Acta Manilana 70 (2022), 01-12

ISSN: 0065-1370

Acta Manilana

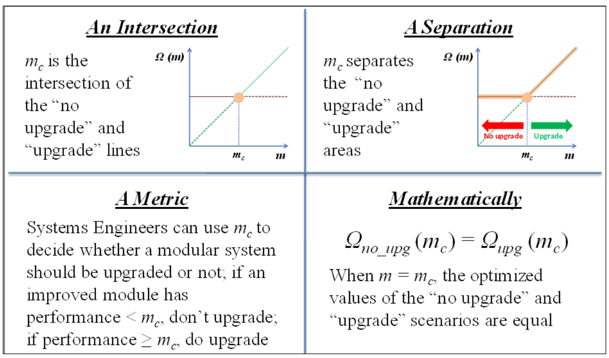
journal homepage: https://actamanilana.ust.edu.ph/

A Parametric Analysis of Modular System Upgrades using Search Optimization Techniques

Romulo Francis J. Broas[1]* & Mo Mansouri[1]

[1]*Stevens Institute of Technology, Hoboken, NJ 07030, USA*

Graphical Abstract



Abstract

Search optimization techniques were used to explore which new innovation or technology to implement in a modular system upgrade amid many options. The model incorporates the expectation of future technologies or innovations, and the time-dependence value of module performance characteristics. A critical module performance characteristic m_c is explored and calculated, and this will be the reference parameter the Systems Engineer will use when deciding to upgrade a modular system.

Keywords: modularity, upgrades, optimization

I. Introduction

Modularity describes the relationships among parts that make up the system called modules. It is "the understanding of systems as the combination of separated components" [1]. The primary concept is the "interdependence within and independence across modules." The module, which is the building block that make up modular systems, "is a unit whose structural elements are powerfully connected among themselves and relatively weakly connected to elements in other units" [2].

Corresponding author: rombro97@gmail.com
DOI: https://doi.org/10.53603/actamanil.70.2022.zpsn3824

©2022 The Authors. Published by the Research Center for the Natural and Applied Sciences, University of Santo Tomas Manila, The Philippines

The modules can be independent of one another in many ways such as design (form, fit, function) and manufacturability, but when connected together, they function cohesively as a modular system. Modules are "composed of features that act together in performing some discrete function that is semi-autonomous in relation to others" [1]. Hence, a system designed to be modular must achieve a delicate balance where it allows modules to be relatively independent of another but also coupling them minimally so that they can be connected together.

In modular systems, interfaces between modules are designed so that modules are independent of each other but can also be linked together to become the system. Interfaces are connections of modules to other modules in a system so that their "mating" achieves the physical and functional requirements of the system. It is through interfaces that modules have the following characteristics: 1) they can be separated from other modules and even the system itself, 2) they can be isolated as distinct, "self-contained, semi-autonomous chunks," and 3) they possess the capability to re-combine with other modules to make up the system [3].

The functional requirements of the modules that are connected to each other through interfaces are paramount in modular systems. A module must perform particular tasks (functions) that will support the overall system performance and requirements. However, it must be noted that in a modular system, exactly how a module accomplishes this task is decoupled from other modules and the system. As long as a module performs its tasks and adheres to the interface protocols to link with other modules, it is irrelevant what is inside the module and how it is designed. In this way, modules are independent of each other in a modular system, and they "can be designed, produced, and tested separately" as long as they maintain the same interfaces to be linked to other modules [4]. Hence, a key attribute of modular systems is how change can be localized to particular areas with minimal impact to other modules.

Among the benefits of modularity are: 1) less complexity through standardization, 2) improvement of remove, replace and/or repair through plug-and-play interfaces, 3) possibilities for re-use across different platforms, 4) reduction of cost through bulk pricing, and 5) minimization of design, test, and manufacturing cycle times.

This paper exploits the interchangeability characteristic of modular systems where a particular module can be upgraded with minimal impact to other modules. A search optimization technique is used to explore which new innovation or technology to implement in a modular system upgrade amid many options. Within the context of modular systems, a model is analyzed to examine how to choose new technologies or innovations so that system upgrades are optimized.

Outcomes of the research are insights on how to do better modular upgrades, which will be the basis of a decision-making tool that should enhance the systems engineering practice of modularity.

The organization of this paper is as follows: Section I reviews the current practice of modularity and states the motivation of the research. Section II discusses the search optimization technique and introduces the critical module performance parameter m_c.

Section III derives the formula for m_c so that it can be calculated. Section IV discusses the results of various calculations of m_c and presents a parametric analysis showing the sensitivity of m_c to various factors. In Appendix A, the formulation of time dependence is derived to show greater clarity to the theory introduced in Section II.

II. Applying the Search Optimization Technique

This section discusses the application of the search optimization technique to aid Systems Engineers in upgrading their modular systems. The importance of the parameter m_c is discussed, and the theory that will be used in its calculation is developed.

In modular systems, the interchangeability characteristic of modules allows for different permutations for which to upgrade the system. Having discussed the properties of modular systems and the dynamics of modular system upgrades, this paper discusses the application of search optimization techniques developed previously in the field of Economics Research to address the optimal conditions in which to upgrade a modular system [5, 6, 7, 8, 9]. In a preliminary model where search optimization was applied to a modular system upgrade, it was seen that the optimization technique is a hybrid of the "no upgrade" and "upgrade" approaches [10, 11, 12].

Central to this model is the critical performance characteristic of a module, denoted by m_c. It is used by the Systems Engineer when deciding to upgrade a modular system. If an improved module with characteristic m is available such that $m < m_c$, then the Systems Engineer should not upgrade and instead continue to receive the performance characteristic associated with the module already in place. If an improved module with characteristic m is available such that $m \geq m_c$, then the Systems Engineer should pursue the upgrade and thus yield the corresponding improvement in the system stemming from the newly-incorporated module m.

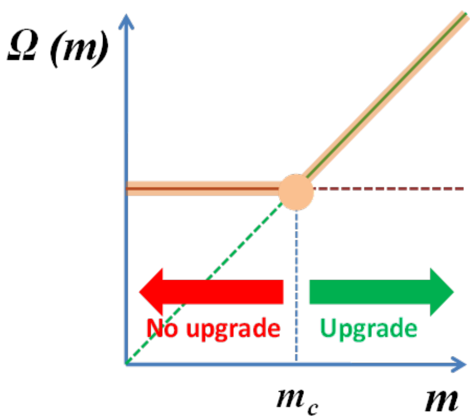


Figure 1. The optimization strategy showing the critical value m_c.

If a module has incorporated a new technology or innovation and therefore an opportunity emerges to upgrade the modular system, the System Engineer has two options:

1. Do not use the improved module and do not upgrade the system. He continues to receive the same performance while waiting for another opportunity to upgrade the system.

2. Use the improved module and proceed to upgrade the system. Henceforth, he will receive the new performance associated with the newly-upgraded system that recently incorporated the improved module.

Let *Ω (m)* be the optimization strategy that is dependent on a module characteristic m. The function *Ω (m)* is the maximum of the "no upgrade" and "upgrade" module performance characteristics. Mathematically, *Ω (m)* = maximum { "no upgrade", "upgrade" }.

II. A. The "No Upgrade" Scenario

When the Systems Engineer decides against upgrading the system, he will continue to receive the default system performance while anticipating another opportunity to upgrade in the future. This expectation of future upgrades is mathematically denoted by the Expected Value *EV* [13]. If *P(m)* is the probability distribution of the modular performance characteristic *m*, then the Expected Value *EV* is the weighted average (mean) of future module performance characteristics based on anticipated technologies and innovations for modular upgrades [14].

$$EV = \sum \Omega(m) \times P(m)$$

If a major breakthrough is about to happen that will improve the system dramatically, the probability of an excellent modular performance will be high, and hence the *EV* of the future module performance characteristics will be high. However, if mediocre improvements to the modules are expected, then the *EV* of future module performance characteristics will be low. For example, if a 3 dB antenna will be upgraded so that there is a 100% chance it can improve its gain to 6 dB, then the *EV* is 6 dB (= 100% x 6 dB).

Therefore, the optimized performance value that the Systems Engineer will receive should he decide against upgrading the system is the default module performance characteristic he is currently receiving (denoted by m_d), plus the value of anticipated module performance values stemming from future upgrades. Mathematically, this is denoted below.

$$\Omega_{no\_upg}(m) = m_d + D \times EV$$

The time-dependence of module performance values used in the optimization strategy is discussed in Appendix A. The Expected Value *EV* is multiplied by the rate *D* to account for time-dependence, since *EV* is essentially a module performance value, albeit a future one.

II. B. The "Upgrade" Scenario

As mentioned previously, if the Systems Engineer decides to upgrade his module, he will yield the corresponding improvement in performance associated with that upgraded module. The time-dependence value of this improvement, expressed as a geometric gradient (percentage), is shown in Appendix A over the entire system lifetime. Hence, the optimized performance value that the Systems Engineer will receive if he upgrades the module is the sum of that upgraded module's performance characteristic plus the value of that upgrade during its entire system lifetime.

$$\Omega_{upg}(m) = m + D^1 \times m + D^2 \times m + \cdots + D^n \times m$$
$$\Omega_{upg}(m) = m \times (1 + D^1 + D^2 + \cdots + D^n)$$
$$\Omega_{upg}(m) = \frac{m}{1-D}$$

Note that both the *EV* of the non-upgraded system, and the upgraded module performance characteristic *m* of the upgraded system, are multiplied by the time factor *D* to account for the progression of time and the time-dependent changes of module performance values.

Hence, as the maximum of the "no upgrade" and "upgrade" module performances, the optimal strategy is written mathematically as the following:

$$\Omega\ (m) = \text{maximum}\ \{\text{"no upgrade", "upgrade"}\}$$
$$\Omega\ (m) = \text{maximum}\ \{m_d + D \times EV,\ m\ /\ (1 - D)\}$$

III. Calculating the Critical Module Performance Characteristic, m_c

When $\Omega_{no\_upg}(m) = \Omega_{upg}(m)$, the parameter m is the critical module performance characteristic m_c (i.e., $m = m_c$).

$$\Omega_{no\_upg}(m_c) = \Omega_{upg}(m_c)$$
$$m_d + D \times EV = \frac{m_c}{1-D}$$

Per Figure 1, if $m < m_c$, then the Systems Engineer should not upgrade. If $m \geq mc$, then the Systems Engineer should pursue the upgrade. So, the optimal strategy becomes the following:

$$\Omega(m) = \begin{cases} \dfrac{m_c}{1-D}, & for\ m < m_c\ (no\ upgrade) \\ \dfrac{m}{1-D}, & for\ m \geq m_c\ (upgrade) \end{cases}$$

The critical module performance characteristic m_c, which will determine if the Systems Engineer should upgrade or not, is solved algebraically to be the following:

$$m_c = (m_d + D \times EV)(1 - D)$$

The goal is to calculate m_c, but m_c has the Expected Value *EV* term which in turn is dependent on the optimization function *Ω (m)*. Finally, *Ω (m)* itself is dependent on m_c, and thus the equation for m_c as shown above is a recursive function that is difficult to solve explicitly. Rather, the equation for m_c as shown above needs to be solved numerically using computer programming codes.

IV. A Parameteric Analysis of Expected Future Module Performance

This section analyzes the results of various calculations of m_c. A parametric analysis centered on the sensitivity of m_c to various factors is discussed.

A parametric analysis was performed to understand the sensitivity of the critical module performance characteristic m_c to the prospects of future module improvements. In particular, two sets of data were calculated that centered on the anticipated modular performance. First, the anticipated modular performance *m* was varied from low expectations ($m = 4$) to higher expectations ($m = 7$). This examined the effect of expecting small improvements versus big improvements.

The second set of parametric analysis focused on the Systems Engineer's degree of confidence that a particular modular performance will occur. Calculations were performed on a Systems Engineer that is very confident that particular modular performance will be available for upgrade (low σ = 0.01) versus one that is not confident (high σ = 5). Insights on how the Systems Engineer should perform modular system upgrades will be discussed under these scenarios.

Consider the following scenario and its corresponding initial values:

- A system has a default module with a performance parameter of 3 ($m_d = 3$). This could be an antenna that is part of a radar system that has 3 dB of Gain.
- The effect of time is assumed to be at an appreciation rate of 80% ($D = 0.8$). Therefore, as discussed in the Appendix, the system value of a particular modular performance characteristic increases by 80% after each time unit.

With his eye on his company's Research & Development (R&D) Department, the Systems Engineer expects improvements to the module currently in the system. For example, the Systems Engineer could be anticipating an upgrade to a Gain of 4 dB at 90% probability. In another scenario, he is anticipating an even better upgrade of 5 dB Gain at 90% probability, and so forth.

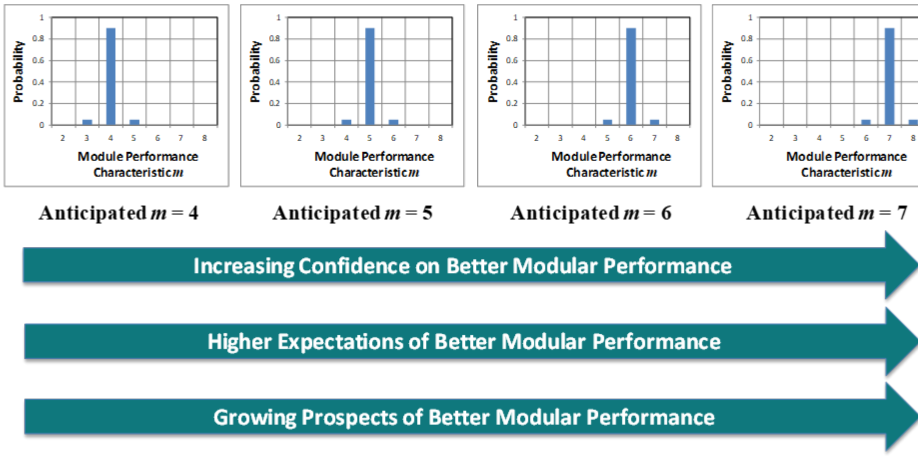


Figure 2. Varying the Module Performance Characteristic Expectation.

In Figure 2, the increase in anticipated modular performance because of an upgrade is shown. As the Systems Engineer increases his expectation of an upgraded module with 4 dB, 5 dB, 6 dB, and so on until 8 dB of Gain, the gradual increase in upgrade performance coincides with increasing confidence, higher expectations, and growing prospects of better modular performance.

In Figure 3, the System Engineer's optimal strategy is shown regarding whether he should upgrade his system or not based on what he is expecting regarding future module performances. Recall that the current system has a performance value of 3 ($m_d = 3$). In the data point of $m = 4$, he is 90% confident that a performance value of 4 will occur, a 5% chance that better than 4 will be available, and a 5% chance that no significant performance upgrade will happen. Then, the calculated critical modular performance characteristic of 3.83 ($m_c = 3.83$) dictates that he should not pursue any upgrade with modular performance less than 3.83. Hence, upgrading his system from a module performance of 3 to 3.83 is not optimal because there is a 95% chance of a module performance upgrade of 4 or above. However, any upgrade above 3.83 should be pursued.

Consider the expectation that there will be a module upgrade available with a performance value of 5. This expectation is quantified as having a 90% chance that a module with performance of 5 will occur. There is a 5% chance that an improvement of less than 5 will occur, and a 5% chance that an even better performance value greater than 5 will occur. In this particular scenario, m_c is calculated to be 4.625. Therefore, even if there is a 95% chance that module performance upgrades of 5 or better will be available, the optimal strategy is to accept all upgrades better than 4.625 since there is that 5% chance that improvements will be less than the modular performance of 5.

Note that after an anticipated module performance of 4, there is a positive slope linear relationship between the critical module performance parameter m_c and the anticipated module performance. Namely, as the Systems Engineer expects better module upgrades, so does his benchmark of when to do an upgrade likewise increases.

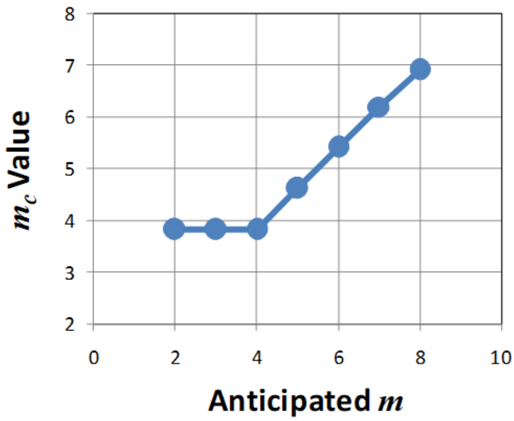


Figure 3. m_c value versus anticipated module performance.

Furthermore, it is rightly so that his metric to do an upgrade (m_c) is never above what is expecting a future upgrade to be. Rather, m_c is always slightly lower than the module performance he is anticipating because there is a 5% chance that improvements will be less than what he expects. Therefore, higher expectations mean higher thresholds for upgrading (increasing m_c), and an aversion to upgrade.

In the prior parametric analysis, the Systems Engineer had 90% certainty on various anticipated modular performances. But it is worth analyzing the cases where he had varying degrees of certainty. In Figure 4, varying the standard deviation of a Gaussian Distribution is proposed as a simple way to quantify uncertainty. With a tight standard deviation σ of 0.01, the Systems Engineer is very certain that a module performance upgrade of 5 will be available at 100% probability. With a standard deviation σ of 0.5, the Systems Engineer is not so certain since there is now an 11% chance with a module performance of 4, a 78% chance with a module performance of 5, and an 11% chance with a module performance of 6. As the standard deviation is increased from 1.0 to 5.0, one can see a gradual flattening of the Gaussian Distribution and the Systems Engineer is less certain about what modular performance will be available in the future.

In Figure 5, critical module performance parameter m_c is plotted as a function of future performance uncertainty as quantified by a standard deviation. There is not a linear relationship between m_c and the standard deviation σ, but it is clear that as the standard deviation σ increases, so does the critical module performance parameter m_c. In fact, the curve is S-shaped, akin to a Sigmoid function where there is a slow increase at low σ, then a steeper slope as σ increases, and then a plateau as σ gets large. When the Systems Engineer is very certain that a module improvement of 5 is forthcoming, then his strategy is to accept any upgrades with module performances of 4.6 and above. However, when he is less certain that a module improvement of 5 will be available, he should become more hesitant about upgrading. Thus, he should increase the critical module performance parameter m_c that will trigger an upgrade. As he becomes even more uncertain about what module upgrade will be available (standard deviation > 3), his benchmark of when to upgrade starts to converge on a particular value, m_c = 5.8. The conclusion is this: as the Systems Engineer is more uncertain about future module performance upgrades (increasing σ), so shall he be hesitant to upgrade the modular system (increasing m_c).

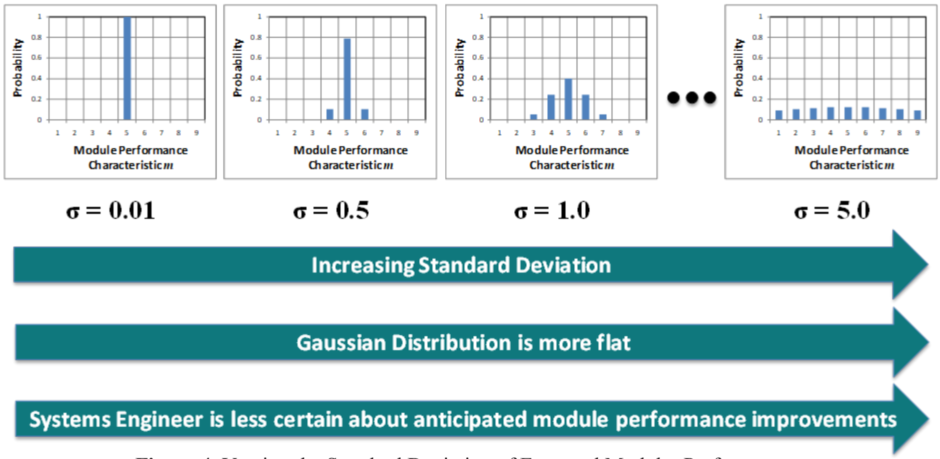


Figure 4. Varying the Standard Deviation of Expected Modular Performance.

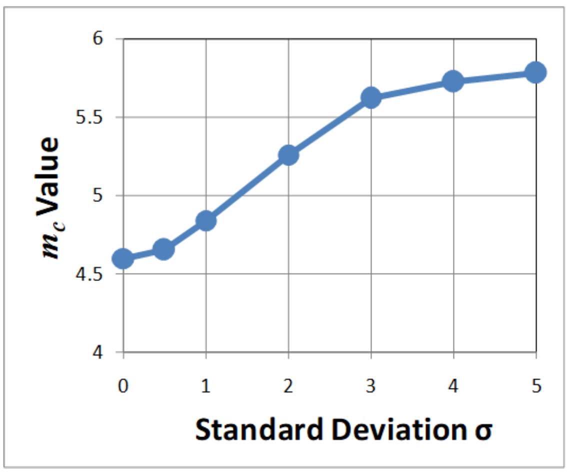


Figure 5. Varying the Module Performance Charactistic Expectation.

Conclusion

This paper refines a previously-published model that seeks to incorporate search optimization techniques in modular system upgrades. The optimization strategy of how to choose a module with a new technology or innovation to upgrade a modular system has been enhanced by the incorporation of two important factors: 1) the expectation of future technologies or innovations, and 2) the time-dependence value of module performance characteristics. The critical module performance characteristic m_c was analyzed and calculated since it is used by the Systems Engineer in determining if he should accept the new technology or innovation and upgrade.

More parametric studies will be performed. Within the context of modular systems, a framework will be proposed that will examine how to choose new technologies or innovations so that system upgrades are optimized. Outcomes of the research should be insights on how to do modular upgrades better, which will be the basis of a decision-making tool that should enhance the systems engineering practice of modularity.

Appendix A. Time-Dependence of Module Performance Characteristics

This appendix discusses in detail the time-dependent aspect of search optimization that was used in Section II.

The preliminary model proposed by Broas and Mansouri was static and non-dynamic [12]. To expand this model and add robustness, a time rate *D* is introduced to account for the progression of time and the time-dependent changes of performance characteristics and values [15]. In particular, the rate *D* allows time dependence to be included in calculations through changes in performance values over time. The change in performance values over time does not have to be constrained to a simplistic linear gradient (constant amounts or values); rather, performance values can change via geometric gradients (i.e., percentages).

Let *m* be defined as the module performance characteristic. In Table 1, the effect of time on module performance values is shown both diagrammatically and mathematically. As time progresses, the module performance value increases successively by a factor of *D*.

In Figure A1, module performance values are plotted over time using different rates *D*. All of the module performance values have the same initial value of m, but as time progresses, they begin to diverge and have different values because they each have a unique growth rate *D*. But note that all of the module performance values eventually converge on a particular value, which again is dependent on the rate *D*.

Time	*Module Performance Value (pictorial)*	*Module Performance Value (equation)*
$t = 0$	[m]	m
$t = 1$	[m] [$D \times m$]	$m + D \times m$
$t = 2$	[m] [$D \times m$] [$D^2 \times m$]	$m + D \times m + D^2 \times m$
⋮	⋮	⋮
$t = n$	[m] [$D \times m$] [$D^2 \times m$] … [$D^n \times m$]	$m + D \times m + D^2 \times m + \ldots + D^n \times m$

Table 1. Time-Dependent Module Performance Values.

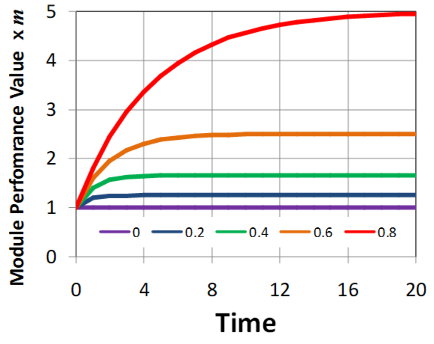


Figure A1. Time-Dependent Module Performance Values with Different Rates *D*.

The convergence of the module performance values to a particular value as time progresses can be derived mathematically. Let *n* be some measure of time that will account for time progression. Then, for a particular time *n* (i.e., $t = n$), module performance values can be expanded as follows:

$$\Omega_{upg}(m) = m + D^1 \times m + D^2 \times m + \cdots + D^n \times m$$

$$\Omega_{upg}(m) = m \times (1 + D^1 + D^2 + \cdots + D^n)$$

$$\Omega_{upg}(m) = m \times \sum_{t=0}^{n} D^t$$

If *t* is allowed go to infinity, such that the Systems Engineer is not so much concerned about his module's performance value at a particular time (i.e., $t = n$), but rather its overall impact over the entire system lifetime (i.e., $t = \infty$), then the module performance value simplifies rather nicely because of the geometrical series $\sum^{\infty}_{n=0} x^n = 1/1\text{-}x$. Hence, as $t \rightarrow \infty$, module performance characteristics approach the value having the factor $1 / 1 - D$ as shown below.

$$\Omega_{upg}(m) = m \times \sum_{t=0}^{\infty} D^t = m \times \frac{1}{1-D} = \frac{m}{1-D}$$

Acknowledgements

There are no acknowledgments.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Romulo Francis J. Broas performed the conceptualization, methodology, software, formal analysis, writing, editing, and visualization of this paper. Mo Mansouri performed the supervision, Romulo Francis J. Broas and Mo Mansouri have read and agreed to the final version of the manuscript..

References

[1] He, J. Modularity: the principle of evolution in complex systems. Saarbrücken, Germany: Lambert Academic Publishing, 2011.

[2] Baldwin, C., & Clark, K. Design Rules: The Power of Modularity. Cambridge: MIT Press, 2000.

[3] Cabigiosua, A., Zirpoli, F., & Camuffo, A. Modularity, interfaces definition and the integration of external sources of innovation in the automotive industry. Research Policy Journal, Elsevier, volume 42 (2013), pages 662 – 675.

[4] Farid, A., Hunt, F., & Moultrie, J. Modularity and Reuse. IfM position paper, March 2005.

[5] Wright, R. An Introduction to the Economics of Job Search and Frictional Unemployment. Center for Economic Research, Department of Economics, University of Minnesota, Discussion Paper No. 194, 1983.

[6] Lippman, S., & McCall, J. The Economics of Job Search: A survey, Part I (Optimal Job Search Policies). Economics Inquiry, volume XIV, 1976, pages 155 – 189.

[7] Lippman, S., & McCall, J. The Economics of Job Search: A survey, Part II (Empirical and Policy Implications of Job Search). Economics Inquiry, volume XIV, 1976, pages 347 – 368.

[8] Kornish, L. On Optimal Replacement Thresholds with Technological Expectations. Journal of Economic Theory, volume 89, 1999, pages 261 – 266.

[9] Rogerson, R. et al. Search-Theoretic Models of the Labor Market: A Survey. Journal of Economic Literature, volume XLIII, 2005, pages 959 – 988.

[10] Broas, R. F. J. An Application of Systems Thinking: Upgrading Modular Systems through the Incorporation of New Technologies or Innovations. IEEE International Symposium on Systems Engineering, Vienna, Austria, October 2020.

[11] Broas, R. F. J. Applying Search Optimization Techniques in the Incorporation of New Technologies and Innovations in Modular System Upgrades. Systems Engineering Research Center (SERC) 8th Annual Doctoral Students Forum, November 17, 2020.

[12] Broas, R. F. J., & Mansouri, M. Search Optimization Applied to a Modular System Upgrade: A Preliminary Model. IEEE 16th Annual System of Systems Engineering (SoSE) International Conference, Västerås, Sweden, June 2021.

[13] Balcer, Y., & Lippman, S. Technological Expectations and the Adoption of Improved Technology. Journal of Economic Theory, volume 34, 1984, pages 292 – 318.

[14] Larson, R., & Farber, B. Elementary Statistics, 5th Edition. New York: Prentice Hall, 2012.

[15] Park, C. S. Contemporary Engineering Economics, Fifth Edition. London: Pearson International, 2013.