraint that the CAD system can represent limits the automatic model generation procedure. For example, AutoCAD does not allow the specification of complex movement paths for components. Consequently, some locking/unlocking movements within a device's cannot be modelled, as only a "snap-shot" of the device is represented. In addition, the system does not currently support dynamic modifications of the connections between components that would be required to represent these device behaviours. However, this is a limitation of the current implementation and not of the techniques in general.

Finally, GENMech provides no temporal grounding, meaning that functions dependent on precise timings cannot be modelled. This is important as the high-level function of a device may incorporate restrictions on the time available for the function to be realised.

Conclusions

It can be seen that traditional quantitative methods for modelling mechanical systems are inappropriate for automated mechanical FMEA production. Functional modelling capitalizes on existing practices of describing components at the functional level when producing FMEAs.

A functional model schema has been defined that is tailored to the reasoning requirement imposed by the automation FMEA production for mechanical systems.

Initial work has been directed to reducing the amount of engineer intervention required when converting pre-existing information resources such as CAD/CAM data into the functional model required for FMEA production. Kinematic systems have been the focus of work to date as the majority of the information stored in CAD/CAM produced files is associated with this domain. By using information about constraints between the geometric properties of components in AutoDesk's Mechanical Desktop assemblies and functional descriptions of component behaviour stored in a library, it is possible to produce a constrained graph of the components in an assembly, which then requires a simple translation into the functional model.

Acknowledgements

Funding: EPSRC Engineering in Manufacturing Grant (GR/L 64133). We would also like to thank the following industrial collaborators: Ford Motors, Jaguar Cars Limited, and Pilkington Optronics.

References

Price, C. J., Pugh, D. R., Wilson, M. S., and Snooke, N., 1995. The Flame System: Automating Electrical Failure Modes and Effects Analysis (FMEA). *Proceedings of the Annual Reliability and Maintainability Symposium*, IEEE, pp. 90-95.

Pugh, D. R., and Snooke, N., 1996. Dynamic Analysis for Qualitative Circuits. *Proceedings of the Annual Reliability and Maintainability Symposium,* IEEE, pp. 37-42

MIL-STD-1629A, 1980. *Procedures for Performing a Failure Mode, Effects and Criticality Analysis*. Department of Defense, Washington DC.

BS5760, 1991. *Reliability of systems, equipment and components part 5. Guide to failure modes, effects and criticality analysis (FMEA and FMECA)*. British Standards Organization.

Ford Motors, 1995. *Failure Mode & Effects Analysis Handbook*. Ford Automotive Safety and Engineering, Automotive Safety and Engineering Standards Office.

Chrysler, Ford, and General Motors, 1995. *Potential Failure Mode and Effects Analysis (FMEA) Reference Manual*. Chrysler Corporation, Ford Motors, General Motors Corporation.

Onodera, K., 1997. Effective Techniques of FMEA at Each Life-Cycle Stage. *Proceedings Annual Reliability and Maintainability Symposium.* IEEE, pp. 50 –56

Bowles, J. B., and Bonnell, R. D., 1994. Failure Mode, Effects, and Criticality Analysis. *Annual Reliability and Maintainability Symposium 1994.* IEEE, pp. 74-79

Dale, B. G., and Shaw, P., 1990. Failure Mode and Effects Analysis In The U.K. Motor Industry: A state-of-the-art Study", *Quality and Reliability Engineering International*. Vol.6, pp. 179-188.

Hudson, J. E., 1994. Understanding Part Failure Mechanisms. *Annual Reliability and Maintainability Symposium 1994.* IEEE, pp. 99-106