

Huge Growth in Revenue and Profit Using AI LLM Complexity Reduction

Michael L George, Founder of the George Group
Executive overview of AI LLM by Dr. Michael Hahsler
www.fightcomplexity.com
Mike@entropy2718.com, 214 215 4455



Prologue:

Early in my career, I left Texas Instruments and became the CEO of a manufacturing company, International Power Machines. Our company had started out producing 5KW Uninterruptible Power Supplies (UPS) which converted DC battery power to protect 60 Hz AC power.

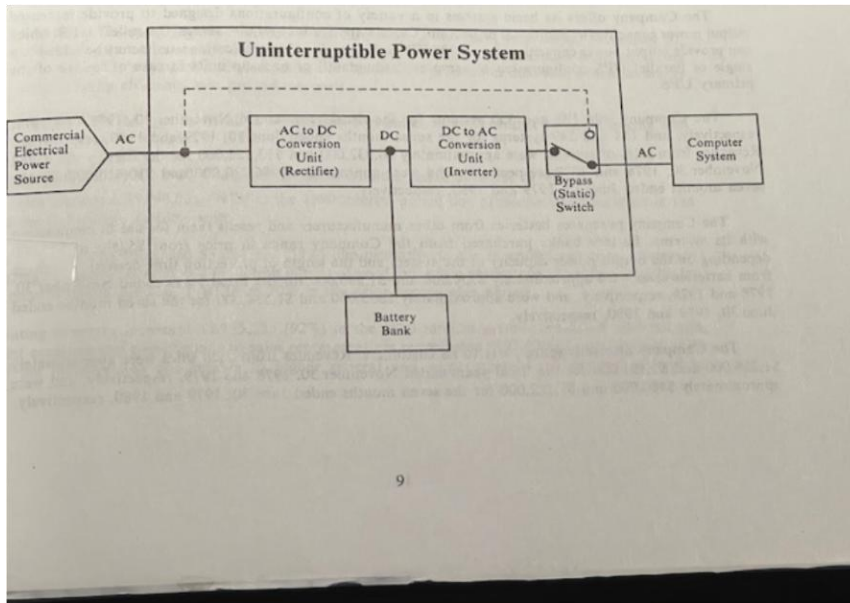


Fig 1

IBM was then dominant in real-time computing, and required uninterrupted power. Over the years, different engineers designed 10KW, 25KW ...to 160KW in our case, plus a 400HZ machine for the IBM 370/168, a total of 8 different machines with differing AC input. Higher power was achieved by increasing internal DC voltage from 130 VDC to 260VDC to 390VDC. Each machine KW rating required different components, and hence over several years Engineers designed eight entirely different mechanical designs to satisfy a growing, but unfortunately breakeven business.

I finally read the book “Kaisha” and learned that, since each KW rating had a different mechanical design, we had 8 different “families” of machines. “Kaisha” data proved that Manufacturing Overhead increases with the number of different families produced. Our company was at the red star in Fig 2. If we could redesign our products to reduce the number of families from 8 to 1, our Overhead cost would fall by 40%!

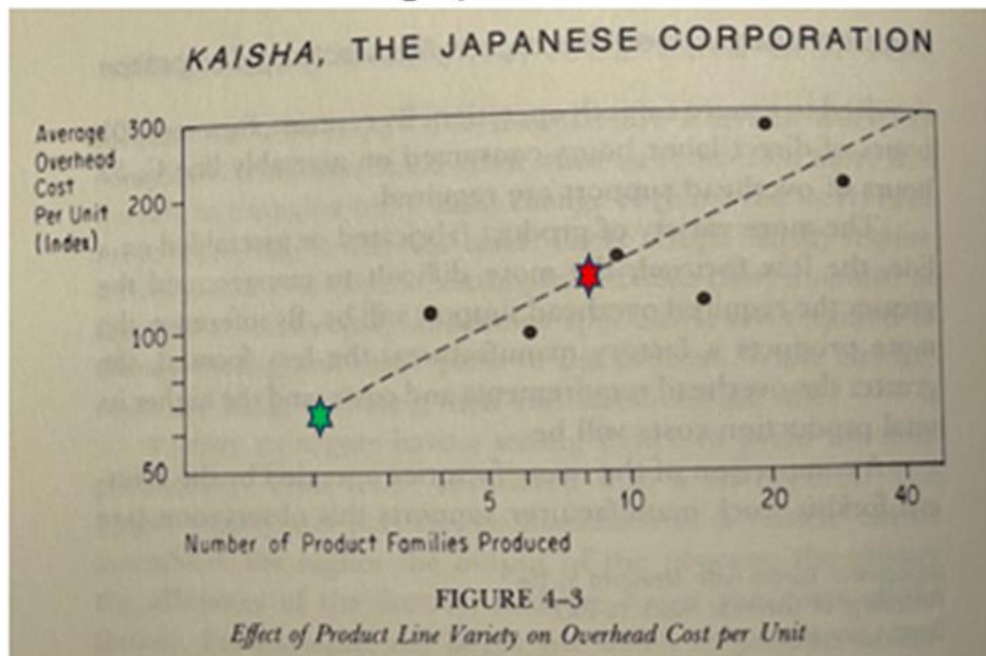


Fig 2

I hired a Manufacturing Engineer named Lee Freeman from Mattel Toys, and assigned my best Design Engineer Frankie Spross, to work with Lee to drive down the number of different families. They manually examined the ERP Parts lists of all eight different families and created a single mechanical design with variation achieved solely due external supplier part numbers which did not affect the mechanical design labor. All internal assembly was achieved with one mechanical design. This was an arduous six month effort of examining 1000 supplier and internally fabricated parts to create one family with only 300 part numbers. Converting from eight to one mechanical design caused Manufacturing Overhead and Labor to fall by 40% and pre-tax profit increased to 20% of revenue. This data appears in the Public Offering memorandum cited in fig 2.

This reduction of Manufacturing Overhead and Labor was only possible because we could *manually* detect and reduce a plethora of 1000 parts to 300, and from eight different mechanical designs with one. As just one of scores of examples, electrical current was conducted in copper bus bars, which increased in thickness with current rating. We had 5 different thicknesses for each bus bar, each requiring a setup in the machine shop, separate inventory stocking, kitting etc. We converted all “near miss” bus bars to the largest thickness, dramatically reducing these overhead costs. The five times higher volume kept material cost constant. A *single mechanical design* replaced all eight families.

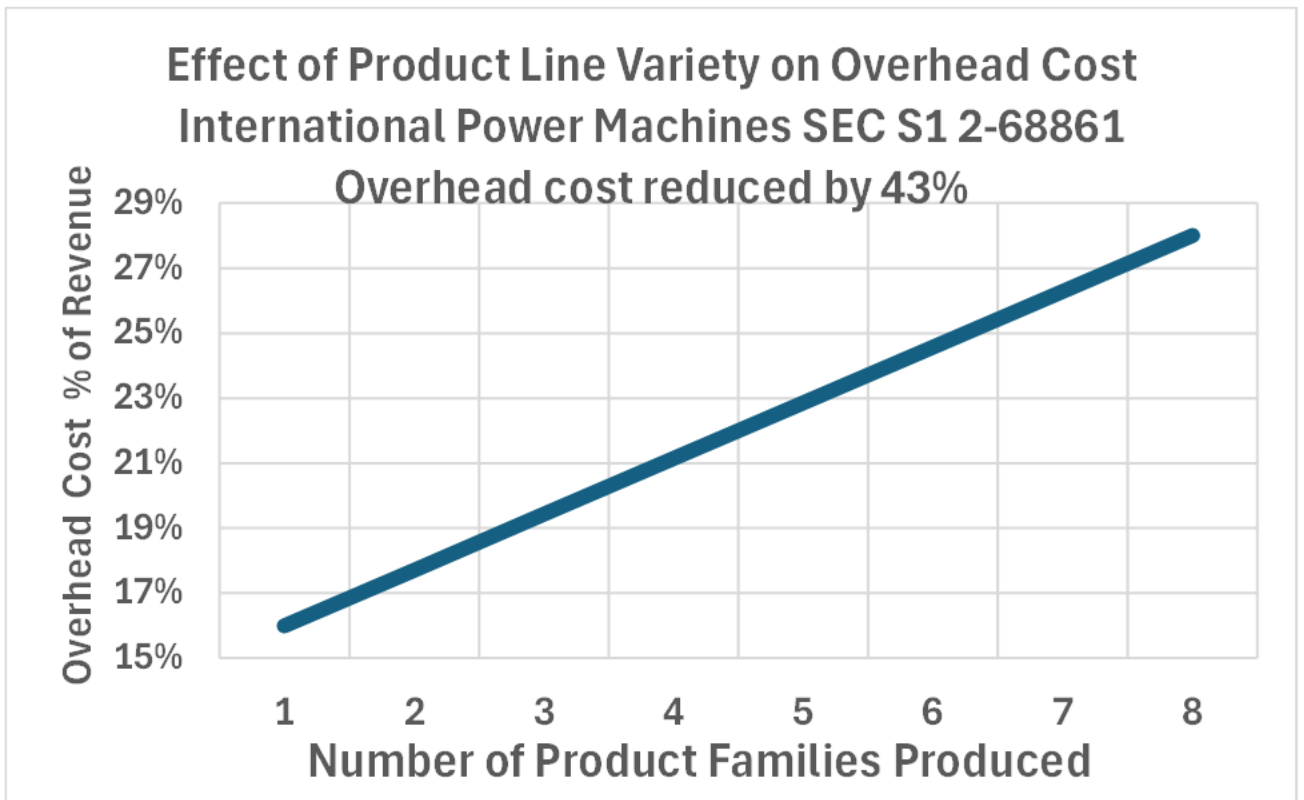


Fig 3

Overhead costs in purchasing, stockroom ,kitting , Quality control etc. were radically reduced. In addition, labor cost was reduced by ~50% by excellent work instruction made possible by 1 versus 8 families, virtually no shortages, no wiring errors, no delay to test , or potential failure at the customer site causing expensive warranty cost resulting in increasing revenue

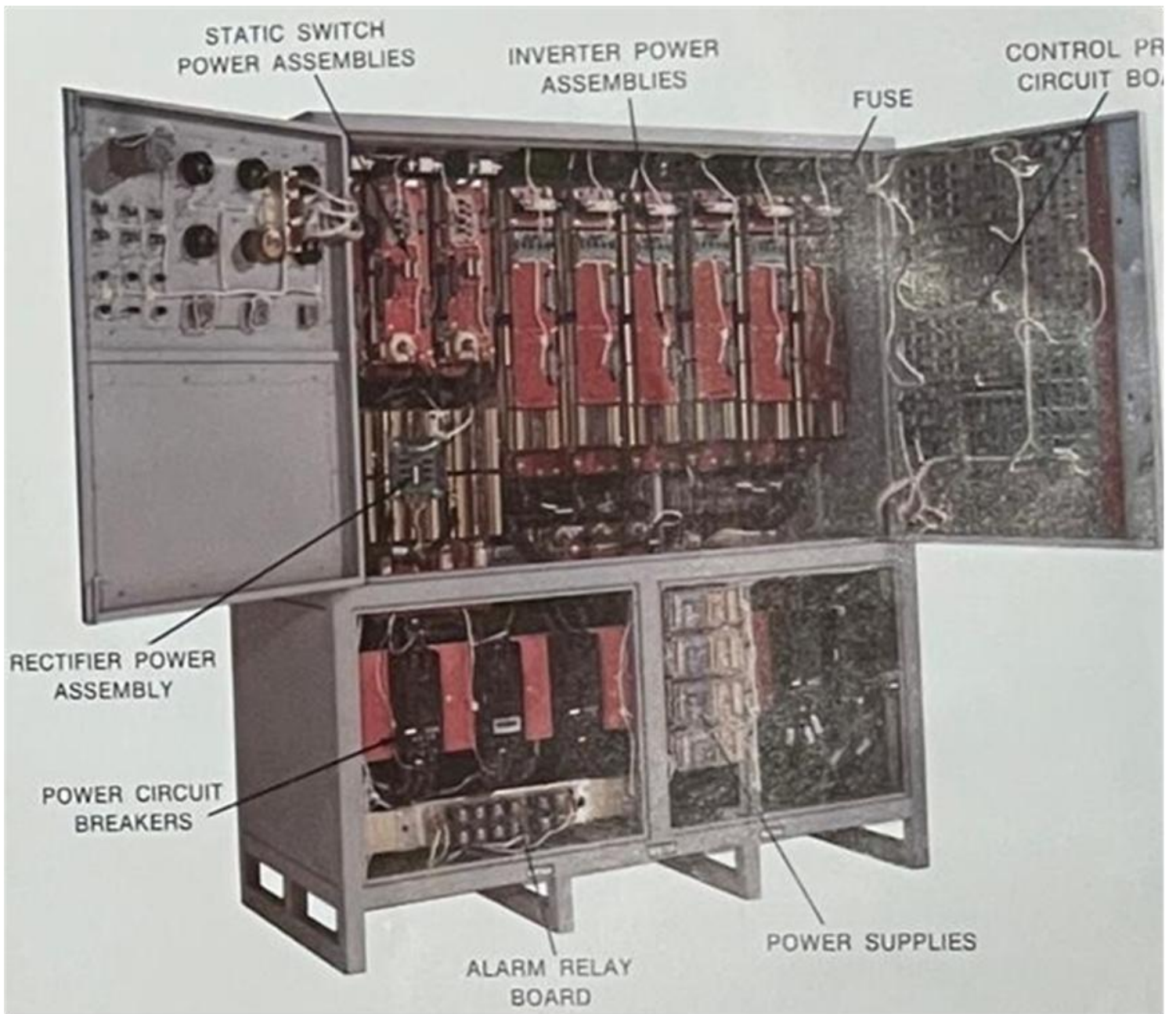


Fig4. 8 separate families were replaced by 1 mechanical design for 10KW through 160KW

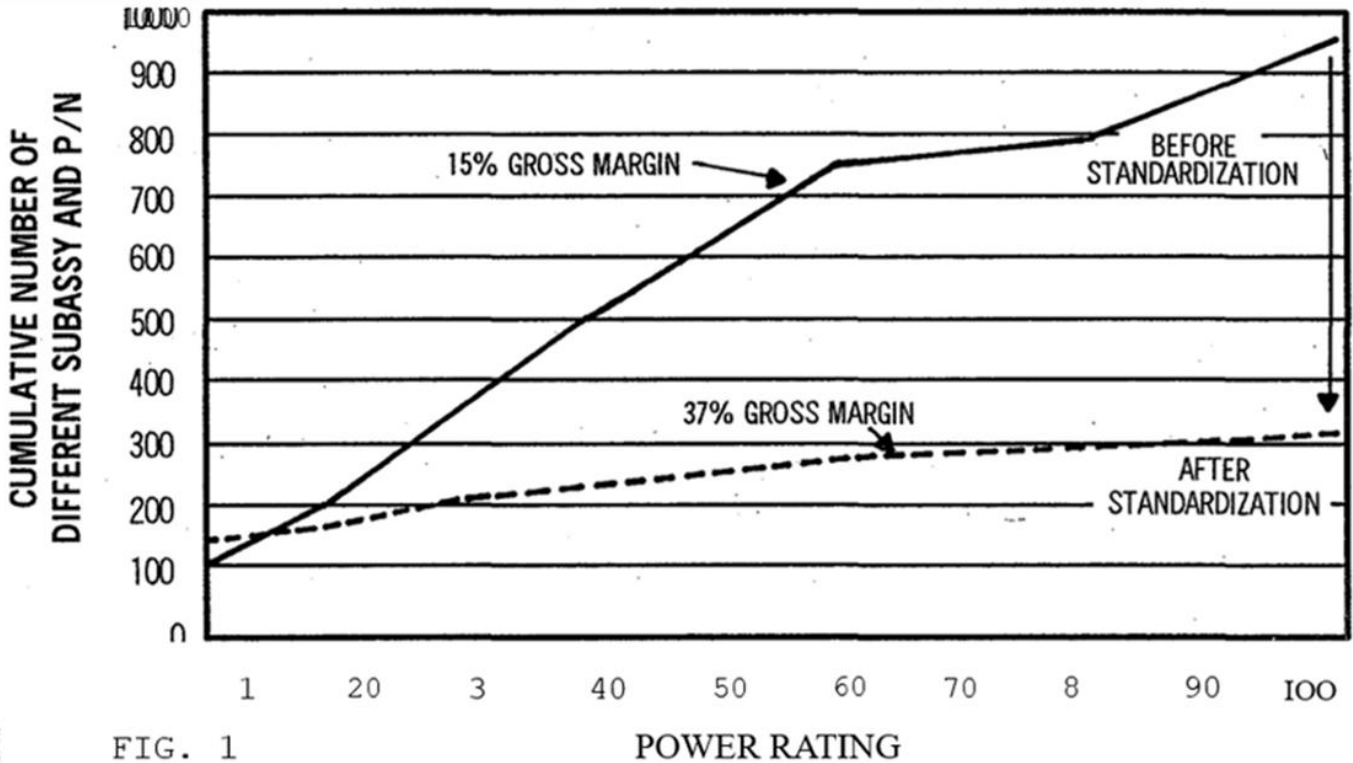


FIG. 1

Fig 5: Gross Profit is more than doubled by Complexity reduction

As the number of different parts fell from 1000 to 300, the amount of Work in Process fell. The delivery time to produce Finished Goods fell, thus reducing the Manufacturing cycle time according to Little's Law:

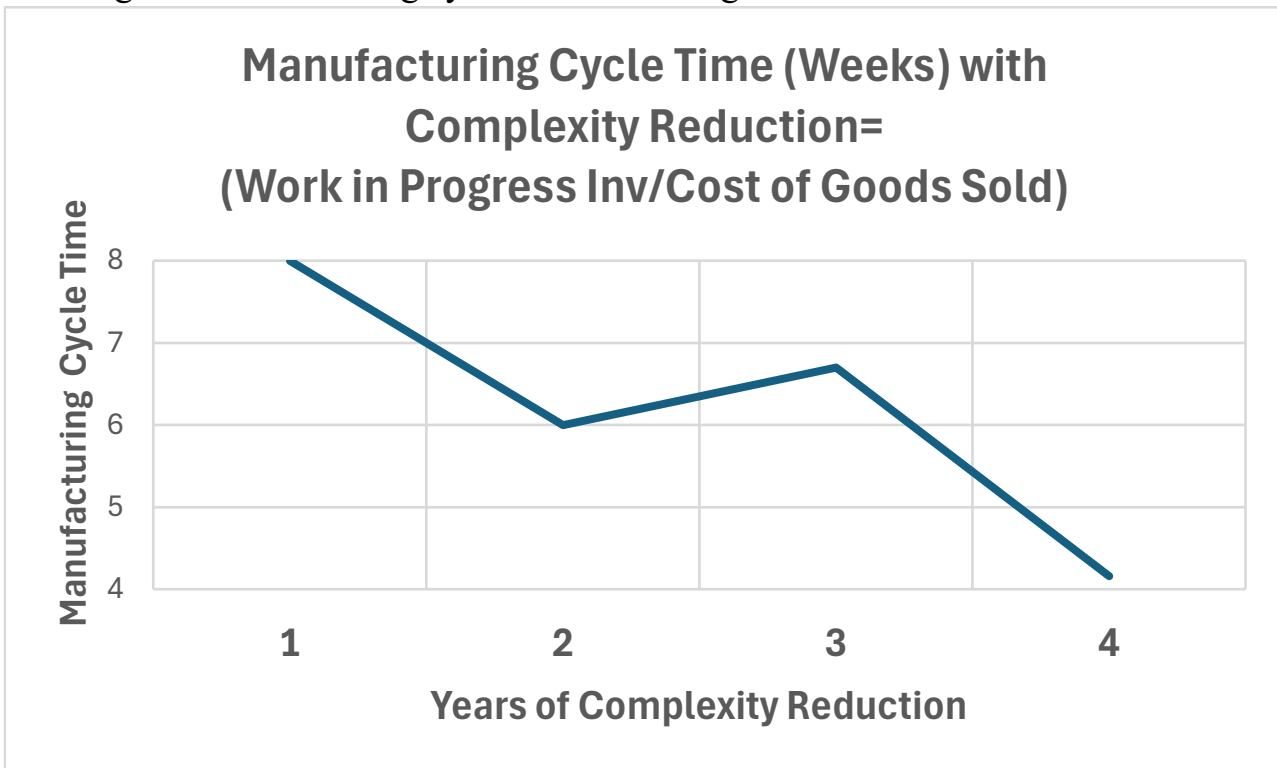


Fig 6

Low cost, high Quality and fast delivery resulted in a five fold increase in Revenue:

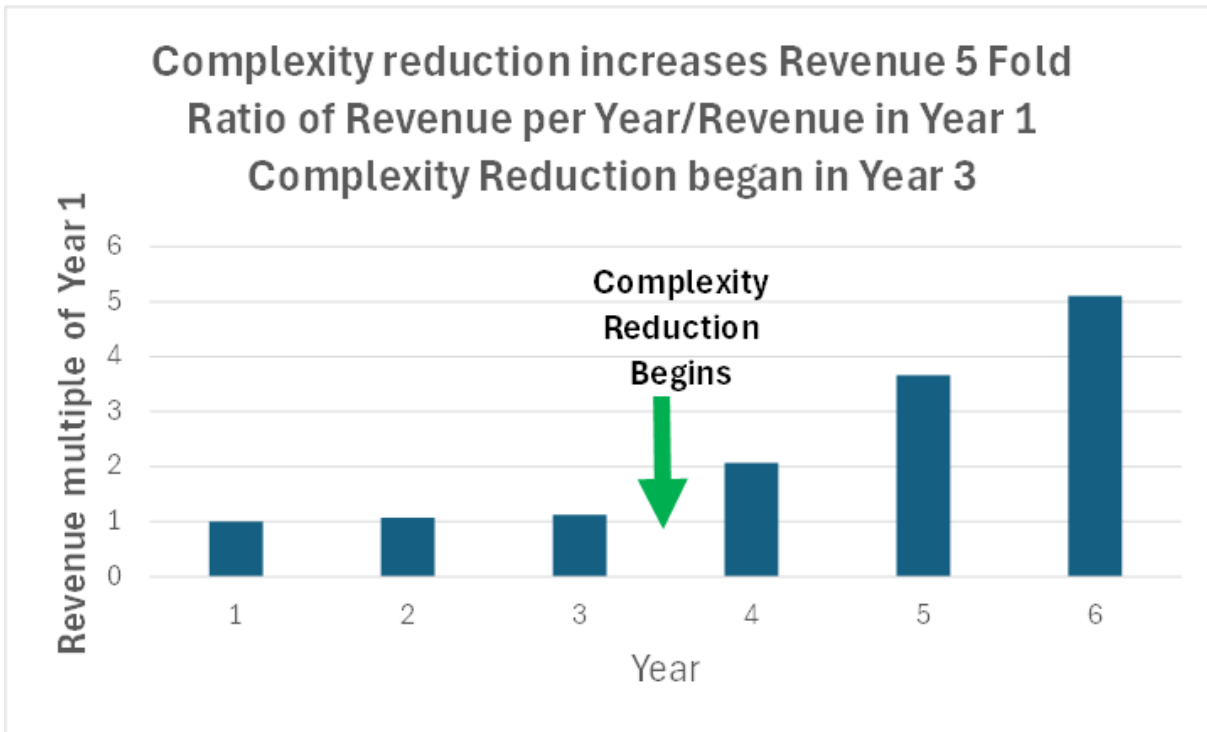


Fig 7

Conclusion: The company made a successful Public offering and was sold to Rolls-Royce. This example empirically proved that reducing the Complexity of the number of different families of products dramatically reduces Manufacturing Overhead and Direct Labor cost. The foundation of Complexity reduction was the discovery of “near misses” . However, this company only had 1000 different part numbers making the discovery of “near misses” possible by difficult and time consuming manual engineering efforts.

This laborious process of detecting all “near miss” labor can now be accomplished automatically using the AI Large Language model, even in a company with 25,000 part numbers! In the many years hiatus awaiting AI LLM, I founded the George Group, created Lean Six Sigma, and sold the company with 400 consultants to Accenture. It has been a long wait for AI LLM, making overhead reduction applicable to large companies.

*“I think that the AI Large Language Model is very exciting!
Extensive
unique Bills of Material are the bane of Factories and
Purchasing operations”*

Lou Giuliano, CEO of ITT Corp, retired

The Current Opportunity: According to Google:

“The cost of Manufacturing Overhead in most companies typically averages between 25% and 35% of revenue for manufacturing businesses”.

AI LLM can cut this cost by ~40% per Fig 1, dramatically increasing Operating Profit. AI LLM makes this goal practical for large companies. What are the actual components of costs that are eliminated by reducing the number of different families? From “Kaisha” p 80:

“Increasing complexity is the bane of a factory manager’s life. With increasing complexity comes an increased number of parts to be purchased and stocked, more material handling, inventories, diverse process flows, supervision, errors, defects, smaller batches , shorter runs and higher Quality costs.”

Notice common the use of the word “bane” in this, and Lou Giuliano’s, quotation

Reduction of Family Complexity using AI LLM

By contrast, some large companies have 25,000 different part numbers, and identifying “near misses” would be a manual impossibility. The AI Large Language Model (AI LLM) makes the discovery of “near misses” a practical solution to the Complexity problems , and hence the reduction of the number of families and overhead cost in Fig 1. The result of reducing the number of families will result in significant labor economies of scale, overhead and quality cost reductions, higher profits and happier customers and employees.

Artificial Intelligence’s Large Language Model (AI LLM) reduces the high Manufacturing Overhead and Direct Labor cost problem following AI LLM training as follows:

Step 1. An ERP download to AI LLM processing will disclose all “near misses” and hence the number of different families currently being produced in one factory per Fig 2 definition. Using Fig 2, estimate the cost saving in Manufacturing Overhead by a projected reduction in the number of current families.

Step 2. Apply AI LLM to a sample of ERP first prototype of a family of products. Increase family member part numbers to determine which are “near misses” by “fuzzy matching” to guide part number reduction and and higher level sub-system Assemblies consolidation into a more repetitive manufacturing process of products supplied to customers.

Step 3 Estimate the Manufacturing Overhead and Direct Labor cost reduction using Fig 1... Eliminate the products that cannot be members of a family

Step 4: Redesign the Bill of Material to one family of Sub-Assemblies, etc. and common highly refined Work Instructions.

Contact Mike George at mike@entropy2718 to make an appointment to discuss your Complexity application with our team.

Executive Overview of AI LLM Manufacturing by Dr. Michael Hahsler, PhD, Associate Professor of Computer Science, Southern Methodist University

Step 2 above requires “Fuzzy matching”, a technique used to find and match parts that are not identical, but have instead similar “Canonical Attributes”, those that make the part unique such as dimensions, tolerances, material type, etc. Instead of requiring an exact match, Fuzzy matching algorithms compare strings and assign a similarity score based on how close they are. It accounts for variations like Typos, Transposed letters, Added or missing letters, spellings or abbreviations, thickness dimensions, material types, etc. Matches are ranked by their similarity score, with exact matches ranked highest and less similar matches ranked lower.

Example: at IPM, the Sprague 118 P Electrolytic capacitor was used in every Inverter. Among the 8 designs, ERP listed the capacitor just as the design engineer had entered...including the following variations: 118 P, Sprague 118 P, 118, etc. Note that any variation in component part number can create a new higher level assembly and kitting part numbers making the labor economies of scale impossible! Fuzzy matching eliminates this problems. The earliest approach to solve the “fuzzy” matching problem and easiest to understand is the Levenshtein distance. Vladimir Levenshtein (1935– 2017) was a Russian mathematician and research professor at the Keldysh Institute for Applied Mathematics at the Russian Academy of Sciences in Moscow in the fields of Information Theory, Error Correcting codes, etc. He is best known for the Levenshtein distance, that measures the difference between two sequences of characters. This algorithm calculates the minimum number of single-character edits (insertions, deletions, or substitutions) required to change one string into another. It is also known as the "edit distance". Levenshtein is used in classic Natural Language processing for correction of spelling mistakes. This method was supplemented by modern techniques based on the latest versions of LLMs (<https://arxiv.org/pdf/2308.09435>). One reason for supplementing Levenshtein is because the LLM can take into account the words surrounding the word in question (i.e., the context) to find both spelling and order mistakes or versions. For example, the person who writes in cursive or shorthand often cannot read what they have written. Sometimes, three people would be trying to decipher shorthand. By looking at the context, LLM can decipher the word, numbers or order in question. Three different engineers may type the part number with slight changes in the name or

number. This is particularly the case with part number identification. Levenshtein lacks this “context capability” of LLM. However, for special applications where mistakes are in part numbers, which are not natural words, Levenshtein distance may still be the best choice. We use both methods in parallel together comparison in actual ERP part number analysis which be-deviled the manual efforts of Chapter 1 with delays and interruptions.

Large Language Models (LLMs) became available in 2023 from OpenAI, Google, Mistral AI, Cohere, etc. These AI LLMs can automatically extract part numbers, quantities and descriptions from ERP and translate into standardized formats like JSON (JavaScript Object Notation). JSON is a human-readable text format for data interchange, used to store and transmit data between a server and a web application. It was first specified in the early 2000s by Douglas Crockford. It is a widely compatible data format based on two structures: a collection of name-value pairs and an ordered list of values. Its’ simplicity makes it a de facto standard for APIs and diverse systems for communicating and processing data. (An API, or Application Programming Interface, is a set of rules and protocols that allows different software applications to communicate and interact with each other. It acts as an intermediary, defining how one application can request and receive data or functionality from another). Key characteristics of JSON

- **Human-readable:** JSON's syntax is compact and easy for humans to read and write.
- **Language-independent:** While its name and syntax are derived from JavaScript, it is compatible with many modern programming languages, including Python, Java, and others.(Note: if you talk to 20 programmers, 19 use Python!)
- **Data types:** Values in JSON can be one of the following data types: o String (in double quotes) o Number o Boolean (true or false) o null o Array: An ordered list of values enclosed in square brackets ([]). o Object: A collection of name-value pairs enclosed in curly braces ({}).
- **Common use:** JSON is commonly used for transmitting data in APIs and is crucial for modern web development.

How JSON is used

- **Data exchange:** It is used to send data between a client (like a web browser) and a server.
- **Configuration files:** Many applications use JSON files to store configuration settings.
- **Databases:** It is used in [JSON document databases](#) to store and manage operational data.

- **Logging:** JSON is also used to format log entries, making them easier for both humans and machines to read and analyze.

LLM converts the two part numbers into two different tokens and then evaluate similarity in the embedding space. In the embedding space, these tokens may be very close together if they are often used in the training data in the same context (e.g., together with similar other parts). The tokens may also be dissimilar in the embedding space, if they have a completely different functionality and just share a very similar part number. Levenshtein distance cannot distinguish between these two cases because it does not use the context around the part number. LLM models that can be used to detect inconsistent parts or evaluate part similarity are typically from the BERT

([Bidirectional Encoder](#)

Representations from Transformers) model family introduced by Google researchers in 2018.

Part number Level: Thus we can now analyze inconsistent part numbers and descriptions reducing errors and duplicate parts as the pre-requisite to Complexity cost reduction. Inconsistent part numbers, which may be textbased, also fit into this category. We suggest you start with a proof-of-concept to demonstrate how LLMs reduce complexity and cost. Experienced consultants can assist your IT professionals in this process and in the selection of LLM software.

Subsystem Level: Thus far we have shown how LLM can find both identical and near identical components by “fuzzy” searches. **The next step was to find a small number of similar standard sub-systems or modules that can then replace the multiplicity of current subsystems.** This allowed the creation of standard Work Instructions with Photos allowing for a massive reduction in hidden Direct Labor Cost, and reduced and Overhead cost related to Purchasing, Stockroom, Quality Control, Lab retest and other hidden costs. In the case of International Power Machines, ALL systems used a rectifier module to transform incoming AC power to DC power, used to recharge the battery and provide DC power to the Static Inverter modules which transformed the DC power (from either source, mains or battery) into AC power. All UPS systems had rectifier modules, inverter modules, control modules and associated power supplies. The question then is: can LLM find, not only common components in the “fuzzy” sense, but also the common subsystems which are composed of the “fuzzy components”? The answer is **YES!**

Large Language Models (LLMs) can find similar subsystems, particularly when the subsystems can be described using natural language or

have associated textual metadata. This capability relies on LLMs' ability to understand context, extract attributes, and compute semantic similarity.

How LLMs Facilitate Subsystem Similarity Analysis

- **Semantic Similarity:** LLMs use embedding models to convert subsystem descriptions (e.g., functional requirements, technical specifications, design choices) into numerical vector representations. The "distance" between these vectors in the embedding space indicates their semantic similarity. Systems or subsystems with closer vectors are considered more similar in meaning or function.
- **Attribute Extraction:** LLMs can parse technical documentation to automatically extract specific, structured attributes (e.g., performance parameters, material, function, interfaces). Comparing these extracted attributes allows for a more detailed, feature-based similarity analysis.
- **Reasoning and Context Understanding:** LLMs can process a large amount of contextual information and perform complex reasoning tasks. This allows them to identify implicit similarities that might not be obvious from simple keyword matching, such as recognizing that "eco-friendly" and "sustainable" refer to similar concepts.
- **Conclusion:** We can now identify
 1. common components using “fuzzy” analysis
 2. Common subsystems below the total system
 3. Final systems each of which is similar from an assembly and test perspective
 4. Common work instructions yielding dramatic reduction in Labor and Overhead cost e.g. IPM,

Practical Applications and Methods

- **Model-Based Systems Engineering (MBSE):** LLMs are being researched and fine-tuned to analyze requirements databases from past missions to identify semantically similar requirements for new designs, helping engineers reuse information and streamline the design process.
- **Product Matching:** In commercial applications, LLMs are used to match similar products across different platforms by comparing descriptions and images, even when the descriptions are in different formats.
- **Automated System Integration:** LLMs can analyze natural language inputs describing desired functionalities and suggest reusable service

compositions or program code for integrating different software services.

Limitations and Considerations

- For high-confidence matches, methods like human curated lookup tables or specific algorithms
- Performance in specific, complex domains can be improved by finetuning models on domain-specific data or using techniques like Retrieval-Augmented

Generation(RAG) to provide relevant context.

Conclusion: AI LLM is a powerful tool that *must* be used as a pre-condition to attain Complexity cost reduction in a Billion dollar company with tens of thousands of different part numbers. Manual methods are impractical and hence are rarely used except when obvious. It is therefore likely that excess costs exist that can be eliminated by LLM. A more detailed discussion of AI LLM is available upon request.

<https://www.bridginglocal.com/post/top-manufacturing-industries-in-the-us>

Executive References

“As COO of ITT Corporation, we selected the George Group to lead our world-wide deployment of Value Based Lean Six Sigma. The results of this effort made a significant improvement in our Operating Profit, while reducing Working Capital. I led this effort while reporting to our CEO Travis Engen. Travis subsequently became CEO of Alcan (acquired by Rio Tinto) and again retained the George Group in a similar successful capacity”. **Lou Giuliano, COO, ITT Corp**

Additional Commercial references: Caterpillar, Eli Lilly, H B Fuller etc.

US Naval Aviation:

“ Mike George’s company trained all levels of Naval Aviation from the operators, maintainers, ...to Three Star Admirals... to achieve the Single Fleet Driven Metric, that of “Aviation Units Ready for Tasking at Reduced Cost”. The results were impressive: Across 180,742 persons, 3827 aircraft, 11 aircraft carriers, and a yearly budget of over \$40.2B/year, Naval Aviation improved aircraft readiness from 30% flyable to 95% Ready for Tasking, with a reduction in cost of \$4.8B/year in operating expense across a four year period.”

Walter B. Massenburg, Vice Admiral, USN (Ret)

USMC, Army, Navy:

“80% of combat casualties were due to IED. I directed 19 George Group consultants to assist USMC, Army, and Navy personnel in increasing MRAP production in the Ladson, SC (a Charleston suburb) from 5 units per day to over 50 per day as demanded by DoD Secretary Dr. Robert M. Gates. This was achieved in 5 months - in the same facility with the same number of workers! This effort resulted in a significant reduction of death and injury to our Soldiers, Sailors, and Marines (Navy MRAP video available). The combat comparison of HMMWV fatalities to MRAP survival in before/after photos depict MRAP combat effectiveness.”
Nick Kunesh, former Deputy Assistant-Secretary of the Navy



HMMWV(fatalities)



MRAP(Soldiers exiting unharmed)

Captain
Herman
Shelanski,
Mike George,
DASN Nick
Kunesh



Review of the Lean Six Sigma process aboard the USS Harry S Truman CVN-75

Contact Mike George at mike@entropy2718 to make an appointment to discuss your Complexity application.