Chapter 11

Solution Design

PAUL D. WEST, Ph.D.

> We can't solve problems by using the same kind of thinking we used to create them.
>
> —Albert Einstein

> You can observe a lot just by watching.
>
> —Yogi Berra

11.1 INTRODUCTION TO SOLUTION DESIGN

The solution to any well-formed problem exists, needing only to be found. Once the decision maker's needs, wants, and desires have been identified, the solution design process develops a pool of candidates in which the "best" solution can be expected to be found. The pool is refined as it grows, checked constantly against the problem definition and measured against stakeholder objectives until the best solution emerges. The process is fluid and may iterate often across the spectrum of define, design, and decide loops, as shown in Figure 11.1.

The resulting *solution* is a process for solving a problem [1]. Given the stakeholders' needs, it is a process of determining candidate *solutions* to present to the decision maker. These present the decision maker with a choice limited to one of two or more possibilities [2]. Each carries with it some degree of uncertainty and risk, so the solution must include techniques to reduce these to make the decision maker's job easier.

Decision Making in Systems Engineering and Management, Second Edition
Edited by Gregory S. Parnell, Patrick J. Driscoll, Dale L. Henderson

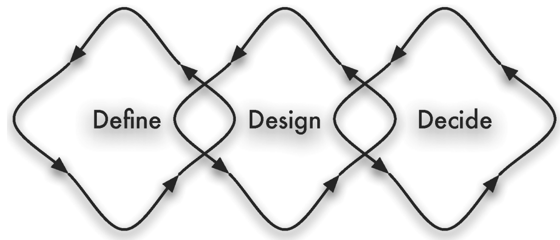

Figure 11.1 Define, design, decide loops.

Solution design is a deliberate process for composing a set of feasible alternatives for consideration by a decision maker.

In this chapter we explore techniques for casting a sufficiently broad net to confidently capture the best solution in a raw form, reduce the catch to most likely candidates, and present a compelling argument for how each candidate solution meets the needs of the decision maker.

Identifying *who* contributes to the solution design is a critical task of the systems engineer, as was seen in Chapter 7. Figure 11.2 shows a part of the Solution Design Concept Map that illustrates the role of the solution designer. The lead systems engineer must identify early on the best mix of skills needed to accomplish the tasks shown here (see Section 7.4).

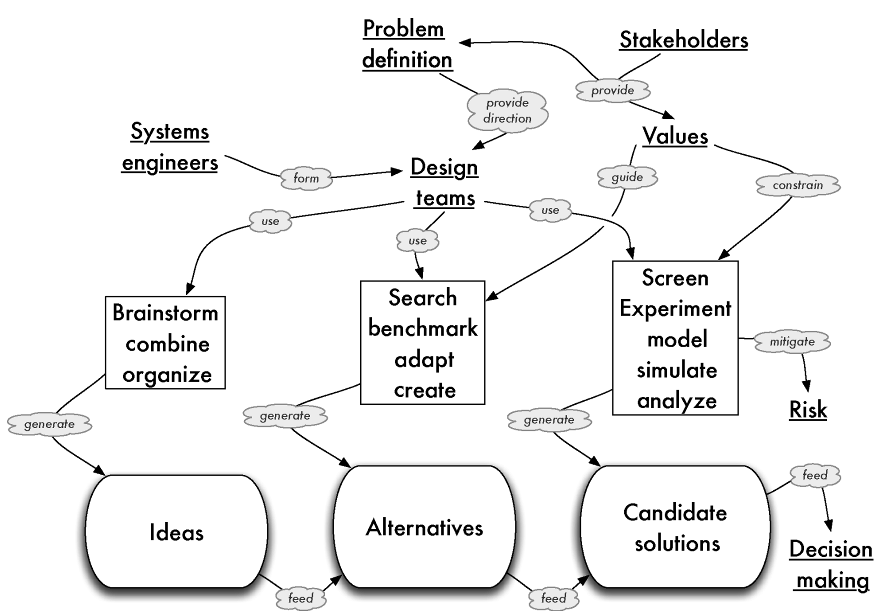

Figure 11.2 Concept map for Chapter 11.

11.2 SURVEY OF IDEA GENERATION TECHNIQUES

Albert Einstein offers solution designers good advice in the opening quote of this chapter. Innovative thinking is essential for successful solution design, but should not be confused with haphazard design, which more often leads to observations like Thomas Edison's that "I have not failed. I've just found 10,000 ways that won't work."

There is no single right way to generate alternatives within a solution design. Much of the art of solution design is in knowing which tools to draw from the toolkit for a specific problem. However, all methods follow the basic model shown in Figure 11.3. The circular arrows indicate feedback and iteration loops, as necessary.

11.2.1 Brainstorming

Brainstorming capitalizes on the idea that a panel of creative thinkers can create a pool of ideas that will include the nucleus of a solution. It adopts the adage that two heads are better than one, or in what early operations researchers proclaimed as the *n-heads rule* [3]: *n heads are better than n heads* −1. The sole product of a brainstorming session is a list of ideas. No attempt at analysis is made except to reduce the total list of ideas to general categories by using such techniques as affinity diagramming, described later in this chapter.

The brainstorming concept was developed by advertising executive Alex Osborn in 1941, who observed that conventional business meetings were conducted in a way that stifled the free and creative voicing of ideas. The term *brainstorming* evolved from his original term to *think up* for his new process [4]. In a series of later writings, he described brainstorming as "a conference technique by which a group attempts to find a solution for a specific problem by amassing all the ideas spontaneously by its members." Osborn became a prolific writer on the Creative Problem Solving process in works including *Your Creative Power: How to Use Imagination* [5], in which he introduced the brainstorming term,*Wake Up Your Mind: 101 Ways to Develop Creativeness* [6], and *Applied Imagination: Principles and Procedures of Creative Thinking* [7].

Osborn developed the following four basic rules for guiding brainstorming sessions:

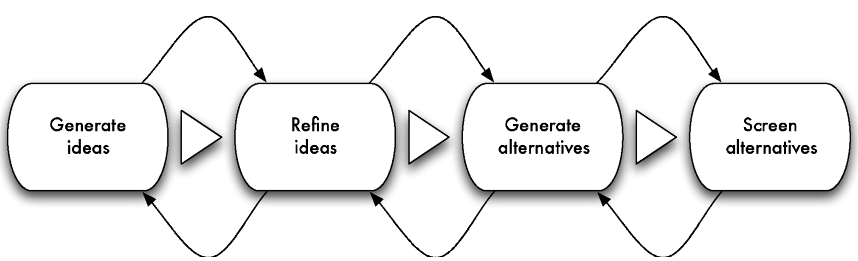

Figure 11.3 The ideation process.

No Criticism of Ideas. All judgment on ideas is deferred and participants feel free to voice all ideas without fear of judgment of the idea or themselves.

Encourage Wild and Exaggerated Ideas. Wild and crazy ideas often contain nuggets of unique insight that may be built upon. Wild ideas will be refined and filtered during later concept evaluation.

Seek Large Quantities of Ideas. Brainstorming sessions are fast-paced and spontaneous. Ideas may not only be useful in their own right, but also serve as catalysts for other ideas.

Build on Each Other's Ideas. Ideas offered from one person's perspective often trigger wholly new ideas from others.

Brainstorming sessions generally take one or more of three forms: *structured, unstructured* (free form), or *silent*. The main difference between the first two forms involves the level of control by the facilitator, who may combine any of the three to meet problem needs.

A typical brainstorming session has six steps and lasts for about an hour. Care is needed in selecting the right mix of participants (see Section 10.2). Groups generally consist of 5–10 participants known for their creativity and initiative and represent a broad range of experience with the problem. A structured session may take the following form [8]:

1. *Problem Definition.* The problem statement should be simple, often a single sentence, and identified in advance. The problem should be more specific than general, and complex problems should be split into functional subproblems that can be addressed in multiple sessions. Osborn recommends that a one-page background memo should accompany the invitation to participants, along with some sample ideas.
2. *Organization.* Seating participants around a round table is ideal for promoting interaction and for enabling the fast pace of the actual session. Distracters such as food and books should be avoided.
3. *Introduction.* Facilitators introduce the problem statement and context. The process is explained and any questions on the problem or process are answered.
4. *Warm-up.* A five-minute warm-up with sample problems may help jump-start the session to a high-tempo level. Sample questions may include "What if restaurants had no food?" and "What if pens wrote by themselves?" [9].
5. *Brainstorming.* Restate the problem and begin. Limit the session to 20–30 min. Pass a My Turn object around the table, with a 10-second time limit for the holder to offer an idea before passing it to the next person. The idea is recorded on a 3-by-5 index card or Post-It® -like note by the holder, a recorder, or facilitator. Ideas should not exceed about 10 words. If the stated idea spawns one from another member of the group, the turn goes

out of sequence to the second person; if the holder cannot think of an idea, the My Turn object moves to the next person in sequence. Trends toward an exhaustion of ideas may be broken by radical ideas interjected by the facilitator. The session ends when time expires or no more ideas are offered. When the session is complete, idea cards are grouped into categories using affinity diagramming and central ideas are identified.

6. *Wrap-up.* Conduct a brief after-action review to identify strengths and weaknesses from the session. Use these comments to describe the environment in which ideas were made during the solution design process and how to improve future sessions.

Unstructured sessions use a more free-form approach, abandoning the My Turn object. Participants contribute ideas as they come to mind.

Brainstorming critics soon emerged. By the 1950s, detractors declared that the effectiveness of brainstorming had been overstated and the tool simply did not deliver what it promised [10]. Much of the criticism, however, was based on poorly planned or conducted sessions, or focused not on brainstorming itself, but on the use of individuals or groups in brainstorming, as was the case of the report that became known as the Yale Study done in 1958 [11].

Critics generally agree on seven main disadvantages of Osborn's process, summarized by Sage and Armstrong [12] below. Several of these factors contribute to what Yale social psychologist Irving Janis called *groupthink*, when group members' striving for unanimity overrides their motivation to realistically appraise alternative courses of action [13].

Misinformation. There is no guarantee that incorrect information will not influence the process.

Social Pressure. A group may pressure individuals to agree with the majority, whether they hold the majority view or not.

Vocal Majority. A strong, loud, vocal majority may overwhelm the group with the number of comments.

Agreement Bias. A goal of achieving consensus may outweigh one for reaching a well thought-out conclusion.

Dominant Individual. Active or loud participation, a persuasive personality, or extreme persistence may result in one person dominating the session.

Hidden Agendas. Personal interests may lead participants to use the session to gain support for their cause rather than an objective result.

Premature Solution Focus. The entire group may possess a common bias for a particular solution.

The ensuing decades witnessed numerous variations on Osborn's technique, each seeking to overcome one or more of the identified flaws. Some of these are described below.

11.2.2 Brainwriting

Brainwriting is a form of silent brainstorming that attempts to eliminate the influence of dominant individuals and vocal majorities. The two most popular variations are written forms of the structured and unstructured brainstorming techniques.

- In structured brainwriting, ideas related to the given problem are written on a paper that is passed from member to member in much the same way as the My Turn object. New ideas may be added or built from others.
- Unstructured brainwriting does not follow a sequential path. Ideas are written on note cards and collected in a central location.

11.2.3 Affinity Diagramming

This technique was introduced in Chapter 10 and categorizes the ideas into groups that can then be rated or prioritized. This is also a silent process, with participants creating logical groups of ideas with natural affinity. Duplicates are first removed from the collection, and notes are stuck to a wall or large board so that all participants can see them. Without speaking, participants move notes into unnamed groups using logical associations until everyone is satisfied with the organization. This often results in notes being moved to different clusters many times until the groups reach equilibrium. The groups are then labeled with descriptive headers, and often form the basis of alternatives, with the header as the alternative name and the member ideas as attributes.

11.2.4 Delphi

Delphi methods seek to minimize the biasing effects of dominant individuals, irrelevant or misleading information, and group pressure to conform. Delphi introduces three unique variations on the brainstorming concept [14].

1. Anonymous response—where opinions are gathered through formal questionnaires.
2. Iteration and controlled feedback—processing information between rounds.
3. Statistical group response—with group opinion defined as an aggregate of individual opinions on the final round.

Delphi methods also differ from traditional brainstorming techniques in their stricter reliance on subject matter experts. Development of the tool began in the mid-1940s shortly after General Henry H. "Hap" Arnold, Commanding General of the Army Air Forces, pushed for the creation of a Research and Development (RAND) project within the Douglas Aircraft Company of Santa Monica, California. Arnold, a 1907 graduate of West Point, who was taught to fly in 1911 by Orville Wright and rose to lead the Army Air Forces during World War II, wanted expert forecasts of technologies reaching decades into the future.

The Air Force's Project RAND, which became the RAND Corporation in 1948, developed a statistical treatment of individual opinions [15], and researchers evolved the process that became known as Delphi. By 1953, Dalkey and Helmer had introduced the idea of controlled feedback [14]. Their project was designed to estimate, from a Soviet strategic planner's view, the number of atomic bombs necessary to reduce United States' munition output by a given amount.

Sessions typically follow 10 basic steps, described by Fowles [16]:

1. Form the team to undertake and monitor the session.
2. Select one or more panels of experts to participate.
3. Develop the first round questionnaire.
4. Test the questionnaire for proper wording.
5. Submit the first questionnaire to the panelists.
6. Analyze the first round responses.
7. Prepare the second round questionnaire.
8. Submit the second questionnaire.
9. Analyze the second round responses. Repeat steps 7 through 9 as necessary.
10. Prepare the report on the findings.

The 10 steps highlight another benefit of the approach in that panelists most likely are not assembled in a central location. However, this also suggests a danger inherent in this process. A Delphi session in which there is insufficient buy-in by participants or is poorly conceived or executed may suffer from participant frustration or lack of focus. Other critics of this technique note that the aggregate opinion of experts may be construed as fact rather than opinion, facilitators may influence responses by their selection and filtering of questions, and future forecasts may not account for interactions of other, possibly yet unknown, factors.

Selecting the right questions is key for Delphi success. The sidebar, *In Case of Nuclear War*..., extracted from Dalkey and Helmer's reflections on applications of the Delphi method [3], shows questions from the first two of five rounds of their 1953 study. Panelists included four economists, a physical vulnerability expert, a systems analyst (precursor of the systems engineer), and an electronics engineer. The paper was published 10 years after the event for security reasons, and even then parts remained classified and were omitted.

IN CASE OF NUCLEAR WAR—A DELPHI STUDY

Questionnaire 1: This is part of a continuing study to arrive at improved methods of making use of the opinions of experts regarding uncertain events. The particular problem to be studied in this experiment is concerned with the effects of strategic bombing on industrial parts in the United States.

Please do not discuss this study with others while this experiment is in progress, especially not with the other subject experts. You are at liberty, though, to consult whatever data you feel might help you in forming an opinion.

The problem with which we will be concerned is the following:

Let us assume that a war between the United States and the Soviet Union breaks out on 1 July 1953. Assume also that the rate of our total military production (defined as munitions output plus investment) at that time is 100 billion dollars and that, on the assumption of no damage to our industry, under mobilization it would rise to 150 billion dollars by 1 July 1954 and to 200 billion dollars by 1 July 1955, resulting in a cumulative production over that two-year period to 300 billion dollars. Now assume further that the enemy, during the first month of the war (and only during that period), carries out a strategic A-bombing campaign against U.S. industrial targets, employing 20-KT bombs. Within each industry selected by the enemy for bombardment, assume that the bombs delivered *on target* succeed in hitting always the most important targets in that industry. What is the least number of bombs that will have to be delivered on target for which you would estimate the chances to be even that the cumulative munitions output (exclusive of investment) during the two-year period under consideration would be held to no more than one quarter of what it otherwise would have been?

Questionnaire 2: As the result of the first round of interviews, it appears that the problem for which we are trying with your help to arrive at an estimated answer breaks down in the following manner.

There seem to be four major items to be taken into consideration, namely:

a. The vulnerability of various potential target systems,
b. The recuperability of various industries and combinations of industries,
c. The expected initial stockpiles and inventories, and
d. Complementarities among industries.

Taking all these into account, we have to:

1. Determine the optimal target system for reducing munitions output to one fourth, and
2. Estimate for this target system the minimum number of bombs on target required to create 50 percent confidence of accomplishing that aim.

We would like to establish the background material consisting of A, B, C, D more firmly. With regard to A and B, the interviews have suggested the following tentative breakdown of possibly relevant factors: two lists of factors were given, related to vulnerability and recuperability, respectively.

11.2.5 Groupware

This is increasingly popular for conducting brainstorming and other collaboration using networked computers. Applications such as GroupSystems [37] mirror the brainstorming process and emphasize anonymity. Other groupware tools include e-mail, newsgroups and mailing lists, workflow systems, group calendars, server-based shared documents, shared whiteboards, teleconferencing and video teleconferencing, chat systems, and multiplayer games.

11.2.6 Lateral and Parallel Thinking and Six Thinking Hats

These are unique creativity techniques for thinking about problems differently. "You cannot dig a hole in a different place by digging the same hole deeper," says concept developer Edward DeBono [17]. He insists that basic assumptions be questioned in lateral thinking. For example, he notes that chess is a game played with a given set of pieces, but challenges the rationale for those pieces. Lateral thinking, he says, is concerned not with playing with the existing pieces, but with seeking to change those very pieces. Parallel thinking [18], he says, focuses on laying out arguments along parallel lines instead of adversarial. Techniques for accomplishing this are described in his Six Thinking Hats approach. In these sessions, participants display or wear one of six colored hats that determine the person's role in the session.

- *White hat*: neutral, focused on ensuring that the right information is available.
- *Red hat*: intuitive, applies feelings, hunches, emotion, and intuition.
- *Black hat*: cautious, provides critical judgment and says why things cannot be done.
- *Yellow hat*: optimist, finds ways that things can be done, looks for benefits.
- *Green hat*: creative, encourages creative effort, new ideas, and alternative solutions.
- *Blue hat*: strategic, organizes and controls the session.

11.2.7 Morphology

Morphology explores combinations of components through the study of forms and structure. This approach, called morphological analysis, was developed by astrophysicist and aerospace scientist Fritz Zwicky in the mid-1960s for studying multidimensional, nonquantifiable complex problems [19]. Zwicky, a scientist at the California Institute of Technology (CalTech), wanted to find a way to investigate the total set of possible relationships between system components, then to reduce the total set to a feasible solution space. He summarized the process in five steps:

1. Concisely formulate the problem to be solved.
2. Localize all parameters that might be important for the solution.
3. Construct a multidimensional matrix containing all possible solutions.

4. Assess all solutions against the purposes to be achieved.
5. Select suitable solutions for application or iterative morphological study.

Zwicky proposed finding “complete, systematic field coverage” by constructing a matrix with all possible system attributes, then producing a feasible set by eliminating combined attributes that are inconsistent with the stated purposes of the system. These inconsistencies could be:

- Logically inconsistent, such as “achieve Earth orbit in an underwater environment.”
- Empirically inconsistent, based on improbability or implausibility, such as “build an aircraft carrier using personal savings.”
- Normatively inconsistent, based on moral, ethical, or political grounds.

Solution designers should focus initially on the first two inconsistencies, Zwicky says, but they cannot ignore the third, which includes legal and regulatory constraints. The U.S. Department of Defense (DOD), for example, publishes DOD Directive 5000.1 and DOD Instruction 5000.2, specifying processes for the Defense Acquisition System.

Although Zwicky’s model is actually an n-dimensional matrix, his original illustration, shown in Figure 11.4, makes clear why it has become known as Zwicky’s Morphological Box or simply Zwicky’s Box. This figure shows a $5 \times 5 \times 3$ matrix containing 75 cells, with each cell containing one possible system configuration. Continuing the rocket design problem from Chapter 10, solution designers might develop the 4×4 Zwicky Box shown in Figure 11.4 to conduct a morphological analysis. Columns are labeled with parameter names and rows with variable names. The matrix is filled in with possible values.

Figure 11.5 shows an alternative generation table for the rocket problem. The systems engineers have decided four system design decisions: the number of fins,

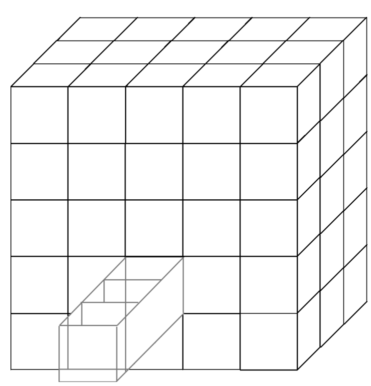

Figure 11.4 Zwicky’s morphological box [19].

Solution design parameters			
Fins	Thrust	Seeker	Guidance
2	1000	Forward looking infared	Inertial
3	1667	Laser	Global positioning system
4	2334	Audio	Wire
5	3000	None	Optical

morphological box

	Solution design parameters			
Strategy	Fins	Thrust	Seeker	Guidance
Global Lightening	5	2334	Laser	Global positioning system
Hot Wired	4	3000	None	Wire
Sight and Sound	2	3000	Audio	Optical
Slow poke	3	1000	Forward looking infared	Inertial
Star Cluster	3	1667	Forward looking infared	Global positioning system

generated alternatives

Figure 11.5 From morphological box to alternatives.

the thrust, the seeker, and the type of guidance. On the basis of the number of options in each column (5, 4, 4, and 4), 320 alternatives can be developed. However, all of these may not be feasible. Systems engineers select system alternatives that span the design space. Five designs are shown in Figure 11.5.

11.2.8 Ends–Means Chains

Thinking about how means support goals challenges solution designers to think creatively by presenting a higher-level objective and seeking different means for achieving it. This is repeated for means—seeking means to achieve previously stated means—to generate lower-level ideas. In this process, described by Keeney as a means–ends objectives network [20], lower-level objectives answer the question of how higher level objectives can be achieved. This differs from a fundamental objectives hierarchy (used in Chapter 10), he says, in which lower level objectives identify aspects of higher level objectives that are important. The process of finding new ways to achieve successive levels of objectives assists solution designers in structuring alternatives. Such a chain for the rocket design is shown in Figure 11.6. Arrows in this figure indicate the direction of influence.

11.2.9 Existing or New Options

Not all ideas must be completely new. Choices for using new or existing options are summarized by Athey [21] in Table 11.1, focusing on a problem to buy or upgrade a residence.

11.2.10 Other Ideation Techniques

Other techniques are cited in MacCrimmon and Wagner's presentation on Supporting Problem Formulation and Alternative Generation in Managerial Decision Making [22]. These include the following:

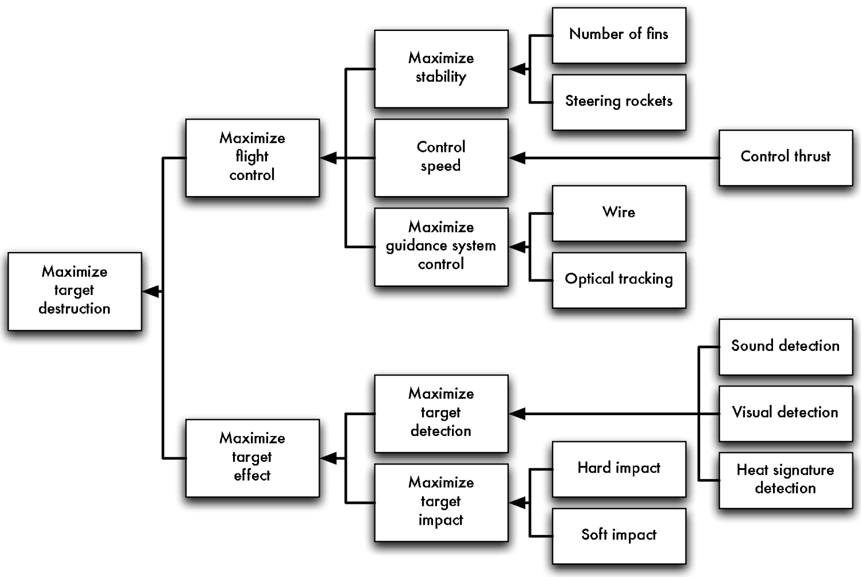

Figure 11.6 Rocket design objectives structure.

TABLE 11.1 Housing Upgrade Example

Method	Housing Example
Existing system	Present house—three bedrooms, new rug, new dishwasher.
Modified existing system	Remodel present house. Add-on a fourth bedroom. Repaint outside.
Prepackaged design	
• Off-the-shelf	• New housing tract.
• Learn from others	• Buy neighbor's four-bedroom house.
• Influential people suggest	• House mother-in-law suggests.
New system design	
• Idealized	• Design custom house.
• Parallel situation	• Build beehive house.
• Morphological	• Consider house, apartments, condo, mobile home.
• Cascading	• Select home by bedroom first, then family room, then backyard, etc.

Metaphoric connections to link disparate contexts by associating the problem with fragments of modern poems.

Loto modifications that alter existing ideas using modifiers such as "make it bigger, make it smaller."

Relational combinations and juxtaposition seek to create new ideas in ways similar to Zwicky's. The first technique applies sentences such as "[Process] by means of [Object 1] *relational word* [Object 2]." The latter technique uses up to three different problem elements at the same time.

11.3 TURNING IDEAS INTO ALTERNATIVES

Converting freewheeling ideas into reasonable alternatives occurs in the conceptual design phase of the systems life cycle. Here, ideas are compared with requirements and constraints and a feasible subset emerges for further analysis. Sage notes that "potential options are identified and then subjected to at least a preliminary evaluation to eliminate clearly unacceptable alternatives" [23]. Surviving alternatives, he adds, are then subjected to more detailed design efforts, and more complete architectures or specifications are obtained.

11.3.1 Alternative Generation Approaches

There is no clear line marking where idea generation ends and alternative screening begins. As seen earlier, this is built into many ideation techniques and the entire process normally cycles through several iterations. While the previous section focused on generating original ideas, practice shows that this represents only a fraction of the tactics used by organizations to uncover alternatives. A study of 376 strategic decisions showed that only about a quarter of the alternatives considered were developed using an original design tactic for generating alternatives [24]. In "A Taxonomy of Strategic Decisions and Tactics for Uncovering Alternatives," Nutt describes a study of 128 private, 83 public, and 165 private nonprofit organizations and the tactics they used to generate alternatives. Organizations ranged from NASA to Toyota and from Hertz–Penske Rental to AT&T. He distilled their tactics into six categories:

1. The *existing idea* tactic draws on a store of fully developed, existing solutions and follows a solution-seeking-a-problem approach with subordinates on the lookout for ways to put their ideas and visions to use.
2. *Benchmarking* tactics draw alternatives from practices of others who are outside of the organization rather than inside, as with the original idea tactic. Nutt draws a distinction between *benchmarking*, which adapts a single practice used by another organization; and,
3. *Integrated benchmarking* uses a collection of ideas from several outside sources.

4. *Search* tactics outsource the process through requests for proposals (RFPs) from vendors, consultants, and others who seem capable of helping.
5. *Cyclical searches* use an interactive approach, while *simple searches* use a one-time RFP.
6. The *design* tactic calls for custom-made alternatives that stress innovation to achieve competitive advantage. These normally require more time and other resource commitment.

Of the decisions studied, either only 36 could not be classified using Nutt's taxonomy, or decision makers switched methods during the process. The remaining 340 decisions are categorized in Table 11.2. Examining these techniques from alternative-focused thinking (AFT) versus value-focused thinking (VFT) perspectives (see Section 9.2) shows that *design* and *search* techniques favor a VFT approach, while *benchmarking, integrated benchmarking*, and *existing idea* techniques use AFT. The *cyclical search* technique could use either or both, depending on circumstances in each iteration.

11.3.2 Feasibility Screening

Feasibility screening techniques were introduced in the discusson of Zwicky's Box in Section 11.2.7. The objective is to reduce the number of alternatives that must be considered by refining, combining, or eliminating those that do not meet critical stakeholder requirements identified during the Problem Definition phase of the SDP (see Chapter 10).

The feasibility screening process can be thought of as a series of increasingly fine screens that filter out alternatives that fail to meet the stakeholders' needs, wants, and desires, as shown in Figure 11.7.

TABLE 11.2 Remaining 340 Decisions

Tactic	Number	Percent	How Alternatives Were Generated
Cyclical search	9	3	Multiple searches in which needs are redefined according to who is available
Integrated benchmarking	21	6	Amalgamation of ideas from several outside sources
Benchmarking	64	19	Adapt a practice used by another organization
Search	69	20	A single search cycle with a decision after RFP responses received
Design	82	24	Develop a custom solution
Existing idea	95	28	Validate and demonstrate benefits of a preexisting idea known in the organization

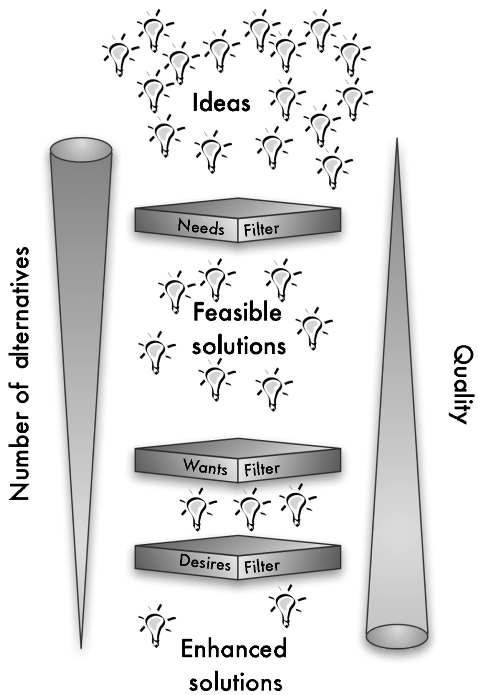

Figure 11.7 Feasibility screening.

- *Needs* are those essential criteria that must exist for the alternative to be considered. These are "must have" requirements identified by stakeholders.
- *Wants* are additional features or specifications that significantly enhance the alternative, but do not cause an alternative to be rejected if missing. These are "should have" requirements identified by stakeholders.
- *Desires* are features that provide a margin of excellence. These are "nice to have" requirements identified by stakeholders.

Solution designers may choose to initially screen at only the *needs* level, or may add screens for *wants* and *desires*.

In all cases, screening is evaluated on a go or no-go basis—an alternative either passes through the screen or it does not. Before rejecting a no-go alternative, however, designers should examine it to see if the offending feature can be modified or deleted so that the otherwise feasible alternative can make it through a repeat screening. The concept of alternative reduction also may be enhanced by combining alternatives or ideas, as described for affinity diagramming in Section 11.2.3. Revisiting the rocket design from Section 11.2.7, assume that the minimum standard for such a rocket is that it must have an effective range of 5 km, be launched from a standard four-wheel drive vehicle, and operate in terrain ranging from open fields to dense forest. A *feasibility screening matrix* with six of the 16 possible

TABLE 11.3 Partial Rocket Design Feasibility Screening Matrix

Alternative	Range Criterion (> 500 km)	Mobility Criterion (> med)	Terrain Criterion (All)	Overall Assessment
Global lightening	Go (800)	Go (high)	Go (All)	Go
Hot wired	Go (600)	Go (high)	Go (All)	Go
Sight and sound	Go (700)	Go (high)	No-Go (limited)	No-Go
Slow poke	Go (550)	Go (very high)	Go (All)	Go
Star cluster	Go (600)	Go (high)	Go (All)	Go
5-fin, 2234# thrust, none, wire	No-Go (400)	Go (high)	No-Go (limited)	No-Go

alternatives identified earlier versus criteria is shown in Table 11.3. Alternatives often take on descriptive names by this point, but in this case we list the parameters of *number of range criterion, mobility criterion, terrain criterion, and overall assessment*. This example shows that four of these alternatives met all of the minimum requirements. A single *no-go* for any criterion triggers an overall *no-go* assessment and the alternative is flagged for revision or rejection. For example, Sight and Sound fails the terrain criterion.

Feasibility screening matrices support alternative generation by quickly eliminating ideas that are clearly not feasible. Recall that two of the four guidelines for brainstorming described in Section 11.2.1 are “encourage wild and exaggerated ideas” and “seek large quantities of ideas.” This will naturally lead to many alternatives that are clearly not feasible. For example, the development of an aircraft requires that it must be able to fly. Any alternative that does not meet this fundamental criterion is eliminated during the feasibility screening process.

Screening criteria should avoid targeting feasible but less desirable alternatives. Feasibility screening sifts out alternatives based on non-negotiable criteria. For all others, finding the best trade-offs will lead the preferred solution. For example, a raw value of 90 for variable A may be acceptable when variable B is 30, but not when it is 50. Setting a No-go criterion for A at 90, then, would unduly restrict the solution space.

All feasible alternatives become solution candidates. Balancing tradeoffs to find the preferred solution is accomplished by enhancing and measuring solution candidates.

11.4 ANALYZING CANDIDATE SOLUTION COSTS

Cost constraints are vital considerations at all stages of a system’s life cycle. Chapter 5 introduced life cycle costing, cost estimating techniques, and the life cycle stages appropriate for the various techniques. In previous phases in this process, the systems engineer identified the components of the life cycle costs of the system under development. Now, the systems engineer must review these cost components and ensure that they completely cover the candidate solutions’ costs.

The systems engineer should know a great deal more about the candidate solutions at this point than at the beginning of the process when the cost components were developed. What has changed? Are there hidden or higher costs, such as development costs, manufacturing costs, or operational costs? Does the cost model consider all components of the newly developed candidate solutions? Once the systems engineer has ensured that the cost model is indeed complete, costs are computed for each candidate solution still under consideration. Although overall costs are needed for the next step, it is important to include all of the components in detail to sufficiently document the work and to answer any questions the decision maker may have about the analysis. It is also useful to note that Monte Carlo simulation can be conducted for the cost model just as it was for the value model.

11.5 IMPROVING CANDIDATE SOLUTIONS

The solution designer, armed with a list of feasible alternatives, must choose which of them to present to the decision maker for action. This choice is based on both qualitative and quantitative measures so that only the best alternatives make the cut to become final *candidate solutions* — the best the solution design team can offer. Key tools at this stage are the models and simulations described in Chapter 4 and a deliberate experimental design strategy for analyzing alternatives.

11.5.1 Modeling Alternatives

Models have been used both directly and indirectly throughout the problem-solving process. Early in the problem definition stage, key stakeholder values were assembled into a value model that identified five key aspects of value:

1. Why is the decision being made (the fundamental objective)?
2. What has a value (functions and objectives)?
3. Where are objectives achieved?
4. When can objectives be achieved?
5. How much value is attained (the value function)?

The value modeling process identified many of the requirements used during feasibility screening. It continues to be used at this stage to assess the value of each feasible alternative as candidate solutions emerge.

Feasible alternatives must also be evaluated using mathematical, physical, or event models developed for the system and assessed using measures of effectiveness or performance, as described in Chapter 4.

11.5.2 Simulating Alternatives

Complex systems with no closed-form solution or that have many interactive variables that change during operation often require simulations to generate effectiveness or performance values. As noted in Chapter 4, a simulation is a model of a system's operation over time.

Models and simulations provide solution designers and decision makers with insights into possible futures of a system given specific conditions. It is important that these conditions are well defined to support the study and its outcomes. In its Defense Acquisition Guidebook [25], the DOD's Defense Acquisition University recommends that modeling and simulation analyses specify the following at the beginning of any study:

- *Ground rules*, including operational scenarios, threats, environment, constraints, and assumptions.
- *Alternative descriptions*, including operations and support concepts.
- *Effectiveness measures*, showing how alternatives will be evaluated.
- *Effectiveness analysis*, including methodology, models, simulations, data, and sensitivity analysis.
- *Cost analysis*, including life cycle cost methodology, models and data, cost sensitivity, and risk analysis.

11.5.3 Design of Experiments

Having identified alternatives that meet stakeholders' needs, wants, and desires, the solution design team must develop a plan for getting the most information about the operation of those alternatives with the least amount of effort. A mathematical process for accomplishing this is the *Design of Experiments* (DOE), developed in the 1920s and 1930s by Ronald A. Fisher [26], a mathematician who worked at the Rothamsted Agricultural Experimentation Station, north of London (after whom the statistical *F-test* was named), wanted to understand the effects of fertilizer, soil, irrigation, and environment on the growth of grain. His technique, published in 1936, examined the main and interaction effects of key variables (factors) in a system design. The development of Fisher's process is chronicled in *Lady Tasting Tea: How Statistics Revolutionized Science in the Twentieth Century* [27].

Concepts DOE provides solution designers with a way to simultaneously study the individual and interactive effects of many factors, thus keeping the number of experiment iterations (replications) to a minimum. The basic question addressed by DOE is, "What is the average outcome (effect) when a factor is moved from a low level to a higher level?" Since more than one factor is at play in a complex system, efficiencies can be gained by moving combinations of factors simultaneously.

Consider an experiment for the new rocket design developed in this text. The solution design team may have initially determined that the key factors are engine thrust and number of stabilizing fins. They also determined the constraints for each factor: The engine thrust must fall between 1000 and 3000 lb at launch, and competing concepts call for 3-fin and 5-fin designs. The primary measure of effectiveness is "distance from target impact point." In this design, the two factors, A and B, are shown with both low (−) and high (+) levels, representing the upper and lower bounds on the constraints (Figure 11.8). Four conditions are possible in this design: point a, where both A and B are at their lower levels; point b, where A is low and B is high; point c, where both A and B are high; and d, where A is high and B is low. Table 11.4 shows a design in which each of these points can be tested. Experiments designed this way are called 2^k *factorial designs*, since they are based on factors and each factor has two possible levels. The k exponent refers to the number of factors being considered—in this case, two. This way of looking at designs tells the experimenter how many possible states—or design points—exist, and it also lays the groundwork for determining the main and interactive effects of combining the factors.

The order in which the table is constructed is important for subsequent calculations. Notice that the column labeled Factor 1 alternates between − and +. This will always be the case, no matter how many design points there are. The second column, labeled in this case Factor 2, will always alternate between two minuses, followed by two pluses. Note that this is because the design moves one factor from its low to its high point while holding other factors constant.

Calculating Main and Two-Way Interaction Effects

> The *main effect* of a factor is the *average change in the response* (the performance measure) that results from moving a factor from its − to its + level, while holding all other factors fixed.

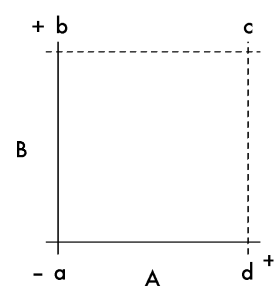

Figure 11.8 Two-factor, two-level design.

TABLE 11.4 A 2 × 2 Design Matrix

Design Point	Factor 1	Factor 2	Response
1	−	−	R_1
2	+	−	R_2
3	−	+	R_3
4	+	+	R_4

The main effect is the effect that a particular factor has, without regard for possible interactions with other factors. In a 2^2 *factorial design*, this is calculated using Equation (11.1) for Factor 1 (e_1).

$$e_1 = \frac{(R_2 - R_1) + (R_4 - R_3)}{2} \tag{11.1}$$

Figure 11.9 shows the relationship between Table 11.4 and Equation (11.1).

Some designers prefer to use capital letters of the alphabet to label factors, while others use numbers. When letters are used, the *effect* nomenclature is the lowercase equivalent of the factor letter. So, the main effect for factor A would be written e_a.

Assume that tests were run on the experimental rocket design and yielded the results shown in Table 11.5. These results show the distance, in meters, from the desired and actual impact points for rockets built using the given constraints. Using Equation (11.1), the main effects of this rocket design can be calculated as follows:

$$e_1 = \frac{(13.0 - 13.6) + (14.8 - 17.6)}{2} = -1.7 \tag{11.2}$$

$$e_2 = \frac{(17.6 - 13.6) + (14.8 - 13.0)}{2} = 2.9 \tag{11.3}$$

This means that by changing from a 3-fin to a 5-fin design, holding thrust constant, there should be an average decrease in error of 1.7 m. Increasing thrust alone

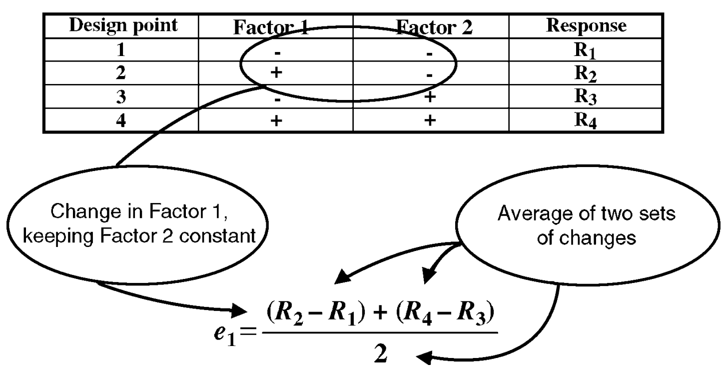

Figure 11.9 Calculating main effects.

TABLE 11.5 Rocket Example Initial Test Results

Design Point	Factor 1	Factor 2	Response
1	−	−	13.6
2	+	−	13.0
3	−	+	17.6
4	+	+	14.8

should result in an average increase in error of 2.9 m. In the equation for e_2, the response for DP 3 is subtracted from that of DP 1, and DP 4 is subtracted from DP 2 before the sum is averaged.

A shortcut that achieves the same mathematical result is to apply the signs of the factor column to the corresponding result (R), sum them, then divide by 2^{k-1}, as shown in Equation (11.4) for calculating the main effect of Factor 2 (e_2).

$$e_2 = \frac{-R_1 - R_2 + R_3 + R_4}{2} \tag{11.4}$$

This method provides a simpler approach for achieving the same result, as shown below:

$$e_1 = \frac{-13.6 + 13.0 - 17.6 + 14.8}{2} = -1.7 \tag{11.5}$$

$$e_2 = \frac{-13.6 - 13.0 + 17.6 + 14.8}{2} = 2.9 \tag{11.6}$$

A strength of DOE is that it also allows the solution design team to understand the *interaction effects* between factors. These effects show the synergistic relationships between factors by measuring how the effect of one factor may depend on the level of another.

> The two-way *interaction effect* of a two-level design is half the difference between the average effects of Factor 1 when Factor 2 is at its + level and when it is at its − level. (All other factors held constant.)

A design matrix with interaction effects is created by adding an interaction column to the standard design. The "sign" values for this column are determined by multiplying the signs of the factors of interest. A matrix for the rocket example that includes the interactions of Factors 1 and 2 (labeled e_{12}) is shown in Table 11.6.

If letters are used to indicate factors, the interaction effect of AB would be written as e_{ab}. Using the definition for interaction effects, the equation for two-way interactions becomes

$$e_{12} = \frac{1}{2}[(R_4 - R_3) - (R_2 - R_1)] \tag{11.7}$$

$$e_{12} = \frac{1}{2}[(14.8 - 17.6) - (13.0 - 13.6)] = -1.1 \tag{11.8}$$

TABLE 11.6 DOE Matrix with Interactions

Design Point	Factor 1	Factor 2	1 × 2	Response
1	−	−	+	13.6
2	+	−	−	13.0
3	−	+	−	17.6
4	+	+	+	14.8

This tells the solution designer that by combining both *treatments*, the overall effect is that the rocket hits 1.1 meters closer to the aim point than without the higher-level alternatives. Following the shortcut logic described earlier, the same result can be found by multiplying the signs of the interaction column (1 × 2) by the response, as shown below:

$$e_{12} = \frac{R_1 - R_2 - R_3 + R_4}{2} \tag{11.9}$$

$$e_{12} = \frac{13.6 - 13.0 - 17.6 + 14.8}{2} = -1.1 \tag{11.10}$$

Designs with More Than Two Factors It is rare for complex systems to be dominated by only two factors. The solution designer must carefully balance the complexity of the design with an ability to find meaningful results. The earlier design could be represented as a two-dimensional square with four corners (design points). The addition of a third factor changes the design from a square to a cube, with 2^3, or eight design points, shown in Figure 11.10. The addition of factor C results in the 2^k factorial design with eight design points shown in Table 11.7. Notice that the repetition of minuses and pluses for Factors 1 and 2 follows a similar pattern as with a 2^2 design. If there were four factors, the pattern of Factor 3 would repeat to fill the 2^4 (16) design points, and Factor 4 would have a series of eight minuses, followed by eight pluses, and so on for however many factors that are being considered.

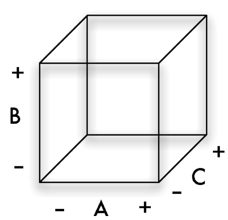

Figure 11.10 Three-factor, two-level design.

TABLE 11.7 2 × 3 Design Matrix

Design point	Factor 1	Factor 2	Factor 3	Response
v1	−	−	−	R1
2	+	−	−	R2
3	−	+	−	R3
4	+	+	−	R4
5	−	−	+	R5
6	+	−	+	R6
7	−	+	+	R7
8	+	+	+	R8

Determining the main and interaction effects of 2^3 designs follows the same logic as for the 2^2 design, although the complexity of the non-shortcut method increases significantly. The two methods for calculating the main effect of Factor 1 described earlier are expanded below to consider three factors.

$$e_1 = \frac{(R_2 - R_1) + (R_4 - R_3) + (R_6 - R_5) + (R_8 - R_7)}{4} \tag{11.11}$$

$$e_1 = \frac{-R_1 + R_2 - R_3 + R_4 - R_5 + R_6 - R_7 + R_8}{4} \tag{11.12}$$

Notice that the denominator is now four, reflecting the four comparisons in the numerator. As before, this value will always be 2^{k-1} for full factorial designs.

When more than three factors are considered, the geometry of the design gets more complex, with the three-way cube becoming part of a larger shape known as hypercube. A four-factor design, for example, can be viewed as two cubes at either end of a line having minus and plus ends that provide for the 16 design points, as shown in Figure 11.11. All 16 design points have unique locations in this geometry. For example, the lower right corner of the right cube has these signs: $\mathrm{A} = +, \mathrm{B} = -, \mathrm{C} = -, \mathrm{D} = +$. Similarly, a 2^5 hypercube design has 32 design points and would appear to have cubes on the corners of a 2^2 (square) design.

Blocking and Randomization A major concern facing solution designers as they consider their strategic approach is that of controllable and uncontrollable factors. This is more of a problem in physical experiments than in simulation-based experiments, where all factors are controllable. Physical experiments generally have an initialization period before reaching a steady operational state. Likewise, samples taken from different lots may show significant variation not found within a lot. Two methods for managing these variations are *blocking and randomization*.

A *block* is a portion of the experimental material that is expected to be more homogeneous than the aggregate [28]. *Blocking* takes known, unavoidable sources of variation out of the design picture. For example, a situation in which conditions are expected to be the same at a given time of day could be *blocked on time-of-day*,

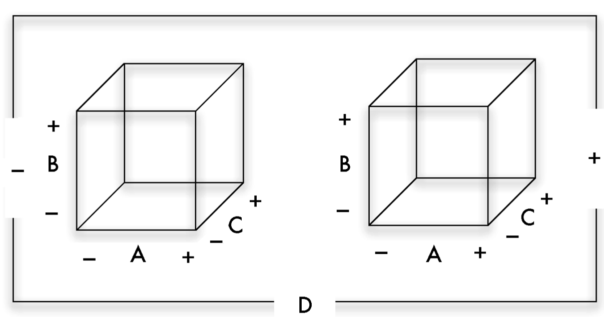

Figure 11.11 Geometry for a 2^4 factorial design.

taking that out of the analysis. Another example might be an experiment comparing hand–eye coordination between dominant and nondominant hands. An approach *blocked on people* would be to select 100 people at random and have each one throw 10 darts at a target using the right hand and 10 with the left. Then compare the difference in results for each person, thus taking individual skill, ability, and similar person-specific variables out of the mix. The result is an experiment isolated on the test issue and avoiding influence by *noise* factors. By confining treatment comparisons with such blocks—say Box, Hunter, and Hunter [28]—greater precision can often be obtained.

The sequence in which trials are run within an experiment must also be planned by the solution design team. A single trial of an experiment that contains random variables is not sufficient to base findings. Concluding that a coin flip will always result in "heads," based on a single flip (trial), is not supportable. "Block what you can and randomize what you cannot" [28] is a useful mantra for the solution designer.

11.5.4 Fractional Factorial Design

Factorial techniques explored so far are reliable ways to identify all main and interaction effects in solution designs when the number of factors is small. But even with increasing computing speed, designs quickly become unmanageable as the number of factors grows. A 2^5 design, for example, has 32 design points, while a 2^7 design has 128. It is not uncommon for designs to be affected by 15 or more factors, which would require 32,768 or more design points. Designs with random variables further increase complexity since they require multiple replications to determine central tendencies. A two-level design with 15 factors, with each design point requiring 10 replications, would require 327,680 separate iterations. Some designers use the terms *replications, runs, and design points* interchangeably. Care must be taken to differentiate between single design points, as used here, and *iterations*, which denote multiple trials to account for randomness.

Fractional factorial designs present two useful solutions to the problem of scale. First, they allow designers to achieve nearly the same results with a fraction of the effort, and second, they provide designers with a tool to further screen out factors that do not make a significant contribution to the outcome.

Fractional designs provide similar results as from a 2^k design, but with 2^{k-p} design points, where p determines what fraction of the total design will be used. We have seen that a full three-factor design requires $2^3 = 8$ design points. If p is set to 1, half as many design points are required, since $2^{3-1} = 4$. This is known as a half-fraction design. A p of 2 produces a quarter-fraction design that would require only a quarter as many design points to achieve nearly the same result.

Having fewer design points does not come without a cost, however. Precision is increasingly lost as the fraction increases. Solution designers must be aware of these tradeoffs and balance the loss of precision with savings in resources (time, material, etc.).

Consider the full factorial design with two levels and four factors shown in Table 11.8. There is one mean effect, four main effects, six two-way effects, four three-way effects, and one four-way effect, for a total of 16 design points.

A half-fraction factorial design promises to provide nearly the same outcome with half as many design points. To do this, start with a full factorial design with $2^{k-p} = 2^{4-1} = 2^3 = 8$ design points, as shown in Table 11.9. Provide for the fourth factor by multiplying the signs of 1, 2, and 3.

The half-fraction matrix directly accounts for eight of the 16 original effects: one mean effect (no contrast between design points—all values are minus), four main effects, and three two-way interaction effects. The "missing" effects—three two-way, four three-way, and one four-way—can be found by examining them separately, shown in Table 11.10.

TABLE 11.8 Full 2^4 Factorial Design

	1	2	3	4	12	13	14	23	24	34	123	124	134	234	1234
DP1	−	−	−	−	+	+	+	+	+	+	−	−	−	−	+
DP2	+	−	−	−	−	−	−	+	+	+	+	+	+	−	−
DP3	−	+	−	−	−	+	+	−	−	+	+	+	−	+	−
DP4	+	+	−	−	+	−	−	−	−	+	−	−	+	−	+
DP5	−	−	+	−	+	−	+	−	+	−	−	−	+	+	+
DP6	+	−	+	−	−	+	−	−	+	−	+	−	+	+	−
DP7	−	+	+	−	−	−	+	+	−	−	−	+	−	+	+
DP8	+	+	+	−	+	+	−	+	−	−	−	+	+	−	+
DP9	−	−	−	+	+	+	−	+	−	−	+	−	−	−	−
DP10	+	−	−	+	−	−	+	+	−	−	−	+	+	+	−
DP11	−	+	−	+	−	+	−	−	+	−	+	−	−	+	+
DP12	+	+	−	+	+	−	+	−	+	−	+	−	+	−	+
DP13	−	−	+	+	+	−	−	−	−	+	−	+	−	−	−
DP14	+	−	+	+	−	+	+	−	−	+	−	−	+	−	−
DP15	−	+	+	+	−	−	−	+	+	+	−	−	−	+	−
DP16	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

TABLE 11.9 2^{4-1} Design Matrix

	1	2	3	4(123)	12	13	23
DP1	−	−	−	−	+	+	+
DP2	+	−	−	+	−	−	+
DP3	−	+	−	+	−	+	−
DP4	+	+	−	−	+	−	−
DP5	−	−	+	+	+	−	−
DP6	+	−	+	−	−	+	−
DP7	−	+	+	−	−	−	+
DP8	+	+	+	+	+	+	+

TABLE 11.10 The Lost 2^4 Effects

	14	24	34	123	124	134	134	1234
DP1	+	+	+	−	−	−	−	+
DP2	+	−	−	+	−	−	+	+
DP3	−	+	−	+	−	+	−	+
DP4	−	−	+	−	−	+	+	+
DP5	−	−	+	+	+	−	−	+
DP6	−	+	−	−	+	−	+	+
DP7	+	−	−	−	+	+	−	+
DP8	+	+	+	+	+	+	+	+

Notice the relationships of the signs in the factor columns. The sequence for factor 14 (shorthand for 1×4) is the same as that for 23 in Table 11.9. Comparing all the columns provides the data for Figure 11.2. A striking observation in Figure 11.2 is that every effect involving Factor 4 is algebraically the same as another effect not using factor 4. These identical terms are said to be *aliases* of each other. They are also described as being *confounded* by one another, since the effect is calculated exactly the same way and therefore it is impossible to know which effect is producing the result. Notice that Figure 11.12 now accounts for all 16 main and interaction effects.

This suggests that the same result should be found using a 2^{4-1} design as with a full 2^4. This relies on the *sparsity of effects* assumption that states that if an outcome is possible from several events, the less complex event is most likely the cause. So, if faced with two identical outcomes, one resulting from a single main effect and one resulting from the interaction of several effects, this assumption claims that the single main effect is most likely the cause and the complex effect can be ignored. This assumption, though convenient, is a source of loss of precision and must be used with caution.

	1	2	3	4	12	13	23	14	24	34	123	124	134	234	1234
DP1	–	–	–	–	+	+	+	+	+	+	–	–	–	–	+
DP2	+	–	–	+	–	–	+	+	–	–	+	–	–	+	+
DP3	–	+	–	+	–	+	–	–	+	–	+	–	+	–	+
DP4	+	+	–	–	+	–	–	–	–	+	–	–	+	+	+
DP5	–	–	+	+	+	–	–	–	–	+	+	+	–	–	+
DP6	+	–	+	–	–	+	–	–	+	–	–	+	–	+	+
DP7	–	+	+	–	–	–	+	+	–	–	–	+	+	–	+
DP8	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

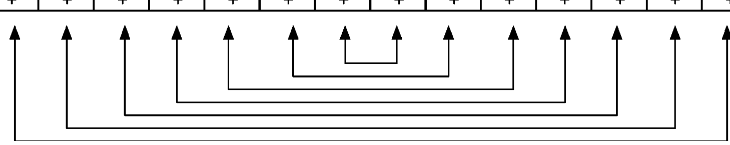

Figure 11.12 Fully expanded 2^{4-1} matrix.

Another observation from Figure 11.12 is that the interaction effect 1234 is composed of all plus signs. Comparing this table with Table 11.8, it becomes apparent that the fractional matrix is identical to the full matrix design points 1, 10, 11, 4, 13, 6, 7, and 16. The annotated hypercube in Figure 11.13 confirms that the fractional design symmetrically targets half of the possible points of a full design. The corners not selected in Figure 11.13 represent the half fraction where factor 1234 is a column of all minus signs. Either half could be used, although the one shown uses the principal fraction (all pluses) and is most common. The column with all plus signs is also known as the *identity* column, and it has a number of unique attributes. Most obvious, it is the only column that does not have an equal number of plus and minus signs. But more importantly, it reflects the fact that any sign multiplied by itself results in a positive value. In the fractional design above, Factor 4 was determined by multiplying factors 1, 2, and 3. Therefore, $4 = 123$, which produced the identity column I, which equals 1234, or $I = 1234$. This is called the *design generator*, because it can be used to generate the entire aliasing structure for the design, eliminating the need for pattern-match tables of pluses and minuses.

To find the alias for a factor, simply multiply the factor times I. Finding the alias for factor 2 in the previous design, for example, is accomplished as shown in steps below:

- Step 1: $I = 1234$, establish the design generator
- Step 2: $I = 2 \times 2$
- Step 3: $2I = (2)1234$
- Step 4: $2(2 \times 2) = (2 \times 2)134$
- Step 5: $2 = 134$, which is verified in Figure 11.12

A simplified rule of thumb for finding aliases is to drop the column in question from the *generating relation*. Therefore, in this example, $1 = 234$; $2 = 134$; $3 = 124$; $12 = 34$; and so forth. Higher-order fractions (quarter, eighth, etc.) have more than one generator, but the fundamental process is the same.

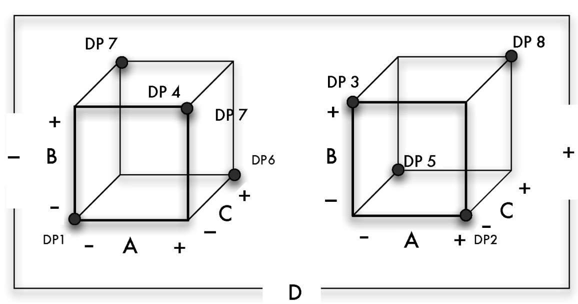

Figure 11.13 2^{4-1} design plot (principal fraction).

Calculating main and interaction effects in fractional factorial designs is accomplished in the same way as with full factorial designs, except that the denominator is 2^{k-p-1} instead of 2^{k-1}. The main effect of Factor 1 for the half-fraction design shown above is developed in Equation (11.14).

$$e_1 = \frac{-R_1 + R_2 - R_3 + R_4 - R_5 + R_6 - R_7 + R_8}{2^{4-1-1}} \tag{11.13}$$

$$= \frac{-R_1 + R_2 - R_3 + R_4 - R_5 + R_6 - R_7 + R_8}{2^4} \tag{11.14}$$

Given a choice of conducting a full factorial experiment with many design points or a fractional design with many fewer, why would a solution designer consider anything but the one with the fewest? The answer is in the balancing of resources available and the loss of precision inherent in fractional designs. The key to achieving that balance lies in the concept of design *resolution*.

> Design resolution is a measure of the degree of confounding that exists in a fractional factorial design.

Generally, and with notable exceptions, the highest resolution design the experimenter can afford to conduct is the preferred choice. Screening designs, discussed later, are the main exceptions to the rule.

The level of resolution is shown using roman numerals to distinguish it from other numbers in the design notation. The numeral indicates the level at which fractional designs are clear of confounding. Two effects are not confounded if the sum of their *ways* is less than the resolution of the design. Main effects are considered one-way effects, while interaction effects such as 12 and 34 are two-way, 134 and 234 are three-way, and so on. In a resolution IV design, no main effects are aliased with any two-way effect ($1 + 2 < 4$), but two-way effects are aliased with other two-way or higher order effects. ($2 + 2$ or more < 4).

Knowing that higher-resolution designs have less confounding and therefore less precision loss, yet will require more design points, the solution designer can balance available resources against precision in constructing an experimental design. It is also important that the designer indicate this tradeoff in discussing the design. The notation for fractional factorial designs is shown in Figure 11.14.

Tables for quickly determining which designs best meet the resolution criteria exist in much of the DOE literature, including *Statistics for Experimenters* [28] and *Simulation Modeling and Analysis* [29]. The table of designs where $k \leq 7$, shown in Table 11.11, illustrates how greater confounding occurs at lower resolutions.

Lower-resolution designs are particularly useful as *screening designs* to eliminate factors that do not contribute significantly to the system's operation. Caution and experience are necessary to prevent too low of a resolution from being selected and a resulting greater loss of precision leading to faulty conclusions.

While feasibility screening, described in Section 11.3.2, provides a *criteria-based* tool for narrowing alternatives, factorial screening provides a *merit-based*

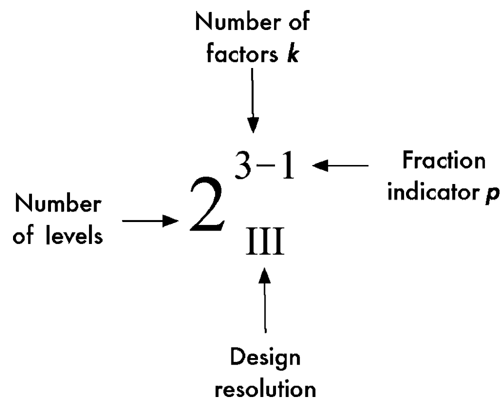

Figure 11.14 Fractional factorial design notation.

TABLE 11.11 Confounding in Fractional Factorial Design

	Resolution		
k	III	IV	V
3	$2_{III}^{3-1} \rightarrow 3 = \pm 12$		
4		$2_{IV}^{4-1} \rightarrow 4 = \pm 123$	
5	$2_{III}^{5-2} \rightarrow 4 = \pm 12$ $5 = \pm 13$		$2_{V}^{5-1} \rightarrow 5 = \pm 1234$
6	$2_{III}^{6-3} \rightarrow 4 = \pm 12$ $5 = \pm 13$ $6 = \pm 23$	$2_{IV}^{6-2} \rightarrow 5 = \pm 123$ $6 = \pm 234$	
7	$2_{III}^{7-4} \rightarrow 4 = \pm 12$ $5 = \pm 13$ $6 = \pm 23$ $7 = \pm 123$	$2_{IV}^{7-3} \rightarrow 5 = \pm 123$ $6 = \pm 234$ $7 = \pm 134$	

tool to further refine the solution space. Revisiting the rocket scenario described earlier, the solution designers' feasibility screening matrix shows that five factors pass the initial test, as shown in Table 11.12.

A full factorial design is developed and tested, with the response being a score of 0–100 where 100 is a bull's-eye target hit. The results are shown in Table 11.13.

Several statistical software packages, such as Minitab®, allow solution designers to construct and analyze factorial designs. Analysis of the responses reveals that only the *seeker type, guidance system*, and *thrust* main effects and the interactions of *seeker-guidance* and *guidance-thrust* have a statistically significant effect on system performance, given an alpha value of 0.05.

Skin type and *number of fins* did not make a significant difference in rocket performance, as seen in the Minitab® chart shown in Figure 11.15. The vertical line at 3.12 on the horizontal axis indicates the threshold for significance.

TABLE 11.12 Modified Rocket Design Factors

Factor	Name	Low (−)	High (+)
1	Skin composition	Aluminum	Composite
2	Seeker	Forward-looking infrared	Laser-guided
3	Fins	3	5
4	Guidance	Inertial navigation	Global positioning system
5	Thrust	1000 lb	3000 lb

TABLE 11.13 Rocket Design Test Results

R1 = 61	R2 = 53	R3 = 63	R4 = 61
R5 = 53	R6 = 56	R7 = 54	R8 = 61
R9 = 69	R10 = 61	R11 = 94	R12 = 93
R13 = 66	R14 = 60	R15 = 95	R16 = 98
R21 = 59	R22 = 55	R23 = 67	R24 = 65
R25 = 44	R26 = 45	R27 = 78	R28 = 77
R29 = 49	R30 = 42	R31 = 81	R32 = 82

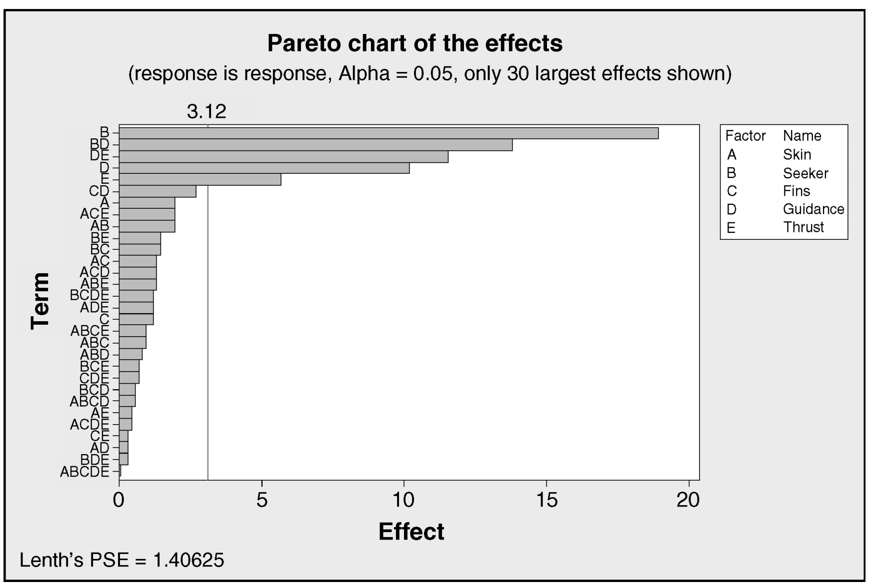

Figure 11.15 Pareto chart of effects, full rocket design.

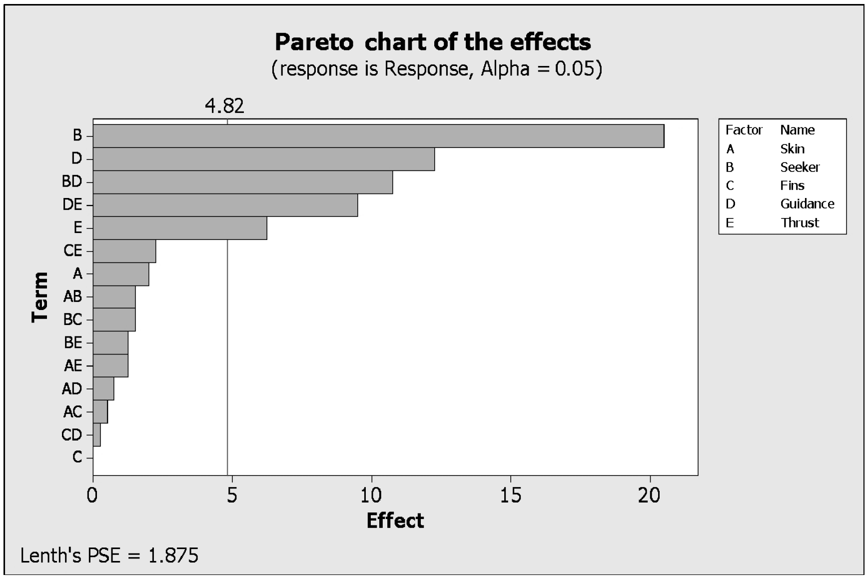

Figure 11.16 Pareto chart of effects, half-fraction rocket design.

A 2_V^{5-1} design should produce essentially the same findings, but with half the physical testing and the associated time and material costs required by the full factorial experiment. Figure 11.16 shows the Minitab® results. Once again, *seeker, guidance*, and *thrust* main effects and *seeker-guidance* and *guidance-thrust* interaction effects are shown to be significant. *Skin type* and *number of fins* could be screened out of the design based on the rocket performance.

Survey of Other Design Strategies Full and fractional factorial designs as described earlier are common, general-purpose approaches to experimental design. However, many others are available to the solution designer, each designed to meet unique situations that may be encountered.

Plackett–Burman designs are two-level, resolution III fractional factorial designs often used for screening when there is a large number of factors and it is assumed that two-way interactions are negligible. Recall that in a resolution III design, main effects are confounded with two-way interactions ($1 + 2 \geq 3$). Introduced in 1946 [30], it is based on the number of design points being a multiple of four instead of a power of two, as seen earlier. The number of factors must be less than the number of design points, so a 20 DP design can estimate main effects for up to 19 factors. In general, these designs will estimate main effects for n factors using $n + 1$ design points, rounded up to the nearest multiple of four.

Latin Squares designs are used when there are more than two factors and there are assumed to be only negligible interaction effects. They are an offshoot of magic square designs that trace their roots to ancient Asia, and are the source of the popular sudoku puzzles. The concept was published in Europe in 1782 by Leonhard Euler [31] and involved a square matrix of n-by-n cells containing n symbols, each of which occurs only once in each row and column. In Latin Square designs, one factor, or treatment, is of primary interest and the remaining factors are considered blocking, or nuisance, factors. Treatments are given Latin characters and are arranged in a matrix. The blocking factors are represented by the rows and columns.

A modified rocket design to explore the effect of four propellant mixtures on thrust is shown in Figure 11.17. This design is blocked on two factors: seeker and guidance system. It is recognized that slight variations may exists between individual components, and this design will eliminate seeker-to-seeker and guidance-to-guidance differences to get a truer understanding of the primary effect being considered.

Table 11.14 shows the general matrix for a three-level Latin Square design that illustrates how this results in n^2 design points. The same approach is used for designs with more levels. Modeling responses for experimental designs are explored in more depth in Chapter 4, but the fundamental model for Latin

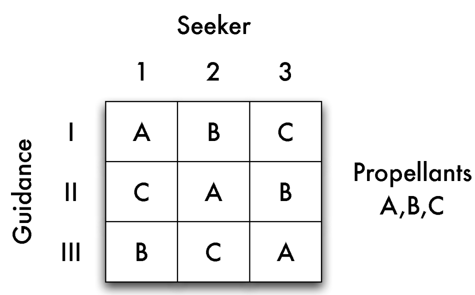

Figure 11.17 Latin Square rocket design focused on thrust.

TABLE 11.14 Three-Level Latin Square Matrix

DP	Row Blocking Factor	Column Blocking Factor	Treatment Factor
1	1	1	1
2	1	2	2
3	1	3	3
4	2	1	3
5	2	2	1
6	2	3	2
7	3	1	2
8	3	2	3
9	3	3	1

Square and related designs follows the form, where Y is the observation, η is the mean, R_i is the row effect, C_j is the column effect, T_k is the treatment effect, and ϵ_{ijk} is random error.

$$T_{ijk} = \eta + R_i + C_j + T_k + \epsilon_{ijk} \tag{11.15}$$

Graeco-Latin and Hyper-Graeco-Latin Square designs are extensions of the Latin Square method for studying more than one treatment simultaneously. Graeco-Latin squares, also introduced by Euler, are constructed by superimposing two Latin Square designs into one matrix and differentiating them by using Greek letters for one of the treatments. This design uses three blocking factors, instead of two. A Hyper-Graeco-Latin Square design considers more than three blocking variables. A Graeco-Latin Square design for a rocket system experiment conducted over three days, the new blocking factor, is shown in Figure 11.18.

Response Surface Method (RSM) designs are used to examine the relationship between a response variable and one or more factors, especially when there is *curvature* between a factor's low and high values, which has been assumed to not exist in designs so far. The full and fractional factorial designs described earlier were assumed to follow a straight line between values. RSM designs reveal the *shape* of a response surface, and are useful in finding *satisficing* or *optimal* process settings as well as weak points.

Figure 11.19, extracted from a Minitab® run of the full factorial design responses from the rocket design, clearly shows a nonlinear relationship between the number of fins and the choice of guidance system. Both plots show the greatest response from the increase in guidance system technology, which is consistent with the Pareto analysis shown in Figures 11.15 and 11.16. What is particularly revealing in both the *surface plot* on the left and the *contour plot* on the right, however, are the spikes in responses when the number of fins is at low and high points. This suggests that if, for example, the higher technology guidance system is not available or is too expensive, the rocket should achieve better results with a 5-fin design than

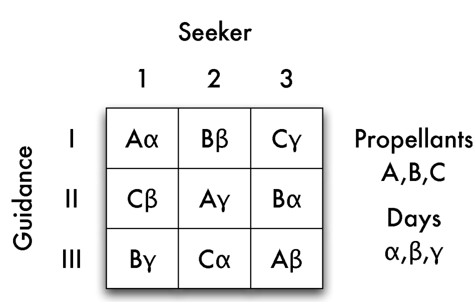

Figure 11.18 Graeco-Latin rocket design.

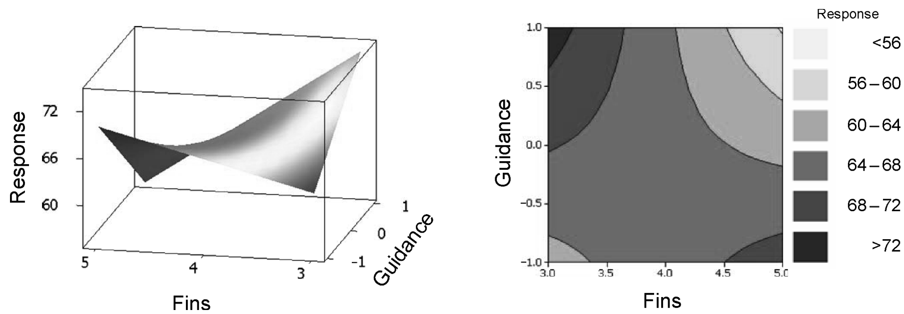

Figure 11.19 Response surface plots of the rocket design.

something less. Clearly, considering these interactions alone, a 4-fin design is least desirable using either guidance systems.

Other Design Techniques have been developed and are in use by solution designers. Though not as universally adopted as the techniques described above, these methods include modified factorial designs proposed by Taguchi in *System of Experimental Design* [32] and variations on Latin hypercubes, including nearly orthogonal designs by Cioppa [33].

11.5.5 Pareto Analysis

Pareto analysis further reduces the complexity of the solution space by critically examining factors that make up a solution and eliminating those that fail to make a meaningful contribution. It continues the drive to condense the problem to only the most essential factors.

The Pareto principle, also known as the *sparsity principle*, the *law of the vital few*, and the *80–20 rule*, was coined by quality management expert Joseph M. Juran in the 1951 edition of his *Quality Control Handbook* [34]. In a 1975 explanation, "The Non-Pareto Principle; Mea Culpa" [35], Juran traces how in the 1930s he came to understand that a small number of factors usually make the greatest contribution to the whole. He confesses that he extended Italian economist Vilfredo Pareto's concepts on the unequal distribution of wealth beyond their intended scope. In Le *Cour d'Economie politique* [36], Pareto noted that 80% of the wealth was owned by 20% of the people. Juran noticed that this ratio held true for many quality issues and used the term *Pareto principle* to describe his concept of the vital few and trivial many. Pareto analysis, regardless of the accuracy of the name, is now widely used to separate the *vital few* from the *trivial many* in many different applications.

Common Pareto analyses use a combination of bar and line charts to visually identify the vital few. Raw values are typically displayed in a bar chart sorted in decreasing order from left to right. A secondary y-axis shows a cumulative distribution using a line chart. Figure 11.20 uses this approach to display the results of the full factorial rocket design. A key to interpreting this kind of Pareto chart is to look for a *knee in the curve* of the cumulative distribution. This will indicate

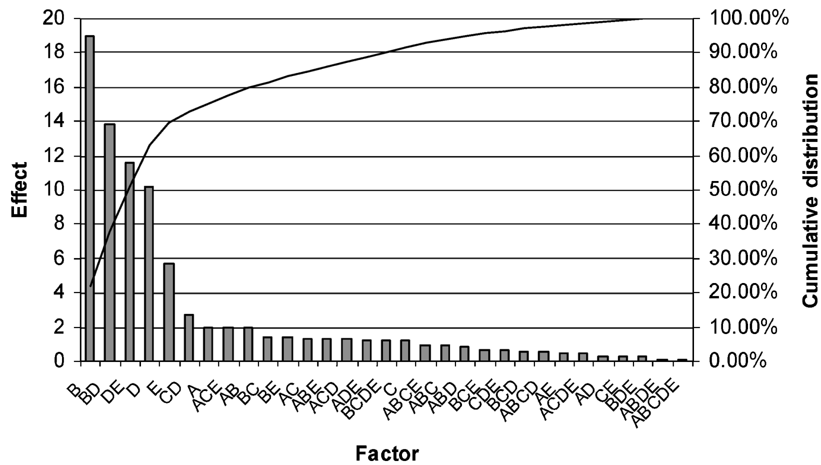

Figure 11.20 Pareto analysis of the rocket design experiment.

a point where the degree of contribution begins to level off, with factors on the steeper slope making a greater contribution. In Figure 11.20, this occurs at about the 70% point. If there is no distinct change in the slope of the line, a conservative approach is to consider factors that contribute 60% of the total. This reinforces Juran's confession that the 80–20 rule is more of a rule of thumb and should not be interpreted as an absolute metric. Table 11.15 summarizes the factors contributing to this point. Figure 11.20 is strong supporting evidence for screening out the trivial many factors in this solution design.

The Minitab® charts in Figures 11.15 and 11.16 show another method for conducting a Pareto analysis, with main and interaction effects plotted in decreasing order and a vertical line marking the boundary of statistical significance. In this case, the vital few are those that exceed the threshold for significance. Note that this figure reveals a conclusion that is identical to that of Figure 11.20: the five factors named in Table 11.15 are the most important vital few factors in this design.

Pareto analysis can be used throughout the systems decision process, whenever there is a large set of values to be prioritized. Other opportunities arise during brainstorming or stakeholder analysis, for example, to reduce the number of inputs to the vital few.

TABLE 11.15 The Vital Few Rocket Design Factors

Factor	Factor Name	Response	Percent of Total	Cumulative Percent
B	Seeker	18.938	21.97	21.97
BD	Seeker* guidance	13.812	16.02	38.00
DE	Guidance* thrust	11.563	13.42	51.41
D	Guidance	10.187	11.82	63.23
E	Thrust	5.688	6.60	69.83

11.6 SUMMARY

Successful solutions to complex problems are products of design, not chance. *Design* is a deliberate process for composing an object, architecture, or process. A *solution* is a process for solving a problem that results in feasible *alternatives* to present to a decision maker. *Solution design*, then, is a deliberate process for composing a set of feasible alternatives for consideration by a decision maker. As seen in Chapter 7, systems engineers build solution design teams specifically suited for the problems at hand. This chapter explores ways that those teams generate innovative ideas and refine them into alternatives that meet stakeholders' needs, wants, and desires. Solution designers further analyze alternatives through modeling and simulation, examined in Chapter 4, to provide greater understanding of how the system works over time. Systems engineers can then confidently provide decision makers with essential candidate solutions that require only the leader's experience and skill for a decision.

11.7 ILLUSTRATIVE EXAMPLE: SYSTEMS ENGINEERING CURRICULUM MANAGEMENT SYSTEM (CMS)—SOLUTION DESIGN

SOLUTION DESIGN

Robert Kewley, Ph.D. U.S. Military Academy

For the solution design phase of this problem, the system designers had to scan the environment for potential IT solutions for the problem identified in the previous chapter. On the basis of the unique constraints and challenges identified, they had to come up with a feasible subset of IT solutions that could be expected to solve the problem. This is truly a value-focused approach because they have defined a broad set of value-added functions that the system could support with limited understanding of the alternatives' abilities to achieve that functionality. They are asking the question, "What do we want the system to do?" as opposed to asking, "Which of these predefined alternatives helps us most?" With this approach, they are more likely to identify opportunities to improve curriculum management functions.

Once they had scanned the IT environment for potential software and development solutions, the team formulated the problem as a sequence of interrelated decisions. For this reason, they used Zwicky's morphological box, shown in Figure 11.21, to represent these alternatives. Each column of the box represents one of the development decisions for the design team. The options for each decision are represented in the rows.

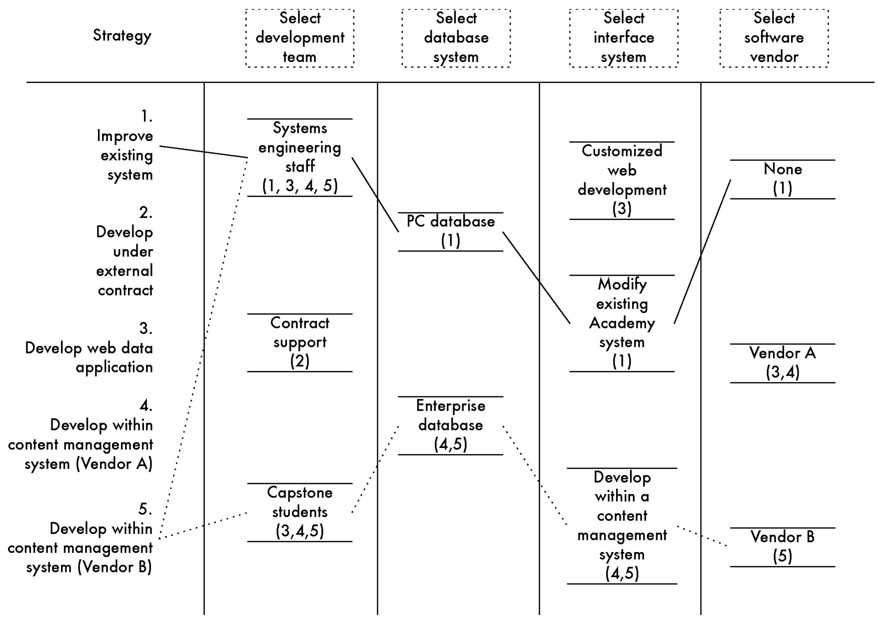

Figure 11.21 Alternative generation table.

Although this box gives a possibility of 54 ($3 \times 2 \times 3 \times 3$) alternatives, dependencies reduce this significantly. Other alternatives are not feasible or clearly inferior with no analysis required. The design team reduced the alternatives to a feasible set of five for which they would do additional analysis.

1. *Improve the Existing System.* Department IT personnel work with department leadership to develop a set of structured templates, storage folders, and processes to be applied to the current system of curriculum management. This alternative does not require significant IT development. Instead it seeks to identify, standardize, and enforce best practices employed in the current system.
2. *Develop the System Under External Contract.* The department hires an external contractor to develop an enterprise-wide system using database and interface solutions determined by the contractor and approved by the department. This outsourcing approach removes the development load from the department's IT staff, but requires significant external coordination with a contractor who may not understand department processes.
3. *Develop Web-Data Application.* Department IT personnel, supported when possible by capstone students (undergraduate students doing a two semester team research project), develop a Web-data application to handle

the curriculum management functions. They do this using Vendor A, Web services, and a service-oriented architecture to maximize usability of services and integration with existing systems. This alternative provides excellent opportunities for flexibility and customization, but it places a heavy burden on the department IT staff.

4. *Develop within a Content Management System (Vendor A).* Department IT personnel, supported when possible by capstone students, performs development within the framework of a content management system from vendor A to integrate structured curriculum data into the portal. With this alternative, the content management system provides a significant portion of the required capabilities with limited or no development required. It does have some additional cost.
5. *Develop within a Content Management System (Vendor B).* This alternative is distinguished from the previous one by a different vendor for the content management system. Vendor B's system has different functionality and cost. Given these alternatives, the design team must score each alternative against the value measures shown in the previous chapter in Figure 10.20.

Because this is a system concept decision, the team does not have sufficient design data to do modeling and analysis of the alternatives. For this decision, it is sufficient to research each alternative to subjectively assess the objectives in the values hierarchy.

11.8 EXERCISES

11.1. Identify and describe the seven main criticisms of traditional brainstorming sessions.

11.2. Delphi methods for idea generation seek to minimize some of the biasing effects of traditional brainstorming. Identify the three effects Delphi methods seek to counter and the three unique variations they use to do it. **Situation**. Assume that you are a member of a solution design team tasked with developing solution candidates for a new parking lot car-finder. The client wants to market a device that will allow shoppers to instantly find their cars in a crowded shopping mall parking lot. The client tells you that the device must be simple to use, very inexpensive to make but durable and reliable, and priced for a middle class mass market. Additionally, it must function in all weather conditions and be effective up to 100 m. It would be nice, he adds, if it could come in designer colors and styles to fit individual personalities. Finally, it must be completely secure, so that no legal action could reflect on the company if it is stolen or misused. Assume that one solution candidate considered material (plastic or metal), indicator (visual or electronic), and transmitter (none or radio frequency). Further assume that a trial takes place with shoppers executing the full factorial design. The average times for shoppers to find their cars for each design point are in the order 145, 133, 128, 118, 127, 110, 102, and 98 s.

11.3. Conduct a morphological analysis of the car-finder device that includes a two-dimensional matrix with at least four parameters and four variables.

11.4. Construct an ends-means chain to illustrate the objectives structure of the car-finder device.

11.5. Using the results of the morphological analysis, conduct a feasibility screening of the alternatives by the client's needs, wants, and desires.

11.6. Referring to the techniques explored in Chapter 4, identify the types of models appropriate for analyzing solution candidates that pass the feasibility screening test, and explain why they are appropriate.

11.7. Construct a full factorial experimental design matrix with all main and interaction effects, listing the factors in the order described above.

11.8. Calculate the main effects of changing each of the three factors.

11.9. Calculate the interaction effects of modifying material and indicator together and the three-way interactions of material, indicator, and transmitter.

11.10. Conduct a Pareto analysis of the car-finder experiment. Identify the vital few and the trivial many and explain how you reached that conclusion.

11.11. Reviewing your original morphological analysis, if only five alternatives passed a subsequent feasibility screening and each factor could be reduced to a single low and high value, how many design points would be required to conduct a full factorial design of these solution candidates?

11.12. Reducing the design in Question 11 to a Resolution III fractional factorial design would require what number of design points?

11.13. Fractional factorial designs run the risk of confounding effects. What combination of effects would be aliased, if any, in the design described in Question 12?

REFERENCES

1. *Webster's II New Riverside University Dictionary*. Boston, MA: Houghton Mifflin, 1988.
2. INCOSE SE Terms Glossary. Seattle: International Council on Systems Engineering, 1998.
3. Dalkey, NC. *The Delphi Method: An Experimental Study of Group Opinion*. Santa Monica, CA: United States Air Force Project RAND, 1969.
4. Osborn, AF. *How to Think Up*. New York: McGraw-Hill, 1942.
5. Osborn, AF. *Your Creative Power: How to Use Imagination*. New York: Charles Scribner's Sons, 1948.
6. Osborn, AF. *Wake Up Your Mind: 101 Ways to Develop Creativeness*. New York: Charles Scribner's Sons, 1952.

7. Osborn, AF. *Applied Imagination: Principles and Procedures of Creative Thinking*. New York: Charles Scribner's Sons, 1953.
8. Durfee, WK. Brainstorming Basics, ME 2011 Course Handout. Minneapolis, MN: University of Minnesota, 1999.
9. De Bono, E. *Serious Creativity: Using the Power of Lateral Thinking to Create New Ideas*. New York: HarperBusiness, 1992.
10. Isaksen, SG. A review of brainstorming research: Six critical issues for inquiry. Buffalo, NY: Monograph 302, Creative Problem Solving Group, 1998.
11. Taylor, DW, Berry, PC, Block, CH. Does group participation when using brainstorming facilitate or inhibit creative thinking? *Administrative Science Quarterly*, 1958;3:22–47.
12. Sage, AP, Armstrong, JE, Jr. *Introduction to Systems Engineering*. Wiley Series in Systems Engineering, New York: Wiley-Interscience, 2000.
13. Janis, IL. *Groupthink: Psychological Studies of Policy Decisions and Fiascoes*, 2nd ed. Boston, MA: Houghton Mifflin, 1983.
14. Dalkey, N, Helmer, O. An experimental application of the Delphi method to the use of experts. *Management Science*, 1963;9:458–467.
15. Girshick, M, Kaplan, A, Skogstad, A. The prediction of social and technological events. *Public Opinion Quarterly*, 1950;14:93–110.
16. Fowles, J, Fowles, R. *Handbook of Futures Research*. Westport, CT: Greenwood Press, 1978.
17. DeBono, E. *The Use of Lateral Thinking*. London, England: Jonathan Cape Ltd. Publishers, 1967.
18. DeBono, E. *Parallel Thinking*. Penguin Press: New York 1995.
19. Zwicky, F. *Discovery, Invention, Research Through the Morphological Approach*. New York: Macmillan, 1969.
20. Keeney, RL. *Value-Focused Thinking: A Path to Creative Decisionmaking*. Cambridge, MA: Harvard University Press, 1992.
21. Athey, TH. *Systematic Systems Approach: An Integrated Method for Solving Problems*. Boston, MA: Pearson Custom Publishing, 1982.
22. MacCrimmon, KR, Wagner, C. Supporting Problem Formulation and Alternative Generation in Managerial Decision Making. Presented at 24th Annual Hawaii International Conference on System Sciences, Honolulu, HI, 1991.
23. Sage, AP. *Systems Engineering*. New York: John Wiley & Sons, 1992.
24. Nutt, PC. A taxonomy of strategic decisions and tactics for uncovering alternatives. *European Journal of Operational Research*, 2001; 132: 505–527.
25. *Defense Acquisition Guidebook*. Fort Belvoir, VA: Defense Acquisition University, 2004.
26. Fisher, RA. *The Design of Experiments*, 9th ed. New York: Hafner Press, 1971.
27. Salsburg, D. *Lady Tasting Tea: How Statistics Revolutionized Science in the Twentieth Century*. New York: Henry Holt and Company, 2002.
28. Box, GE, Hunter, WG, Hunter, JS. *Statistics for Experimenters*. New York: John Wiley & Sons, 1978.
29. Law, AM, Kelton, WD. *Simulation Modeling and Analysis*. New York: McGraw-Hill, 2000.

30. Plackett, RL, Burman, JP. The design of optimum multi-factorial experiments. *Biometrika*, 1946;33:305–325.
31. Euler, L. Recherches sur une nouvelle espece de quarres magiques. *Verhandelingen/uitgegeven door het Zeeuwsch Genootschap der Wetenschappen te Vlissingen*. 1782;9:85–239.
32. Taguchi, G. *System of Experimental Design*, Vols. 1 and 2. White Plains, NY: UNIPUB/Kraus International Publications, 1987.
33. Cioppa, TM. Efficient nearly orthogonal and space-filling experimental designs for high-dimensional complex models. Ph.D. Dissertation. Monterey, CA: Operations Research Department, Naval Postgraduate School, 2002.
34. Juran, JM. *Quality Control Handbook*. New York: McGraw-Hill, 1951.
35. Juran, JM. The non-pareto principle; mea culpa. *Quality Progress*, 1975;8:8–9.
36. Pareto, V. *Le Cour d'Economie Politique*. London: Macmillan, 1896.
37. GroupSystems website, http://www.groupsystems.com. Accessed August 22, 2010.