



SIEMENS

Ingenuity for life

Siemens PLM Software

Modeling and optimizing wire harness costs for variation complexity

Executive summary

This paper will focus on the quantification of the complexity related costs in harness variations in order to model them, allowing automated algorithms to optimize for these costs. A number of real world examples will be provided as well. Since no two businesses are alike, it is the aim of this paper to provide the foundational knowledge and methodology so the reader can assess their own business to model how variation complexity costs affect their business.

Sjon Moore
Siemens PLM Software

*This paper was presented at the SAE World Congress, April 2016.
It is reproduced by permission of SAE.*

Introduction

An automotive wire harness rarely has just a single part number that can be ordered and installed in a vehicle. Typically, there are many different versions of the same harness based on the orderable content in the vehicle. These versions (often called harness levels) will have unique part numbers. The quantity of these levels and their content is what is typically called complexity and it has a significant impact on the cost of the harness.

Quantifying these costs is often very difficult especially with manual methods of deriving and costing the complexity solution. Therefore, traditionally, harness costing has focused on the piece cost of each harness level.

When these complexity related costs are considered it is typically with overly simplified cost modeling techniques.

This paper will focus on the quantification of these complexity related costs so that they can be modeled allowing automated algorithms to optimize for these costs. A number of real world examples will be provided as well. Since no two businesses are alike, it is the aim of this paper to provide the foundational knowledge and methodology so the reader can assess their own business to model how variation complexity costs affect their business.

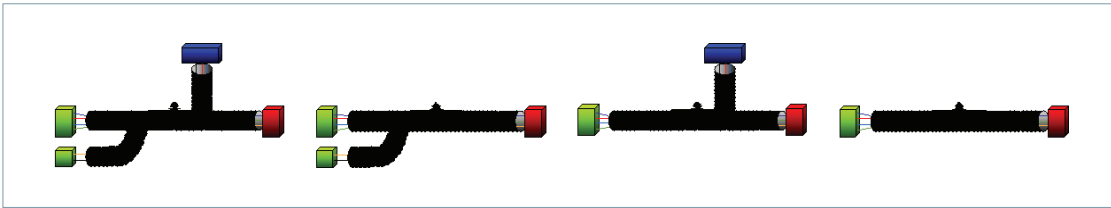


Figure 1: Different versions of a single harness based on what options are present on the vehicle.

The state of the art

It's not uncommon for organizations to have existing rules-of-thumb as a means of trying to capture these costs. For example, an organization might simply state that every new part number (harness level) equates to some standard cost (example: \$25k). This method is certainly better than not considering complexity related costs at all, and historically they've been "good enough".

Today, many automotive OEMs and tier ones are struggling with very thin margins and are looking to take pennies out of the material cost of the vehicle. In this kind of an environment, over-simplifications and rules-of-thumb just aren't "good enough". Without proper incentives, designs will be suboptimal and money will be wasted.

Some judgment is required to balance keeping the models simple enough that they can actually be understood and created, and not capturing enough in the

model such that engineers are incentivized to make bad decisions. This is particularly true in the case of complexity related costs since they're often transparent to most organizations and are absorbed in things such as logistics, plant down time, part scrap, etc.

The reality is that there are thousands of variables that slightly affect the cost of designing, manufacturing and shipping a harness. The prevailing ocean currents might affect the amount of fuel required for an ocean freighter to cross the Pacific. But, is such an unchanging and insignificant variable likely to cause one complexity solution to be more optimal than another? If not, it is probably safe to ignore it in the model.

It is ultimately up to each organization to determine the level of detail that should be captured and modeled for their business realities with the ultimate goal of enabling optimal design decisions.

Piece cost versus complexity management cost

For the purposes of this discussion, there are two primary variables of interest. The first and most obvious is cost which changes based on the variation complexity. Therefore, the second variable needed is the number of levels in the resulting complexity solution. This paper will use the graph in figure 2 as the foundation for discussion.

Wire harness manufacturers have a very good understanding of what it costs to assemble a wire harness. This piece cost includes the raw materials themselves from the various tier two suppliers, the labor associated with manipulating and assembling these raw materials, the logistics for storing and shipping a single finished product and so on. The cost of the wire harness is generally inversely related to the number of harness levels. In other words, as the complexity solution reduces the number of levels, the costs go up. To decrease the number of levels, content must be given away. For example, assume there are two harness levels that are identical except for some wires needed to support the fog lamp option. One could eliminate the level that does not have the wires, and use the level with the wires for all vehicles. By doing so, one has given away the wires on any vehicle that doesn't have fog lamps. Giveaway increases the cost of the harness because material is being consumed and handled that doesn't need to be.

There are additional costs that are not captured in the piece cost and are adversely effected by having additional levels. These are called complexity management costs and will be described in more detail later in this paper. By definition, these are costs that are directly related to the number of levels. As a short example, the more levels required for a harness, the more inventory that must be kept on hand. Additional inventory equates to additional costs. As mentioned, it's these costs that are difficult to quantify and are often ignored or simply estimated with rule-of-thumb calculations which typically just aren't good enough.

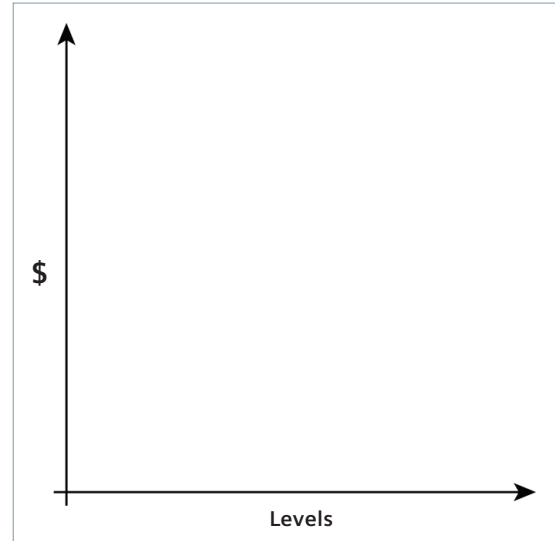


Figure 2: Number of levels and cost are the key variables.

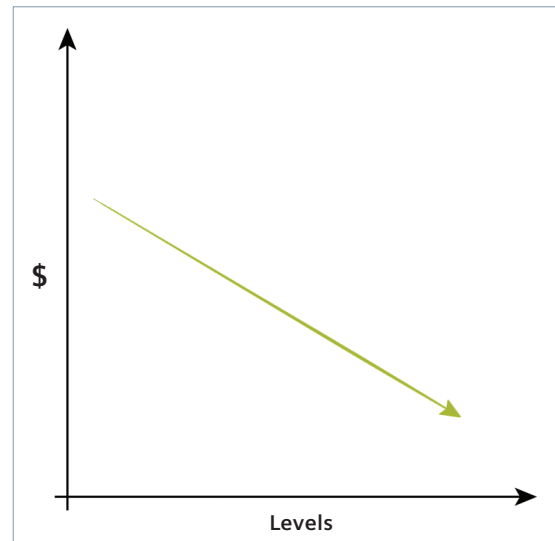


Figure 3: Piece costs decrease as the number of levels go up.

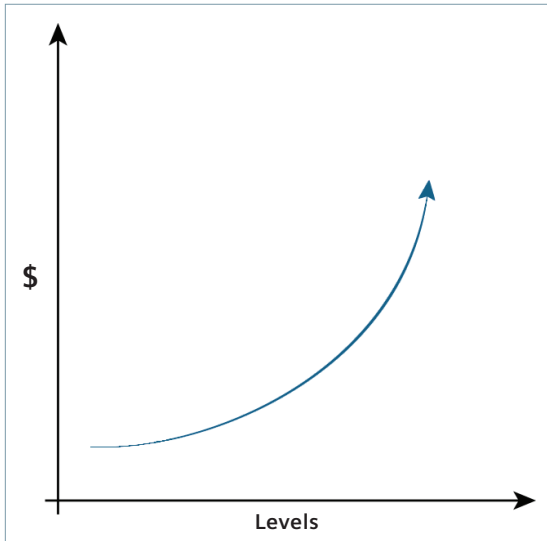


Figure 4: Complexity management costs increase as the number of levels increases.

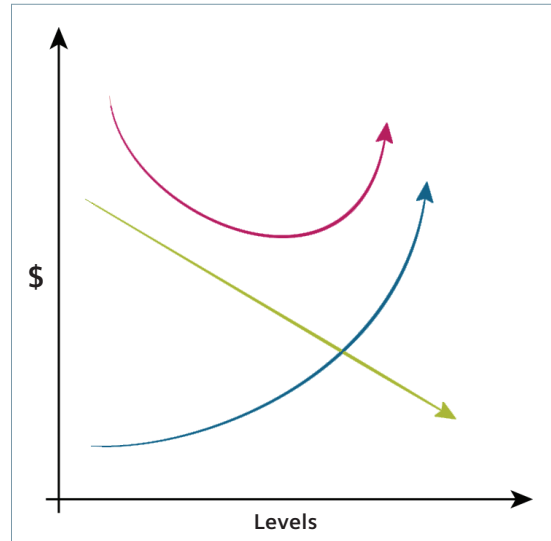


Figure 5: The actual cost of a harness is the complete cost model and is a combination of the two curves.

The reality of what a wire harness truly costs is the summation of these two curves which is called the complete cost model. The ideal complexity solution is the one that is at a minimum spot on the combined cost curve. Without adequately modeling this combined cost curve, it is impossible for a human or an algorithm to arrive at this minimum point except by pure chance.

The cost curves are typically shown as in the previous figures as continuous lines. However, since the number of levels is always an integer, in reality these aren't continuous lines. Also, there is rarely ever one complexity solution that will arrive at a number of levels. For example, there may be dozens of ways to give away different options to arrive at 12 harness levels. Some will be more efficient and less costly than others. This means that the complexity solution could fall anywhere above the optimal point.

When describing the costs, this paper will use the simpler continuous curves and assume that the human or algorithm arriving at the complexity solution is 100 percent efficient and always arrives at the ideal solution. This will be revisited later when considering practical applications.

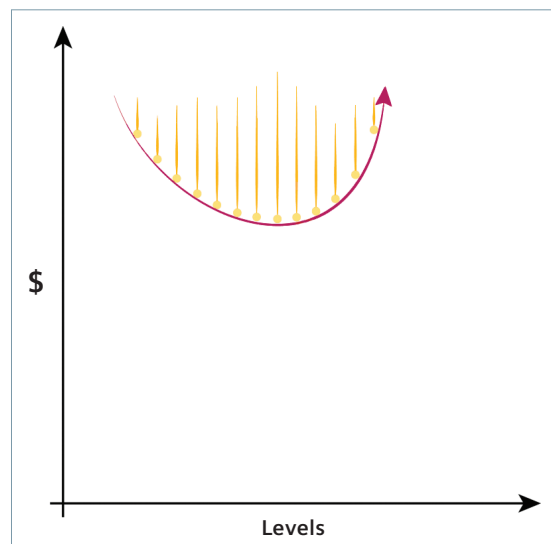


Figure 6: The use of continuous lines is a bit of a simplification since the number of levels is always an integer. Complexity solutions fall along the yellow lines.

Examples of complexity management costs

A number of factors contribute to complexity management costs. Some common examples are described in the next subsections. Many of these could be broken down further into unique costs. Different businesses will be impacted differently by each one, and they will certainly have additional ones not mentioned here. These examples are meant to cover some common cost drivers and lay a foundation for the reader to uncover similar costs in their own organization.

Engineering and development

Each harness level in the harness design requires effort to engineer and validate, the cost of which is typically absorbed in the design and manufacturing engineering organizations. If there is only one level, the development and validation costs are at their lowest. By adding a single additional level, this new level must be considered every time the design is touched. It must be validated. And, manufacturing engineering must understand this level and ensure that the resulting production processes consider it and incorporate the necessary tooling to accommodate change over.

It is probable that an adequate model of engineering and development would want to consider product engineering and manufacturing engineering separately as they likely will have different regional labor costs. It's also possible that validation costs should be a unique cost.

Production

The production processes must be capable of manufacturing any and all levels within a given harness design. This means that tooling must exist for each and every level. Not only must this tooling be created, it must also be validated. Additionally, any time that the line must be shut down so that the tooling can be changed to start manufacturing a different level costs are incurred, including labor down time and the cost of physically changing the fixtures.

Inventory/logistics

It's very common for an automotive assembly plant to broadcast their projected orders no sooner than several days in advance (perhaps only a few hours). It typically takes longer than this to manufacture and ship parts from the harness manufacturing facility (usually in a low cost country) to the final assembly plant. This often requires the harness supplier to have some amount of inventory on-hand within the logistics pipeline to cover very short order times.

As there are more harness levels, more inventory is required because it's impossible to know what the assembly plant will order. The pipeline must get thicker. This will require more space in storage facilities, or perhaps even new brick and mortar. The trade-off might be new and perhaps more expensive logistics avenues such as air freight versus truck or ships.

Sequencing

In the final assembly plant, the line worker retrieves a part from one or more bins located very near the station in which they are working. If the number of harness levels is low, one bin for each level can normally be placed near the station and the worker can grab the correct part from the correct bin. If the number of harness levels is high, there simply isn't enough floor space to store all of the bins for the worker to choose from, plus the chance of mistake increases.

To combat this, parts are sequenced onto the line. There are different methods of doing this, but a common is for a third party to be hired to manage a facility very near the final assembly plant that has inventory from the wire harness supplier. From the assembly plant, they receive the orders for each vehicle to be built that day, which harness level each requires and in which order the vehicles will be built. They then take harnesses out of their inventory (see the previous point) and place them in the bins based on the sequence in which the vehicle will be built. This way, the line worker simply has to pull harnesses from the bins in the correct order.

This third party obviously absorbs the same inventory costs described earlier, but the business arrangements made with the OEM are typically different. The OEM will want to model this cost based on the contractual agreement. This can often be a significant complexity management cost.

Obsolescence

Obsolescence occurs when a part has been manufactured, but due to design changes, is no longer usable. This often occurs at the end of a production year when new parts (or revisions) are required due to new feature content. Sometimes old model year parts can still be used. When they can't, they must be reworked such that they are usable, or recycled for their raw materials (copper, plastic, etc.). The process of reworking parts and the loss associated with scrapping material is a significant cost which should be modeled and accounted for.

When there are very few part numbers, the expected annual volume of each level is relatively high. For example, assume there are four levels and there will be 100,000 units built in one year. Also assume a completely even distribution, each level will be produced 25,000 times. When the number of levels increases, the volume for each level thins out. When real world take rates (the percentage of vehicles that will require a given option) are applied, some levels will only be used a few times, or not at all.

Harness manufactures normally have a minimum purchase amount. In other words, they aren't going to change the line over to build just one unit of one level. They're going to build some minimum number and the OEM is required to purchase some minimum number. Assume it is 32 because that's what fits in a standard container. Now assume that the OEM only used one of those throughout the year. The remaining 31 will be obsolete and must be recycled. The harness supplier will typically have some inventory themselves that will be obsolete.

Mistakes

Any process that has human interaction is bound to result in mistakes. Methods of reducing the number and severity of mistakes include automation of as many design tasks as possible, elimination of re-entry of data, improving data flow through the design process and so on. But even state-of-the-art tools and methodology still rely on some human interaction and simply cannot catch every human error. As such, mistakes are a fact of life.

One example of a mistake that is difficult to catch are errors when an engineer releases their part but provides the incorrect options or feature codes. If they release a level and say that it supports an option that it in reality doesn't, cars will be built that don't work because wiring will be missing. Automation can assist in this by removing the re-entry and translation of this data by a human, but this is relatively rare with many OEMs. Another example which is very difficult to defend against is if an engineer incorrectly defines the option relationships in the wiring or logical system designs. Say a device must be present if the customer orders option A or option B. If the device is simply tagged with option A, the device's wiring will not be present on the vehicle that ordered option B and again, cars will be built that don't work because wiring will be missing.

Quantifying the impact of the number of harness levels to the number and severity of mistakes is challenging because it's often based on the likelihood that a mistake will happen.

Modeling complexity management costs

The previous section described a number of common complexity management costs that are seen by organizations. The aim now is to properly model these costs so that something meaningful can be done with them. To reiterate, the goal is to capture and model these costs in sufficient detail to enable humans and automation to make the right decisions and arrive at optimal designs. To do this, one must first characterize the costs in such a way that mathematical models can be applied. Following are the fundamental questions that must be considered.

Penalty or bounding?

In most cases it's preferable to model the costs as penalties that are affected by the complexity solution. For example, the engineering and development costs will increase as the number of levels increase. In other cases, a penalty can't be used because it's a bounding factor based on some physical reality or a business contract. For example, perhaps there is a contract in place between the OEM and Sequencing facility that states that the maximum number of levels is 20. This is a bounding factor and must be handled appropriately such that no complexity solutions are considered that have more than 20 levels, no matter how suboptimal the solution might be.

Bounding factors simply remove complexity solutions that can be considered. In an automated complexity optimization algorithm, the ideal design is to allow the user to enter this bounding factor and design the algorithm to respect it. If not, then a simple work around is to create a step (discussed later) with an extremely high cost at the bounding point. This should prevent any algorithm from attempting complexity solutions above the bound.

Penalties are the more common and far trickier costs to capture. The remaining fundamental questions focus on these penalty type costs.

What triggers the penalty?

The penalty is being affected by some aspect of the design. Is it the number of harness levels? This is the most common case. Examples of this include production costs, sequencing, and engineering and development costs. Or is it the volume of any specific level that drives the penalty? This is common for obsolescence costs. As the volume of the levels thins out and becomes too low, the likelihood that a good portion of those levels will have obsolete parts at the end of the year becomes much higher.

What is being penalized?

This is a question of accounting more than anything else. What object or aspect of the design is absorbing the cost? Typically, this will be either the harness itself or individual harness levels. This is usually tied very closely with what is triggering the penalty. For example, if the trigger is low volume levels, the harness level is typically what will be penalized.

What is the math?

Once the variables are understood, it's now important to understand mathematically how the variables affect the cost. Here are the mathematical concepts often used.

Continuous versus step

The relationship between the variable and the cost might be a single continuous function¹ or it might have discontinuities that create "steps"². These steps can occur anywhere there are tipping points in the relationship. For example, the inventory costs might have a tipping point if a certain number is exceeded because it will require an additional storage facility which causes a sharp jump in the cost if the number of levels exceeds that threshold. When modeling a complexity management cost, always look for these telltale tipping points so that they can be modeled as steps.

Scaling

How do the costs scale based on the change in the variable? Is it a direct linear relationship (example: each new harness level adds \$0.25 to the overall cost)? Is it an exponential relationship (example: each new harness level doubles the overall cost)? The way in which the cost scales with the variable will indicate which type of mathematical function³ that is required.

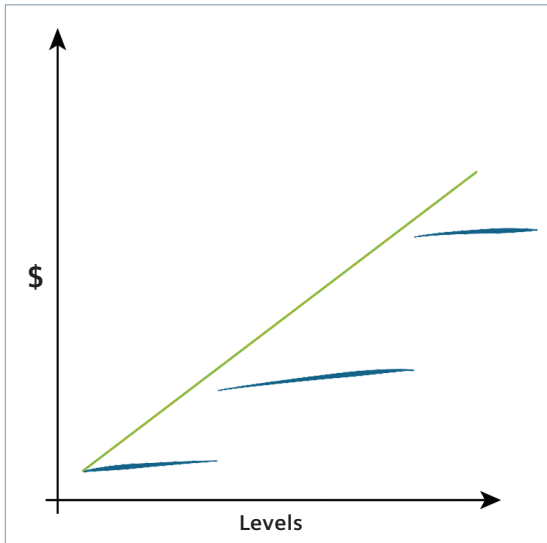


Figure 7: Continuous versus step functions.

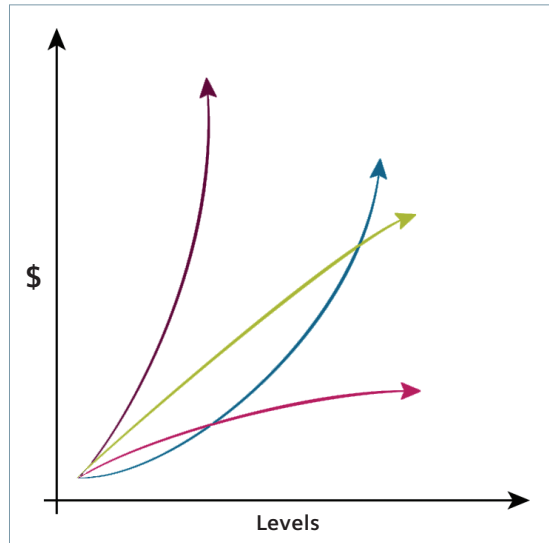


Figure 8: Costs can scale linearly, exponentially, logarithmically, etc.

Risk and probabilities

Some costs aren't guaranteed to happen and are strictly based on chance. A mistake is a common example. It's difficult, if not impossible, to precisely predict when a mistake will occur nor the severity of the consequence. For these types of costs, a risk assessment type calculation is the best method of quantifying the costs that will be incurred and spread the cost across each unit built throughout a year. This is similar to how insurance works⁴. A "premium" will be "paid" on each harness created which should pay for the few very costly mistakes that are likely to happen. Of paramount importance is that the "premium" must be sensitive to the factors that increase the probability of a mistake. Normally this is the number of harness levels. This provides a method of quantifying and capturing this cost and understanding how it varies by complexity such that optimization is properly incentivized to minimize the risk of a costly mistake.

Risk is a product of the probability of an event and the severity of the consequence of the event⁵. For the manufacture of harnesses, the consequences and their probabilities are best banded⁶ based on when they would be captured in the process. Table 1 illustrates this.

Each band has a progressively more expensive consequence⁷. Table 1 provides an example of how an

organization might want to band their risks. If the cost of a mistake is known from historical data for each band, enter it into the corresponding cell. If not, estimate the cost of the mistake if caught in the first band (in table 1, it would be the engineering band), then use the scaling factors provided in table 2 to estimate the cost for the remaining bands. If historical data is not available for the probability cells, table 3 provides a set of data that can be used for initial estimations.

For purposes of modeling this cost and understanding how an increase in harness levels affects the risk, generally speaking, additional levels results in a higher risk. This is why the risk calculation for each band should include the "number of harness levels" variable. The risk for each band is the product of the probability of the event, the cost of the failure, and the number of levels.

Lastly, the model should sum the risks from each band (see figure 9). Once this is done, each harness will be penalized a nominal amount for potential mistakes, and this calculation will adjust up and down based on the number of harness levels.

$$C_{total} = R_{eng} + R_{val} = R_{mfg} + R_{log} + R_{assy} + R_{field}$$

Figure 9: Total cost for mistakes is the summation of all the risk related costs for each band.

Band	Cost (\$)	Probability (\$)	Risk (\$)
Engineering	C_{eng}	P_{eng}	$R_{eng} = C_{eng} \times P_{eng} \times N_{levels}$
Validation testing	C_{val}	P_{val}	$R_{val} = C_{val} \times P_{val} \times N_{levels}$
Manufacturing	C_{mfg}	P_{mfg}	$R_{mfg} = C_{mfg} \times P_{mfg} \times N_{levels}$
Logistics	C_{log}	P_{log}	$R_{log} = C_{log} \times P_{log} \times N_{levels}$
Assembly	C_{assy}	P_{assy}	$R_{assy} = C_{assy} \times P_{assy} \times N_{levels}$
Field	C_{field}	P_{field}	$R_{field} = C_{field} \times P_{field} \times N_{levels}$

Table 1: Risk associated with different stages.

To reiterate, tables 2 and 3 below provide some sample values that an organization might want to use as a testing or starting point until actual studies of the organization's performance can be studied. Once values specific to the organization are captured, they should be used instead.

Alternatively, an organization may decide that rather than taxing the harnesses (a penalty) it might be preferable to simply use the sum of the risks as a bounding factor. In other words, any complexity solution that results in a risk greater than some pre-determined value is unacceptable and must be reduced further until the risk is within the provided limit.

Variable	Sample value
C _{eng}	1
C _{val}	5
C _{mfg}	10
C _{log}	30
C _{assy}	1,000
C _{field}	10,000

Table 2: Sample values for the banded costs to use as a starting point until studies can be done to determine better numbers for a given organization.

Variable	Sample value
P _{eng}	1%
P _{val}	0.1%
P _{mfg}	0.01%
P _{log}	0.001%
P _{assy}	0.0001%
P _{field}	0.000001%

Table 3: Sample values for the banded probabilities.

Putting it all together

Any single complexity management cost may incorporate several of these factors. It might include a couple of steps with different scaling factors in between. It might also include a bounding factor based on a contractual obligation or an unacceptable risk of mistake. When complete, a single cost will have been modeled. But as mentioned from the outset, to truly optimize the complexity solution, all such costs must be modeled adequately. The complete cost model isn't fully defined until all such complexity management costs are modeled.

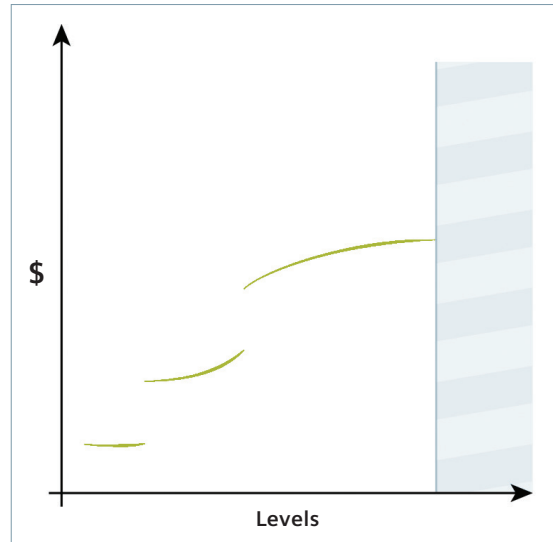


Figure 10: An example of a single complexity management cost incorporating many concepts previously discussed.

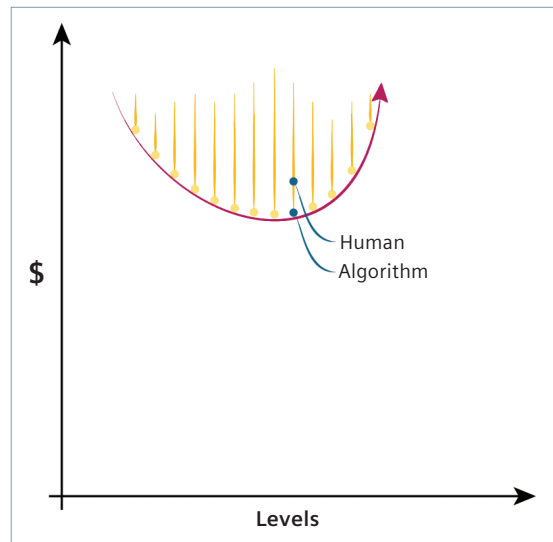


Figure 11: Is it reasonable to expect a human to outperform a machine in calculating optimal complexity solutions?

Practical applications

So far this discussion has focused heavily on creating a theoretical model to describe the real world. It might be tempting to try to make the model more accurate than is really required. The model only has to be good enough to make good decisions. It's not really important that the model take into account slight differences in productivity between the first and second shifts unless they're large enough that they could actually alter how a person or an algorithm would optimize the complexity solution.

It's also important to note that many of these costs are actually shared between OEM and Tier 1 and each may model the cost differently. For example, both will suffer from obsolescence costs, but how they do so is different. The OEM's cost will be largely dictated by the minimum order sizes defined in the contract or by minimum container sizes. The Tier 1 might need to model obsolescence based on the plant's minimum build run as well as the pipeline of inventory. Contractual obligations will dictate who might assume the cost of obsolescing inventory in the sequencing facility.

Earlier in this paper, it was assumed that the human or the algorithm determining the complexity solution would be 100 percent efficient and would always arrive at the most cost effective solution. But, is it practical to assume that a human could understand the models that have been described up to this point, then use them to manually derive a solution more effectively than a computer algorithm can? Spending time to accurately model complexity management costs is of little value if these models cannot be used to make good decisions and arrive at an optimal solution very quickly and repeatedly. Only a computer algorithm could quickly balance the variables described while testing such a large solution space and arriving at an optimal solution. Additionally, only a computer would allow one to adjust a few variables in the models (perhaps to simulate a new business contract) and quickly arrive at a new complexity solution with complete costs to fully understand the impact. Computer based solutions also reduces the inevitable mistakes which will lower the probabilities in the bands of table 1.

Case study

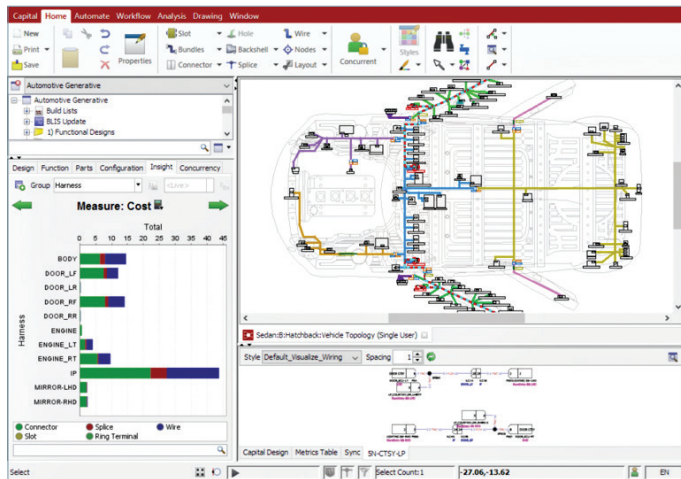


Figure 12: Platform-level design for the case study.^{8, 10}

Application of the above concepts using real world software will provide a better understanding of a) how one might model their complexity management cost, b) how those models can be used to arrive at optimized complexity solutions, and c) how adjustments to the cost models will affect the optimized results. Such software does exist and is commercially available.

This paper will use the Capital suite from Siemens PLM Software to show the above concepts. The functions applicable to this case study include:

- Modeling and reporting the complexity management costs⁸
- Level management for automatic optimization of the complexity results⁹
- A design environment for the platform wiring design¹⁰

Assumptions

The design to be used in this case study will be the out-of-the-box sample data provided with the Capital installation. It is intended to be representative of a modern automobile and will provide the basic design data required. This case study will use this sample set of data to represent the vehicle wiring design.

For this study, assume the following complexity management costs are present and must be modeled and accounted for:

Cost 1 There is a 0.25 cost for each harness level in the solution above 10 levels (11 or more). This cost is incurred by various design, production and logistics costs.

Cost 2 The contract with the sequencing facility requires that there be less than 20 harness levels (19 or fewer).

Modeling

The complete cost model is captured as a metric. Individual metric elements will capture the various piece and complexity management costs. Based on the description for the use case, each complexity management cost has the following answers to the questions outlined above.

Cost 1

Penalty versus bounding?
This is a penalty.

What triggers the penalty?
The number of harness levels.

What is being penalized?
The harness.

What is the math?
It's 0 for levels 1 through 10, then there is a step at 11. For 11 and beyond it is a linear scale of 0.25.

Cost 2

Penalty versus bound?

This is a bound at 20 levels (19 and fewer are acceptable). The rest of the questions are irrelevant.

Putting this all together, the model looks like figure 13.

As shown in figure 14, the complete cost model includes piece costs such as costs for wires, splices and connectors, as well as the complexity management costs just described. Cost 1 will be modeled by creating a metric element with a single definition that penalizes the harness (question d). The condition for the element will test whether the number of harnesses (question b) in the solution exceeds 10. If it does, then the calculation will be applied. In this case, the calculation subtracts 10 from the number of harnesses (because the model only penalizes those levels that are above 10), then multiplies this by 0.25.

Cost 2, which is the bounding factor could also be captured as its own complexity management cost. To do so, create a single metric element with a single definition. It will apply to the harness and have a condition such that it only triggers when the number of harnesses is greater than or equal to 20. Then, a very large value is applied to prevent optimization algorithms from going above 20. Though it is shown here as an example, this isn't required for Capital software since it allows the user to define these bounding factors directly.

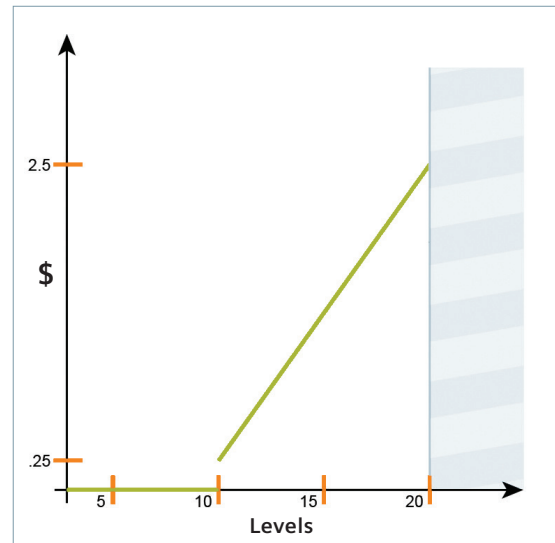


Figure 13: Theoretical model of the case study's complexity management costs.

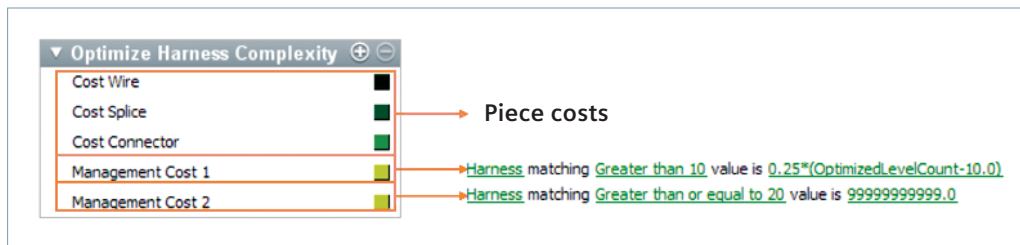


Figure 14: The complete cost model including complexity management costs modeled in Capital.⁸

Automatic optimization

Software automation is then used to optimize the complexity solution. The bounding factor from cost 2 which is 20, is entered as "19" into the define giveaway dialog as the upper limit, shown in figure 15. This tells the algorithm to never consider solutions above this value. The optimization algorithm then finds the optimal solution based on all of the inputs and returns an overall cost based on the complete cost model which includes all piece and management costs.

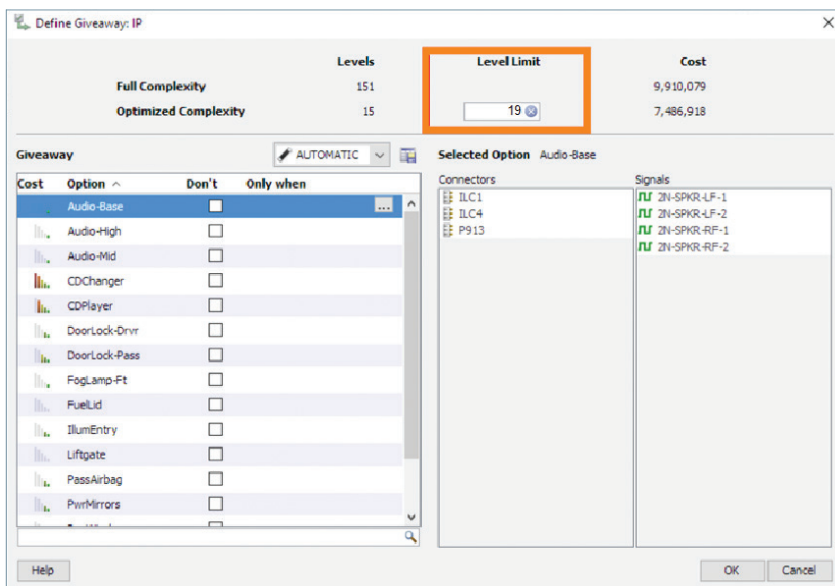


Figure 15: The upper limit provides the bounding factor in the Capital tool suite, which makes it unnecessary to model cost 2.^{8, 10}

Visualizing the complete cost model

The optimization algorithms always return the single most optimal solution given the design data and the complete cost model. If one wishes to visualize the resulting complete cost model for the given design, it can be done by many optimizations. By entering progressively smaller upper limits, then recalculating the optimization results, one can walk the complete cost model to the left. By taking the resulting cost output after each optimization and placing them in a spreadsheet program, a graphical view of the left side of the complexity cost model can be easily constructed.

Figure 16 shows the comparison of the theoretical complete cost model against what was achieved through models and automation with the Capital suite. This includes piece costs for all of the harness and wiring content for an example platform design, plus the complexity management cost that was modeled per the case study.

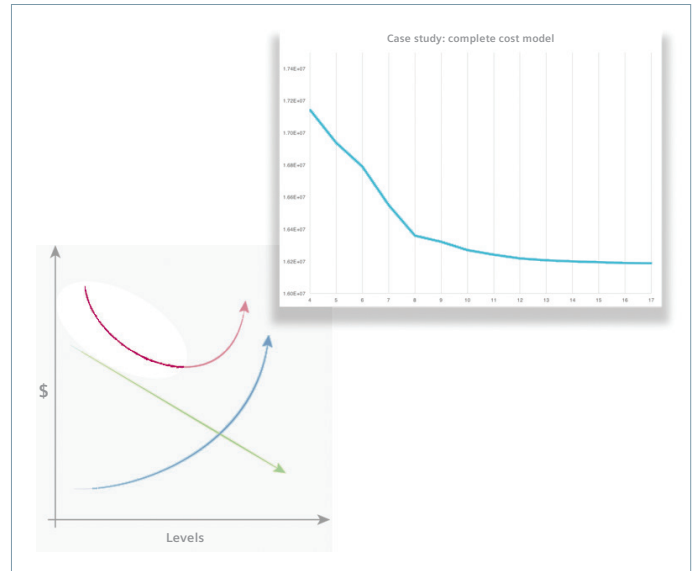


Figure 16: Comparison of the theoretical complete cost model versus the results of modeling in Capital.

Observing impact of change

One key value in using robust modeling techniques as advocated in this paper is that changes in the business environment can be relatively easily captured with simple modifications to the models. Using the same case study, assume that a change in labor rate causes the cost 1 scale of 0.25 to adjust to 0.3. It's a simple matter to modify the model.

Next, automated optimization is again used to determine the new optimal solution. Figure 18 shows a comparison between the two curves. Note that the updated cost model has resulted in a different curve (orange). It has a slightly different shape, is overall more expensive and even resulted in a different number of harness levels for the optimal solution (18 versus 17).

In this way, changes in the organizations business and operating parameters that will affect harness costs can be quickly analyzed, understood and accounted for. In the example above, a change in one variable shows that the current complexity solution is no longer the optimal solution. Armed with this type of data an organization can decide if they wish to change to the new optimal design. If they choose not to, they'll at least have an understanding of how far from optimal they now are.

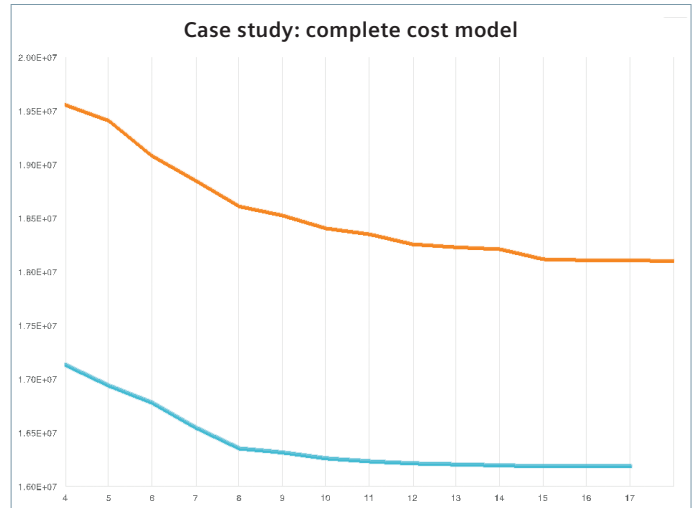


Figure 18: Comparison of original cost model to updated model.

Before	<u>Harness matching Greater than 10 value is 0.25*(OptimizedLevelCount-10.0)</u>
After	<u>Harness matching Greater than 10 value is 0.3*(OptimizedLevelCount-10.0)</u>

Figure 17: Adjusting the model to reflect new business realities⁸.

Conclusion

Historically, automotive companies have focused on understanding the piece costs of a wire harness and in many cases largely ignored the impact of variation complexity on the overall costs absorbed by the organization. This paper has provided a number of real world examples of such costs, has provided a methodology for defining these factors so that they can be modeled, and given guidance on methods for modeling them based on how their variability affects overall costs.

Doing so is less daunting than many might initially think since the resulting models only need to be good enough to make good decisions. Also, the use of automation is highly recommended as it's unlikely that a human could repeatedly arrive at solutions as optimally as a computer can.

Lastly, a case study was shown in which just such an algorithm was used to show how such costs could be modeled, how the complexity solution can be optimized based on these models, and how the solution and its overall costs change as variables in the model change.

Organizations are encouraged to seriously consider the provided methodologies for capturing and modeling their complexity management cost as well as utilize an automated optimization solution so they can fully understand the costs that they are incurring, and can optimize their designs accordingly.

References

1. Wikipedia, "Continuous function", https://en.wikipedia.org/wiki/Continuous_function, accessed January 5, 2016.
2. Wikipedia, "Step function", https://en.wikipedia.org/wiki/Step_function, accessed January 5, 2016.
3. Wikipedia, "Function", [https://en.wikipedia.org/wiki/Function_\(mathematics\)](https://en.wikipedia.org/wiki/Function_(mathematics)), accessed January 5, 2016.
4. Wikipedia, "Underwriting and Investing", https://en.wikipedia.org/wiki/Insurance#Underwriting_and_investing, accessed January 5, 2016.
5. Wikipedia, "Risk", <https://en.wikipedia.org/wiki/Risk>, accessed January 5, 2016.
6. Wikipedia, "High-reliability organization", https://en.wikipedia.org/wiki/High_reliability_organization, accessed January 5, 2016.
7. Stecklein, Jonette M., et. al., "Error Cost Escalation through the Project Life Cycle", <http://ntrs.nasa.gov/search.jsp?R=20100036670>, June 19, 2004.
8. Mentor Graphics, Capital Insight description, <https://www.mentor.com/products/electrical-design-software/capital/integrator>, accessed January 5, 2016.
9. Mentor Graphics, Capital Level Manager description, <https://www.mentor.com/products/electrical-design-software/capital/capital-level-manager>, accessed January 5, 2016.
10. Mentor Graphics, Capital Integrator description, <https://www.mentor.com/products/electrical-design-software/capital/integrator>, accessed January 5, 2016.

Siemens PLM Software

Headquarters

Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 972 987 3000

Americas

Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 314 264 8499

Europe

Stephenson House
Sir William Siemens Square
Frimley, Camberley
Surrey, GU16 8QD
+44 (0) 1276 413200

Asia-Pacific

Unit 901-902, 9/F
Tower B, Manulife Financial Centre
223-231 Wai Yip Street, Kwun Tong
Kowloon, Hong Kong
+852 2230 3333

About Siemens PLM Software

Siemens PLM Software, a business unit of the Siemens Digital Factory Division, is a leading global provider of software solutions to drive the digital transformation of industry, creating new opportunities for manufacturers to realize innovation. With headquarters in Plano, Texas, and over 140,000 customers worldwide, Siemens PLM Software works with companies of all sizes to transform the way ideas come to life, the way products are realized, and the way products and assets in operation are used and understood. For more information on Siemens PLM Software products and services, visit www.siemens.com/plm.

siemens.com/plm

© 2019 Siemens Product Lifecycle Management Software Inc. Siemens, the Siemens logo and SIMATIC IT are registered trademarks of Siemens AG. Camstar, D-Cubed, Femap, Fibersim, Geolus, GO PLM, I-deas, Infolytica, JT, Mendix, NX, Parasolid, Polarion, Simcenter, Solid Edge, Syncrofit, TASS, Teamcenter and Tecnomatix are trademarks or registered trademarks of Siemens Product Lifecycle Management Software Inc. or its subsidiaries or affiliates in the United States and in other countries. All other trademarks, registered trademarks or service marks belong to their respective holders.

77779-A12 4/19 C