Component Diversity and Minimizing Multiple Failures

PONI Summer Conference
June 24, 2015

Andrew Mastin
Ariane 5 Rocket 501 (in 1996)

- Exploded within the first minute of flight.

- Software bug: guidance computer generated a number too large to process, so it shut down and handed control to the other computer.

- Other computer was a replica of the first, so it made the same decision and shut down.

This is an instance of a **common cause failure**, defined as a *Failure, that is a direct result of a shared cause, in which two or more separate channels in a multiple channel system are in fault state simultaneously, leading to system fault.* (Rausand 2014)

(We should not use the term common mode failure; refers to components being in the same functional mode.)

Related notion is **multiple failure with a shared cause**, defined as

*Failure, that is a direct result of a shared cause, in which two or more items are in fault state simultaneously.* (Rausand 2014)

(In a MFSC, the system still functions.)

---

This motivates the use of **diversity**: instead of making the second computer a replica for Ariane 5, what if we had forced it to be different?

- For example, if we had required it to have software written by a different group of people, it’s possible that the bug would not have been present in both.

Diversity guards against **coupling factors** in a system:

- Same design, hardware, software, installation/maintenance/operation staff, procedures, environment, location.
- Coupling factors reduce independence of failures.

Stockpile diversity

- What do people mean when they talk about stockpile diversity? Here are two proposed definitions.

- **Strategic diversity**: Ability to strike from air, land, water (i.e. the triad); different weapon types (B61-3, B61-7); varying yields.

- **Component diversity**: The use of different components to minimize the impact that a faulty component has on the stockpile.

- This talk is about the latter – the concern of multiple failures with a shared cause in the stockpile.

For strategic diversity, see, e.g., Lafleur, J. Triads, Diads, and Interoperability: A Structured Approach to Tracing the Implications of Diversity on Deterrent Force Reliability. PONI, 2013.
Models for diversity

- What about the beta-factor model?
  - Widely used method for common cause failure analysis.
  - Modify reliability block diagram by inserting a common cause failure “component”.
  - Beta is fraction of failures due to common cause.
  - But, determining beta without sufficient data is tricky.

- Littlewood/Miller/Hughes models
  - Interplay of environments and components
  - Having observed the failure of one of the components makes us more confident that this is a stressful environment, i.e. that the probability of failure is greater for every component, than would otherwise be the case.
  - Environments/components can also be thought of as inputs/programs.
  - Proofs (!) for the benefits of diversity.

- This presentation: a new model based on a minmax criterion.

Example problem

- Building a fleet of nine rockets.
- Analogous to a collection of, e.g., B61s.
- Each rocket requires nose, body, tail.
- We have multiple brands of each part.
  - Three nose brands
  - Three body brands
  - Three tail brands
- (Brands can be interpreted as manufacturing date, installation team, software version.)
- As far as we can tell, part brands are equally reliable, but we haven’t tested them extensively.
Example problem

- One option is

- A problem with black noses affects 77% of fleet.
- A problem with blue bodies affects 88% of fleet.
- A problem with purple tails affects 100% of fleet.
- Interpretation: not diverse
Example problem

- Another option is

- A problem with any one brand now only affects 33% of fleet.
- *Interpretation*: more diverse
- This configuration **minimizes the maximum occurrence of each brand**.
- Can we do better?
Consider versus

Latter fleet minimizes *pairs* of brand occurrences, and is rightly considered more diverse.

For example, if orange bodies and gray tails have poor interaction, only 22% of fleet is affected in latter case.
Implementation

- For larger, more realistic problems, can formulate a mixed integer program.

- Match parts \( p \) to system slots \( s \) to minimize the occurrence of each brand \( b \).

\[
\begin{align*}
\min & \quad z \\
\text{s.t.} & \quad \sum_{p : (p, s) \in E} x_{ps} = 1, \quad \forall s \in S, \\
& \quad \sum_{s : (p, s) \in E} x_{ps} = 1, \quad \forall p \in P, \\
& \quad \sum_{s \in S} \sum_{p \in P : B(p) = b} x_{ps} \leq z, \quad \forall b \in B, \\
& \quad x_{ps} \in \{0, 1\}, \quad \forall p \in P, \forall s \in S.
\end{align*}
\]

- Possible extensions: minimizing pairs of brands; multiple brands for each part; multiobjective approach for second, third most frequently occurring brands.