Integration of Technology Roadmapping Information and Business Case Development into DMSMS-Driven Design Refresh Planning of the V-22 Advanced Mission Computer

Jessica Myers and Peter Sandborn
CALCE, Department of Mechanical Engineering
University of Maryland

Abstract

Design refreshes and various reactive mitigation solutions are used to manage technology obsolescence (DMSMS) in systems. Design refreshing solely to manage obsolescence is, however, not practical for many systems, and therefore, obsolescence management refresh activities need to be coordinated with technology insertion roadmaps. This report describes the development of an information model that details technology roadmapping information for use within obsolescence-driven design refresh planning, and also describes a business case analysis for ascertaining the value of the resulting refresh plans. A case study on the V-22 Advanced Mission Computer using the MOCA (Mitigation of Obsolescence Cost Analysis) refresh planning tool is described in which optimum refresh plans (coupled with bridge and lifetime buys) with and without the inclusion of technology roadmapping constraints are determined.

Introduction

As the pace of technological progress increases, technology obsolescence problems will have a greater effect on traditionally sustainment-dominated industries. Many organizations rely solely on reactive approaches to manage obsolescence events as they occur, often employing lifetime buys, aftermarket sources and other mitigation approaches to ensure enough parts to last through the platform's lifecycle. Strategically planned design refreshes coupled with various mitigation approaches can, in many cases, lead to greater cost avoidance than reactive mitigation alone.

Design refresh planning is performed by organizations who wish to avoid the high costs of purely reactive obsolescence solutions. Planning to phase-out specific parts at certain times lessens the reliance on reactive solutions (and the resulting quest for obsolete parts) and, in turn, lessens the total cost of sustaining a system. Planned refreshes also make reactive solutions more manageable, i.e., they provide better defined *end* dates for mitigation solutions as opposed to moving targets, which are difficult to manage to. However, design refreshing solely to manage obsolescence is not practical for many systems, and therefore, obsolescence management refresh activities need to be coordinated with technology insertion roadmaps. Technology insertion roadmaps are developed to dictate how the system's functionality and performance must be changed over time to meet evolving customer requirements. Technology roadmaps reflect an organization's internal technology goals and budget cycles, and often reflect the expected or articulated needs of the customer.

The MOCA (Mitigation of Obsolescence Cost Analysis) software tool has been developed to generate and select an optimum design refresh plan for a system [1]. This report discusses extensions to MOCA that include technology roadmapping constraints and an automated business case analysis. The integration of technology roadmap information into MOCA's decision analysis ensures that selected refresh plans meet roadmap imposed timing constraints, and that the costs of roadmap specified actions are included in relevant refreshes. The automated business case analysis presents a quantitative cost avoidance calculation method of comparing the optimum solution to all-mitigation and all-refresh baselines.

These new developments in MOCA are discussed in the context of the V-22 Advanced Mission Computer (AMC) system. The mechanics of the MOCA tool's optimization analysis with roadmapping considerations will be described. The resulting cost avoidance associated with the optimum refresh plan is presented in business case form.

Mitigation of Obsolescence Cost Analysis (MOCA)

A tool has been previously developed to aid organizations in creating a plan for managing part obsolescence before it occurs. The Mitigation of Obsolescence Cost Analysis (MOCA) tool has been designed to output a plan consisting of design refreshes, lifetime buys, and bridge buys where the total sustainment cost of the plan has been minimized, [1]. MOCA takes as its input the bill of materials (BOM) for a given product, along with the procurement cost and projected obsolescence date of the individual parts. MOCA can model multiple levels of hierarchy for the bill of materials, so that an entire system made up of different circuit boards with different parts may be loaded into the tool. MOCA also requires a production schedule as an input, and this production plan along with a forecast of required spares is used to locate all possible refresh dates for the system. MOCA creates a timeline of all possible design refresh dates and couples it with a timeline of all of the projected obsolescence dates for the parts contained in the bill of material. It is assumed that if a part is obsolete at the time of a refresh that it will be refreshed, and a 'look-ahead-time' can be applied at refresh dates so that parts that are about to go obsolete can also be refreshed. MOCA generates candidate refresh plans consisting of zero refresh dates (all lifetime buys), exactly one refresh date in the lifetime of the product, exactly two refresh dates, etc. Every possible candidate plan is generated and ranked according to the total cost of the plan. By selecting the lowest life cycle cost plan, MOCA is able to optimize a system's design refresh plan with respect to cost.

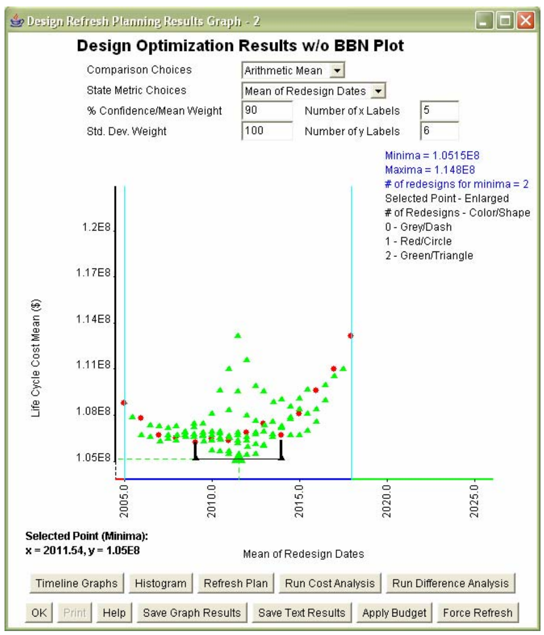


Figure 1 - Sample MOCA output.

Figure 1 shows an example MOCA output, where each dot on the graph represents a unique refresh plan. Refresh plans can contain anywhere from zero to six actual refresh dates (the result in Figure 1 contains plans with exactly one or two refreshes in them). At these refresh dates MOCA generates a list of parts that are obsolete or about to go obsolete so that they can be refreshed. Parts that become obsolete before the designated refresh date are managed using a user defined "short term" mitigation scenario (in the V-22 example discussed in this report, the parts are bridge bought) until they can be replaced. The cost of the bridge buy, along with the storage and handling costs and the costs of the design refresh itself are all included in MOCA's total life cycle cost calculation for each refresh plan. Once the plans have been generated and their costs estimated they are represented on plots like the one in Figure 1. The vertical axis on the graph is life cycle cost and the horizontal axis is time. The data points corresponding to the plans are plotted at the mean of the refresh dates they represent (note, one plan is expanded in the graph to show the actual two refresh dates it includes). In Figure 1, the optimum plan would be the lowest data point in the vertical direction. This data point has the lowest total life cycle cost and represents the greatest cost avoidance combination of design refreshes and bridge buys.

The V-22 Osprey Advanced Mission Computer (AMC)

Figure 2 – The V-22 Osprey.

The V-22 Osprey is a tilt-rotor helicopter with vertical take off and landing capabilities originally designed jointly by Bell Helicopter Textron and Boeing Helicopters, Figure 2. The V-22 has been in development and production since the late 1980's. Because of the V-22's long-term development and production schedule, the system as a whole has faced and will continue to face the problem of part obsolescence, since many of the electronic components in the system are no longer manufactured.

The analysis in this report focuses on the V-22 Advanced Mission Computer (AMC). The AMC is used for mission computing, navigation, targeting and onboard data processing. An early predecessor of the AMC was the AN/AYK-14(V), shown in Figure 3. It consists of standard plug-compatible modules and multiple chassis types. It can be configured and designed to meet individual user requirements, and its off-the-shelf microelectronics technology building block approach allows for a variety of technology insertions and permits the system to keep pace with evolving processing [2].

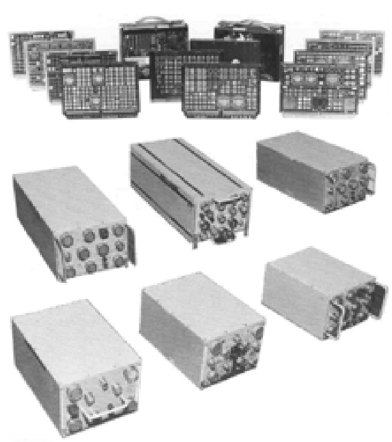


Figure 3 – The AYK-14, [2].

Combining Refresh Planning and Technology Insertion Analysis

Because of the scope and timeframe of the V-22 system, NAVAIR has developed a technology roadmap that articulates goals and targets for the Osprey's production and development. Technology roadmaps usually contain information on big-picture budget cycles and technology goals, and these types of constraints need to be included within the MOCA refresh planning analysis. Without inclusion of the roadmapping information, MOCA's obsolescence strategies are developed from the bill of materials, production plans and part obsolescence dates alone, ignoring constraints placed on the management of the AMC by technology insertion and upgrade plans.

Roadmaps differ in the types of information they can contain, but the information concerning obsolescence and sustainment can be grouped into the following four categories:

- Timeline Events (Exclusive) – This category includes budget cycles and schedule constraint events. These events eliminate particular MOCA generated refresh plans from consideration if they contain refreshes during periods when no refreshes are allowed to take place.
- Timeline Events (Inclusive) – These events require a specific action during a specific time period, and force MOCA generated plans to contain one or more refresh events in specified time periods.
- Costs – Roadmaps often dictate the specific actions that must take place at a refresh, which will in turn increase or decrease the cost of events in MOCA's timeline.
- Individual Parts – Roadmaps may also reference specific parts that need to be phased out or introduced at specific times.

These types of events must be included within the MOCA analysis along with BOM, part costs, part obsolescence dates, production dates, etc., in order to attain more viable refresh solutions that better reflect an actual implementable design refresh strategy.[1]

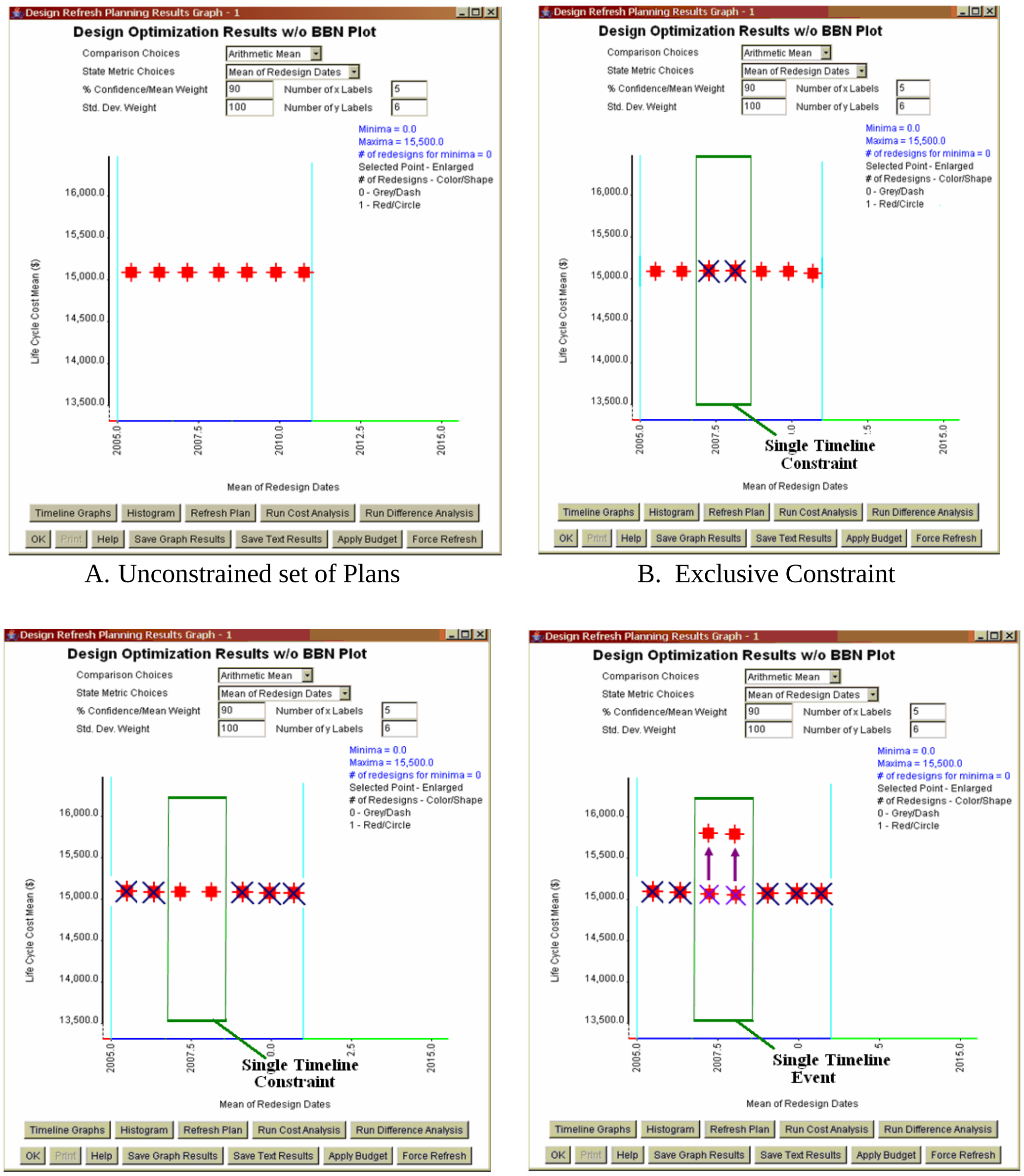
When designing an information model to represent data gleaned from technology roadmaps, one must consider the fact that roadmaps exist in a wide range of forms and contain a variety of data types. Keeping the information model as broad as possible will allow the users to interpret the roadmap as they see fit, and then express the data in acceptable terms. A redesign roadmap constraint parameter is defined to allow the concurrent accommodation of multiple roadmap constraints.

All roadmapping constraints can, at their very broadest, be considered as timeline events. In this sense, all constraints included in the information model should have a start date and an end date, since constraints will be applied over a period of time. These constraints can be interpreted as either exclusive events, where only plans that have no events in the time period are considered, or inclusive events, where only plans that have events that contain some action within the time period are considered. Note, inclusive events do not preclude plans that contain actions outside of the constrained period, they only required that the plan include some action within the inclusion period. Additionally, the information model must also accept a cost constraint along with the timeline start and stop dates. This cost constraint can be used to adjust the life cycle costs of plans that are affected by the timeline constraints. That is, the cost constraint will either increase or decrease the total life cycle cost of a given plan or set of plans. Finally, any information model with timeline and cost constraints should also include a list of affected parts (or groups of parts) that will be either redesigned out of the system or last time bought because of a given timeline constraint.

Figure 4 shows how roadmapping constraints are applied to candidate refresh plans. Figure 4A shows a set of candidate refresh plans before any roadmapping constraints have been applied, i.e., no plans have been eliminated ('x'-ed out) from consideration. This represents a very simple MOCA output graph. Each plotted point on the graph represents a single design refresh plan that dictates when lifetime/bridge buys and design refresh events should be performed. In this illustrative example, every refresh plan costs the same amount to implement; hence every point on the graph has the same y-axis value. The other three graphics in Figure 4 (B, C, and D) show how the set of possible solutions is changed by the addition of technology roadmap constraints. In Figure 4B a single exclusive timeline event has been applied, and any design refresh plans within the event's start date and end date have been eliminated from consideration and marked with an 'x'. This case models a blackout period, where no design refreshes can take place, possibly because of budget or personnel limitations. Similarly, if the timeline event was considered inclusive, as in Figure 4C, only plans falling within the timeline constraint's start date and end date will be considered; all other plans are eliminated. This models a scenario where a design refresh must take place within a certain time period, similar to a planned design refresh where new technology is inserted or an operating system upgrade must be performed. Finally, Figure 4D shows an inclusive timeline constraint with an additional

[1] In the context of this report, "part" is used generally to refer to hardware and software, i.e., a part could be an electronic chip or a piece of application software.

cost constraint added to the viable plans. In this case, only plans included within the timeline event are considered and a cost has been added to each. Like the purely inclusive case shown in Figure 4C, this models a planned design refresh scenario, and the added costs in Figure 4D exemplifies a case where there are additional costs associated with this planned design refresh.



A. Unconstrained set of Plans

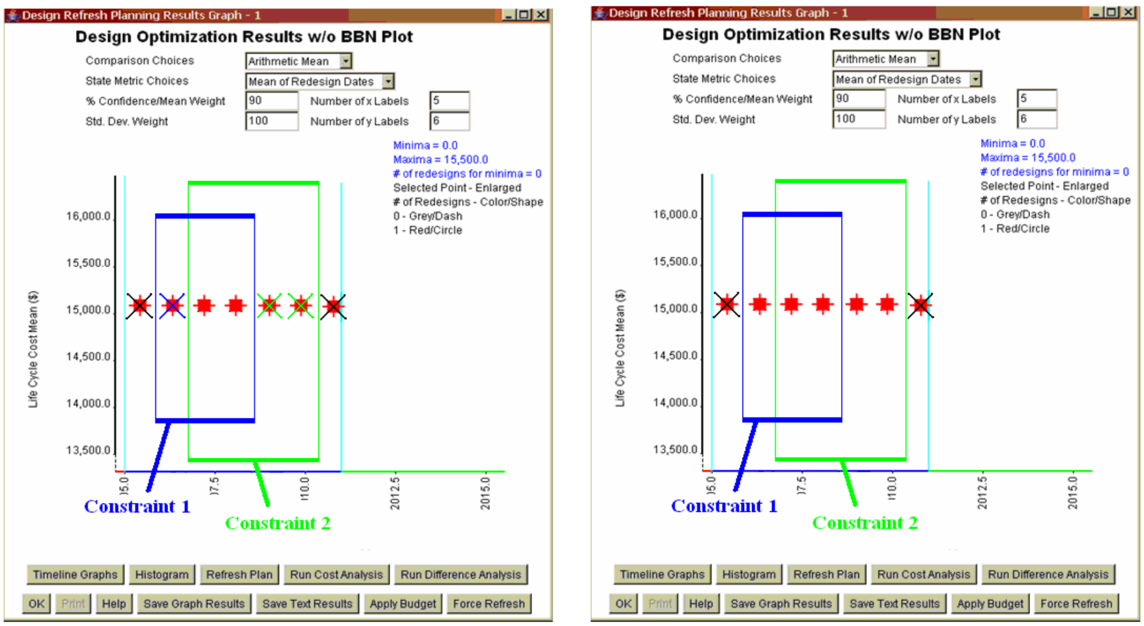
B. Exclusive Constraint

C. Inclusive Constraint

D. Added Costs Associated with Events

Figure 4 – Sample roadmapping constraints affecting MOCA generated refresh plans.

Additionally, a method for applying multiple constraints must be devised, i.e., more than one constraint could be active in a given time period. There are two ways that multiple constraints could affect possible solutions: 1) any viable plan must meet all of the provided constraints (constraint 1 AND constraint 2), and, 2) any viable plan must only meet only one of the provided constraints (constraint 1 OR constraint 2). Any information model describing technology roadmaps must be able to re-create either scenario. An example of how the results would differ because of the scenario chosen is shown in Figure 5. In Figure 5, two separate inclusive constraints have been applied to the sample output. Case 1 shows an example where only plans meeting both constraints are considered viable. This is similar to a situation where a single board or system is used on two different platforms, and each platform is scheduled to be design refreshed at a different time. In Case 1, the planned design refresh of each platform is modeled as a separate timeline event, and the place in time where these two plans overlap represents the only possible time when this system could be refreshed. Case 2 shows a case where plans are accepted if they meet either one of the constraints. This approach to dealing with multiple constraints is useful if there are two periods of time that a design refresh could occur, as in two different budget cycle periods. Allowing users to model scenarios with multiple roadmapping constraints is a way to ensure that all types of roadmap information can be included into a design refresh analysis, and allowing either an 'AND' relationship or an 'OR' relationship between the constraints ensures that constraints are dealt with properly.



Case 1: AND Relationship Case 2: OR Relationship

Figure 5 – Applying roadmap constraints.

V-22 AMC Refresh Planning and Technology Insertion Analysis

The AMC was modeled in the MOCA design refresh planning tool as a set of daughter and parent boards with four levels of hierarchy. The loaded system consists of 303 total parts distributed on 12 boards. The hierarchy of the boards is shown in Figure 6. In addition to the hierarchy, 177 discrete production events, 165 unique obsolescence dates, and the cost of the system were also inputted to MOCA. From these inputs, MOCA is capable of generating a strategic obsolescence plan consisting of bridge buys, lifetime buys, and design refresh events, which, if followed, will result in cost avoidance when compared to a purely reactive obsolescence strategy.

A version of MOCA extended to include roadmapping constraints has been applied to the V-22 Advanced Mission Computer, allowing the NAVAIR roadmap to influence MOCA's refresh planning output. Figure 7 shows a sample technology roadmap depicting one functional area of the V-22 system. This roadmap will be used to demonstrate refresh planning with concurrent obsolescence management and roadmap constraints, and will show how roadmaps should be interpreted when being input into refresh planning analysis.

The roadmap in Figure 7 depicts four subsystems, each of which NAVAIR has scheduled for replacement or design refresh. In the figure, the bars show systems that should be replaced because better technology is anticipated to become available. This roadmap represents NAVAIR's proactive plan for aging systems, which differs from the obsolescence information used by MOCA. The obsolescence dates used in the MOCA analysis indicate when a part will no longer be available, while NAVAIR's roadmap is

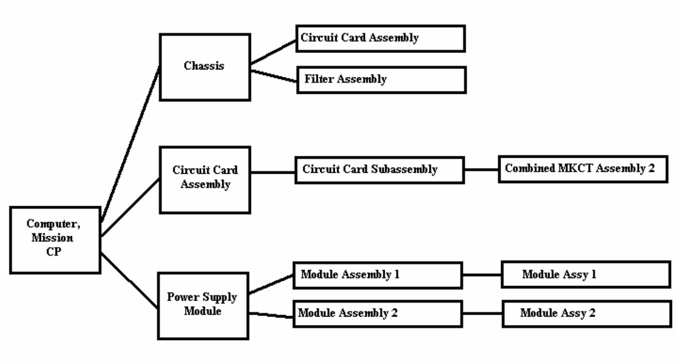
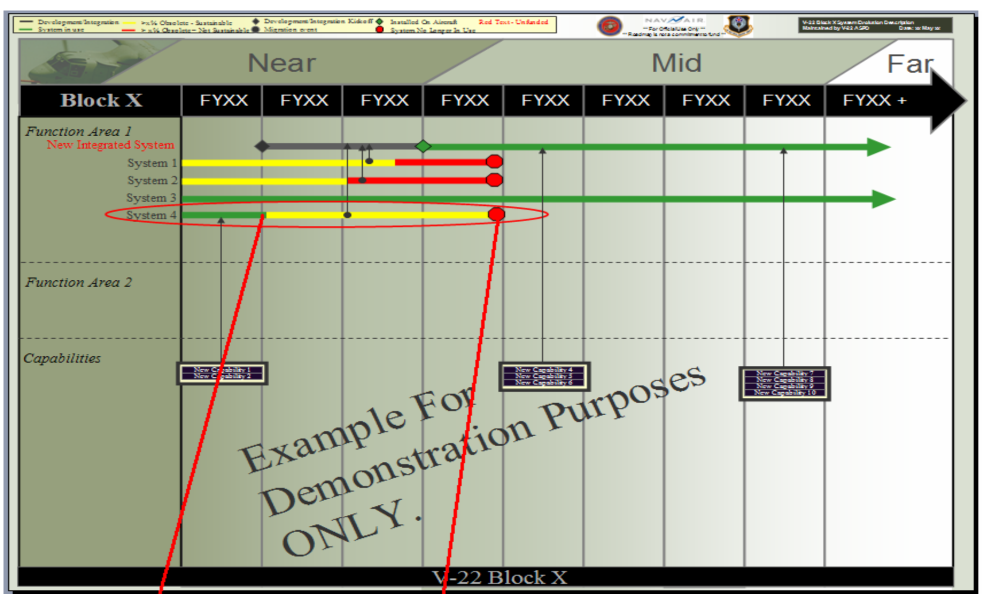


Figure 6 – Hierarchy of the AMC as loaded into MOCA.

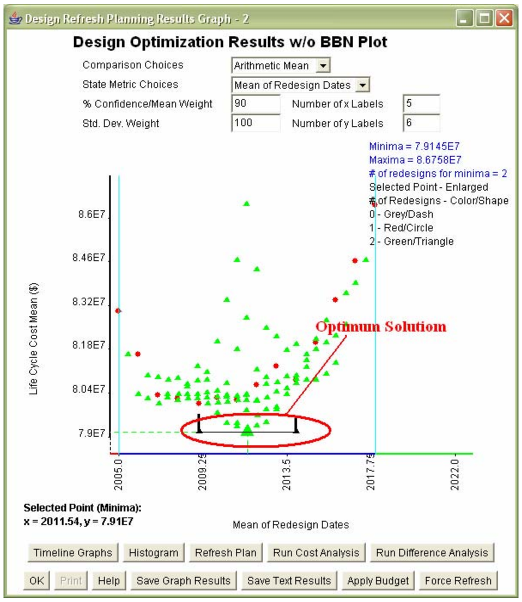
concerned with keeping the system up to date and capable of performing a required and potentially evolving function, even though the system may be sustainable after their given refresh deadline. The subsystems shown in Figure 7 are all scheduled to be refreshed within the same 2 year period, assumed in the example MOCA analysis to be 2008 and 2010.



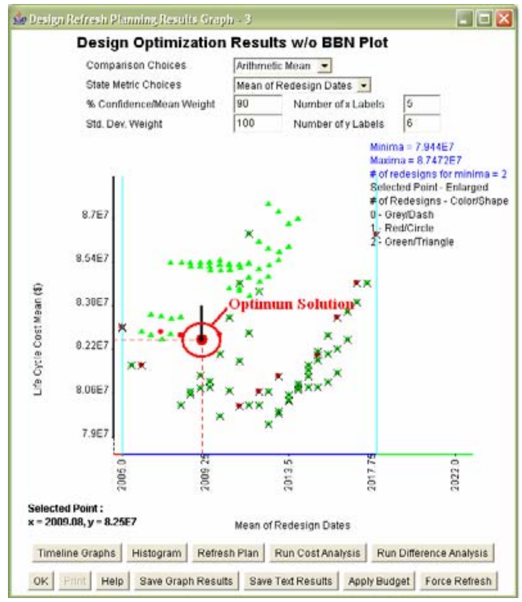
Constraint Start Date	Constraint End Date	Constraint Cost	Consider Multiple Constraints Separately?
2007	2010	$2.5M	Yes

Figure 7 – Sample V-22 roadmap.

Because of the roadmap criteria, MOCA only considered refresh plans with a design refresh between 2008 and 2010. An additional cost of $2.5 million was added to design refreshes between 2007-2010.



Solution without Roadmap Constraints
(Refreshes at 2009, 2014)



Solution with Roadmap Constraints
(Refresh at 2009)

Figure 8 – MOCA results with and without roadmap constraints applied.

Figure 8 shows the MOCA simulation outputs for the V-22 AMC before and after the roadmap constraints have been applied. The figure on the left of Figure 8 shows a basic MOCA output without roadmap constraints, while the figure on the right shows the same results with the roadmap constraints applied. The refresh plans that do not satisfy the roadmapping constraints are crossed-out in the graph on the right and one can see that all the viable refresh plans (plans that satisfy the roadmapping constraints) have been shifted upwards in the graph on the right. This is because of the additional cost constraint that was applied to all design refresh plans with a design refresh between 2007 and 2010. The graph on the right in Figure 8 shows that the optimum refresh plan, changes because of the roadmap constraints, and shifts from a solution with two refresh dates (2009, 2014) to a solution with a single refresh date (2009). This is not always the case, sometimes adding roadmap constraints does not change the refresh dates associated with the optimum solution. However, this case study shows that adding time and cost constraints to the MOCA simulation may actually change the final recommended refresh plan.

Business Case Analysis

In order for the refresh planning predictions to be useful, the impact of the plans must be articulated as a business case. The V-22 AMC analysis considers two primary alternatives when evaluating design refresh plans: the option to lifetime buy parts when they go obsolete, or the option to bridge buy parts when they go obsolete and later refresh the design so that the obsolete part is no longer needed. MOCA seeks to optimize the number of design refreshes and lifetime buys with respect to cost, and the final plan will have a combination of lifetime buys and bridge buy/design refresh events. In order to evaluate the utility of the optimum plan it is compared to the a “perfect world scenario” where no parts go obsolete, a purely reactive strategy where all obsolete parts are lifetime bought and stored, and a strategy where every obsolescence event is solved with a design refresh. These four scenarios (perfect world, no refresh, all refresh, and optimum) are compared by breaking down the total cost of obsolescence management into sub-costs to identify where the money is being spent, and by deriving an obsolescence cost ratio to measure the relative costs of each of the obsolescence strategies.

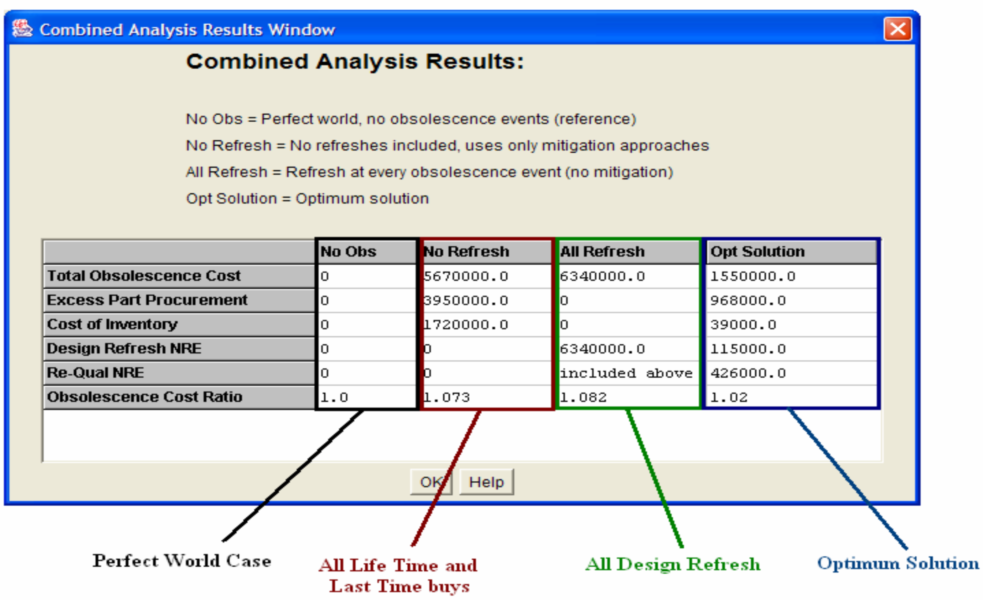
There is assumed to be no obsolescence management cost in the perfect world scenario, since this scenario assumes that no part will ever become obsolete. This perfect world case is a baseline for comparing the other obsolescence mitigation strategies, including MOCA’s optimum plan. One can determine the true cost of obsolescence management for a given strategy by taking the total cost of the plan and subtracting from it the cost of the perfect world scenario,

$$O_c = T_A - T_{LCP} \tag{1}$$

where
O_C = obsolescence management cost
T_A = actual total life cycle cost of the system with the selected obsolescence management approach
T_{LCP} = total life cycle cost in the perfect world scenario.

The T_A includes all costs associated with procuring parts and building the system, all costs associated with design refresh and re-qualification costs, all lifetime buy and bridge buy costs, as well as all inventory costs for storing parts. The T_{LCP} includes only those costs that are not associated with obsolescence, and simply includes the recurring costs of building the system and procuring the parts. Thus by subtracting the T_{LCP} from the T_A the obsolescence management cost can be obtained.

MOCA then breaks down this obsolescence management cost into the sub-costs associated with the excess part procurement (the difference between part procurement costs if there was no obsolescence and part procurement costs associated with lifetime and bridge buys of obsolete parts) as well as the inventory cost (cost of storing the parts over the long term). The obsolescence management cost also includes any costs associated with the redesign and re-qualification and any other costs associated with a design refresh. All the obsolescence management costs include cost of money (they are Net Present Value quantities indexed to the analysis starting year) and include the effects of the budgeting period duration. A sample output from MOCA's business case analysis is shown in Figure 9.



	No Obs	No Refresh	All Refresh	Opt Solution
Total Obsolescence Cost	0	5670000.0	6340000.0	1550000.0
Excess Part Procurement	0	3950000.0	0	968000.0
Cost of Inventory	0	1720000.0	0	39000.0
Design Refresh NRE	0	0	6340000.0	115000.0
Re-Qual NRE	0	0	included above	426000.0
Obsolescence Cost Ratio	1.0	1.073	1.082	1.02

Figure 9 – Sample business case comparison output.

The sample output in Figure 9 represents the business case analysis of the optimum solution for the NAVAIR AMC Mission Computer.[2] From Figure 9 one can ascertain that there are no costs associated with obsolescence in the perfect world (i.e., the first column) – the cost of all other cases are measured relative to the perfect world. When interpreting Figure 9, it is important to realize that the first row (Total Obsolescence Cost) is a sum of the four rows below it, which make up the constituent parts of the total

[2] Note, the total obsolescence cost of the optimum solution in Figure 9 is not the same as the total life cycle cost of the optimum solution point in Figure 8. This is because the total obsolescence cost is only a portion of the total life cycle cost. Their relationship is detailed in (1).

obsolescence cost. The no refresh reactive case (second column) portrays a strategy where all money spent on obsolescence either goes toward mitigation (in this case lifetime buys) and no money is budgeted for design refresh or re-qualification. The other extreme of this is the third column (All Refresh), where all money is spent on design refreshes, i.e., every obsolescence event is solved via a design refresh and no lifetime or bridge buy parts are bought and stored. The optimum solution (fourth column) represents a combination of lifetime buys (mitigation), and bridge buys coupled with design refreshes.

In order to compare these four scenarios, the total obsolescence management cost from (1) can be used, however, the difference in (1) is not independent of the year that the money is indexed to, i.e., it can lead to misleading results as the discount rate and year money is indexed to are varied. Alternatively, the obsolescence cost ratio is calculated by dividing the actual total life cycle cost by the perfect world cost where no parts become obsolete,

$$O_{CR} = \frac{T_A}{T_{LCP}} \tag{2}$$

The obsolescence cost ratio is independent of the year money is indexed to and therefore represents a better comparison metric between strategies.

This cost ratio is a way of comparing the relative cost of the optimum plan with respect to the extreme scenarios as well as the perfect world. Figure 9 shows that the optimum solution has an obsolescence cost ratio of 1.02, as compared to the more extreme cases that have obsolescence cost ratios of 1.073 and 1.082. The perfect world scenario has an obsolescence cost ratio of 1.0, since the actual total life cycle cost is equal to the total life cycle cost if no parts had gone obsolete in this case.

Cost measures are only one facet of building a business case. Another key piece of information needed is the confidence that you have in the predicted cost (or cost avoidance). We need to be able to generate the confidence level for the predicted amount of cost avoidance between any two plans, or any two obsolescence management scenarios, i.e., a confidence interval to quantifying the level of certainty there is that one plan is better than another. This is illustrated in Figure 10, where the difference in total cost is compared for the optimum solution (the circled triangle) and the solution where only lifetime buys have been made (the circled dash). It should be noted that the optimum solution contains two scheduled design refreshes (2009 and 2014). In order to produce the confidence interval, a Monte Carlo analysis must be performed. In the V-22 AMC case study, the obsolescence dates and parts costs were assumed to be uncertain. They were modeled as symmetric triangular probability distributions with variations of ±10% for costs and ±1 year for dates. The input values to the refresh cost model were sampled, and using the sampled set of parameters, the difference between the selected plans is calculated. Performing this difference analysis 1,000 times and plotting the results as a histogram produces the plot on the right side of Figure 10 represents the life cycle cost difference between the two plans once uncertainty has been applied to the inputted data. The histogram shows the probability of the lifecycle cost of the optimum plan being less than the reactive solution (all lifetime buy). That is, the histogram represents the difference between the optimum and reactive plans, as if the optimum plan was subtracted from the reactive solution. One can see from the histogram that the mean difference between the two plans is $3.3 million. More importantly however, is where the '$0' cost avoided line falls on the histogram. The area under the histogram to the left of $0 is the probability that the optimum solution will end up costing more than the reactive solution. In the case of the NAVAIR AMC study, this portion of the histogram is very small, so there is very little chance that the optimum solution will cost more than a purely reactive solution. Finally, the histogram shows how much confidence there is that choosing the optimum plan will have cost avoidance. For example, there is a 50% confidence that choosing the optimum plan will result in a cost avoidance of $3.3 million or more, while there is an 84% of a cost avoidance of $2.2 million or more.

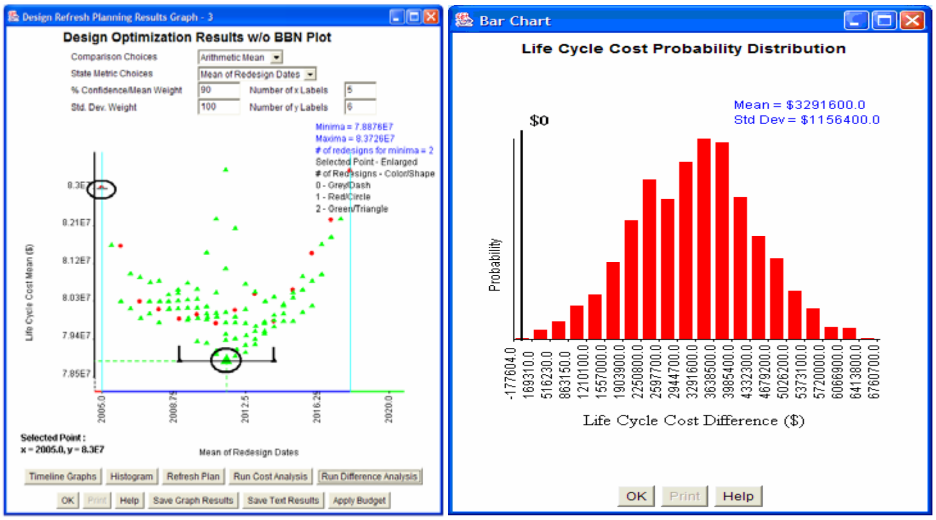


Figure 10 – Sample Monte Carlo confidence for the difference in cost between two selected plans.

Summary and Conclusions

This report describes the development of an information model that details technology roadmapping information for use within design refresh planning, and also describes a business case analysis of the resulting refresh plans.[3] These advancements result in a better obsolescence analysis because they allow the user's own obsolescence strategy and organizational forecasts to affect the final obsolescence strategy that refresh planning develops. Additionally, the business case model presents a way for the user to evaluate the chosen solution in terms of cost avoidance, thus allowing the user to quantify the value of the suggested strategy. The business case model allows the user to appraise the risk associated with a given plan by demonstrating the uncertainty linked with the plan's projected cost avoidance. The addition of a business case analysis and a technology roadmapping information model present a more realistic obsolescence management strategy and a way of evaluating the value of the chosen strategy, the end result of which is a better, more accurate solution.

References

1. P. Singh and P. Sandborn, "Obsolescence Driven Design Refresh Planning for Sustainment-Dominated Systems," *The Engineering Economist*, Vol. 51, No. 2, pp. 115-139, April-June 2006.
2. "AN/AYK-14 (V)". General Dynamics Advanced Information Systems. http://www.gd-ais.com/Capabilities/offerings/sr/Processors/ayk141.htm

[3] One observation from the analysis in this paper is, however, that many technology roadmaps do not contain sufficient detail to enable constraints to be determined at a level required for the type of analysis demonstrated in this paper.