Automotive Component Product Development Enhancement Through Multi-Attribute System Design Optimization In an Integrated Concurrent Engineering Framework

by

Massimo Usan

M. S. Aeronautical Engineering (1992)

Polytechnic of Turin

Submitted to the System Design and Management Program
in Partial Fulfillment of the Requirements for the Degree of

Master of Science in Engineering and Management

at the

Massachusetts Institute of Technology

February 2005

Signature of Author__________________________________

Massimo Usan
System Design and Management Program
February 2005

Certified by______________________

Daniel Whitney
Thesis Supervisor
Center for Technology, Policy & Development

Certified by__________________________________

Olivier de Weck
Thesis Supervisor
Engineering Systems Division

Automotive Component
Product Development Enhancement
Through Multi-Attribute System Design Optimization
In an Integrated Concurrent Engineering Framework

by

Massimo Usan

Submitted to the System Design and Management Program
in Partial Fulfillment of the Requirements for the Degree of

Master of Science in Engineering and Management

February 2005

ABSTRACT

Automotive industry is facing a tough period. Production overcapacity and high fixed costs constrain companies' profits and challenge the very same existence of some corporations. Strangulated by the reduced cash availability and petrified by the organizational and products' complexity, companies find themselves more and more inadequate to stay in synch with the pace and the rate of change of consumers' and regulations' demands.

To boost profits, nearly everyone pursue cost cutting. However, aggressive cost cutting as the sole approach to fattening margins results invariably in a reduction of operational capabilities which is likely to result in a decline in sales volume that leads to further cost reductions in a continuous death spiral.

Long-term profitable growth requires, instead, a continuous flow of innovative products and processes. The focus should be, therefore, shifted from cost reduction to increased throughput. Automotive companies need to change their business model, morphing into new organizational entities based on systems thinking and change, which are agile and can swiftly adapt to the new business environment. The advancement of technology and the relentless increase in computing power will provide the necessary means for this radical transformation.

This transformation cannot happen if the Product Development Process (PDP) does not break the iron gate of cycle time-product cost-development expenses-reduced product performance that constrains it. A new approach to PD should be applied to the early phases, where the leverage is higher, and should be targeted to dramatic reduction of the time taken to perform design iterations, which, by taking 50-70% of the total development time, are a burden of today's practice.

Multi-disciplinary Design Analysis and Optimization, enabled by an Integrated Concurrent Engineering virtual product development framework has the required characteristics and the potential to respond to today's and tomorrow's automotive challenges. In this new framework, the product or system is not defined by a rigid CAD model which is then manipulated by product team engineers, but by a parametric flexible architecture handled by optimization and analysis software, with limited user interaction. In this environment, design engineers govern computer programs, which automatically select appropriately combinations of geometry parameters and drive seamlessly the analyses software programs (structural, fluid dynamic, costing, etc) to compute the system's performance attributes. Optimization algorithms explore the design space, identifying the Pareto optimal set of designs that satisfy the multiple simultaneous objectives they are given and at the same time the problem's constraints.

Examples of application of the MDO approach to automotive systems are multiplying. However, the number of disciplines and engineering aspects considered is still limited to few (two or three) thus not exploiting the full potential the approach deriving from multi-disciplinarity.

In the present work, a prototype of an Enhanced Development Framework has been set up for a particular automotive subsystem: a maniverter (a combination of exhaust manifold and catalytic converter) for internal combustion engines. The platform, adequately simplified to cope with the project constraints, features a bus architecture where the different analyses modules can be excluded and included with minor effort. Commercially available software is used, with some customization for the particular use. Particular emphasis is placed on the breadth of the engineering disciplines considered – which include fluid dynamics, pressure waves propagation, thermal management, vibrational behaviour and mass properties – and on the inclusion of business elements, in the form of a parametric cost model.

The development process executed in the new framework, benchmarked with current practice, resulted in a reduction of 75% in development time and cost and projections of 85% reduction are made for a full-functional tool running on adequate hardware. In addition, thanks to the possibility to evaluate many different maniverter configurations, an innovative design solution with better performance and greatly lower cost was identified.

The efficient interface management coupled with the 24/7 working capability of computers let us think that the application of "Product Development Computerization" could reasonably lead to 50% reduction of the development cost and budget of many automotive systems, in addition of delivering products with enhanced performances. Benefits are expected to be higher for complex systems where the diverse conflicting requirements narrow the trade space and for products that feature high levels of similarities from one application to another.

The deployment of such an environment improves the change management capability of a company, enabling new levels of business agility and triggering a virtuous spiral. Faster time to market, lower development costs and more cost efficient products can result in an increase of market share and profit margins. Engineering resources, released by the burden of time-consuming modelling, routine CAE analyses and continuous rework in a schedule pressure environment, can focus on the delicate decision-making phase and more proficiently devote their time to research for next-generation products, to explore new business segments and to innovation. These projected efficiencies more than offset any cost for the development of the framework and any additional software or hardware that may be necessary.

Yet, its successful implementation requires winning some challenges. The most critical are:: 1) overcoming the resistances of engineers that "like doing things by hand"; 2) creating in the engineering community an open mindset that is ready to accept possible radically innovative solutions found by optimization algorithms scavenging traditionally unexplored areas of the design space; 3) breaking the traditional organizational divisions by functions to establish a network of knowledge management teams.

No global automotive company, especially Tier 1 suppliers, can afford to wait. In fact, as an organization approaches irrelevance, latitude for constructive action diminishes. Times call for immediate action: we need to invest in research and development so we can continue to prosper and grow. The winner will be the one who has a clear vision of the final agile state, starts earlier on the journey to achieve this vision, and implements it piece by piece.

ACKNOWLEDGEMENTS

In the beginning was the Vision. Then came the Goal and with it the challenge to achieve it. Achieving was then finding the best path to reach the goal: the higher the summit, the harder the climb. Descending from the goal, I have worked from whole to part, dividing the total scope of work into fragments and working on them one by one and in the end, bringing them all together.

At the beginning of the project, many warned me that the summit was way to high; some, illuminated by the experience and seniority, were even more explicit and said that achieving the goal was nearly impossible, given the working conditions. I want to thank them because they were right: my youth and my enthusiasm drove me into a rough ocean. After having succeeded and approaching a safe harbour, I can realize how challenging the task was. Only, the great teamwork and the combination of long-sighted minds and outstanding skills make possible to reach this big result.

In addition to my wonderful Thesis Advisors, I would like to thank firstly my great Supervisors that have sponsored and supported me throughout the SDM effort up to the Thesis stretch: Silvio Angori, Jack Grace, Massimo Impavidi and Winfried Willeke.

In addition, my never-ending gratitude for their contribution and help goes to very many companies, universities and professionals that have worked together with me have to make the Vision come true:

- ArvinMeritor: Mohamud Aden Awed, Daniel Esteban, Valter Gorrea, Dorothee Hartmann, Juan Marcos, Youssef Mochkaai, Edgar Neuhaus, Martin Roll, Ian Rutherford, Jurgen Schaefer, Klaus Spindler
- Autostudi S.r.l: Carlo Angiono, Bruno Cuzzoni, Antonino di Franco, Piero Taruffi
- AVL: Johann Krammer, Andreas Muenzer, Ivo Prah, Maik Suffa
- Centro Ricerche Fiat: Paolo Dondo, Luciano Pera, Cristina Randazzo, Maria Luisa Sanseverino, Marco Zanzola
- Engineous Software Inc.: Jeff Bond, Mary Carr, Jack Cofer, Mike Sheh, Holger Wenzel
- Fiat-GM Powertrain: Giovanni Basso, Luigi Pilo, Luigi Strazzullo
- Gamma Technologies: Marilyin DeVries, John Silvestri, Jinny Tremmel
- MSC.Software: Cinzia Chillemi, Luca Fattore, Valerio Giorgis, Mauro Parodi, Fabrizio Trentini, Ernesto Troncarelli
- Massachusetts Institute of Technology: Bill Foley, Pat Hale, Ted Hoppe, Denny Mahoney, Donna Rhodes, Thomas Roemer, David Wallace
- Pennsylvania State University: Timothy Simpson
- Politecnico di Torino: Vilma Boaglio

- UGS PLM Solutions: Tavia Carson
- University of Strathclyde Business School: Valérie Belton

A big thanks goes also to all my friends that in many ways have accompanied me in this long journey. They took out part of the pressure when it was too high for me to resist, and fed me with energy when I was short of it.

And, at last, I want to wholeheartedly thank my beloved parents who have helped me like no others in these two years of the System Design and Management Program and that have shared with me every single moment of this Thesis effort.

This achievement is dedicated to them, to their force and to their love.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

Acronym	Meaning
AFM	Abrasive Fluid Machining
ANSI	American Standard National Institute
BSFC	Brake Specific Fuel Consumption
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CAS	Computer Aided Simulation
CAx	Computer Aided x activity
CEO	Chief Executive Officer
CFD	Computational Fluid Dynamics
COA	Center Of Application
COC	Center Of Competence
CPU	Central Processing Unit
DI	Direct Injection
DoE	Design of Experiment
ECU	Engine Control Unit
EDF	Enhanced Development Framework
FEA	Finite Element Analysis
FEM	Finite Element Method
FLOPS	FLoating point Operations per Second
FM	Functional Manager
FTE	Full Time Equivalent
FMVSS	Federal Motor Vehicle Safety Standards
GA	Genetic Algorithm
GUI	Graphical User Interface
HPC	High Performance Computing
HW	Hardware
IC	Internal Combustion
ICE	Integrated Concurrent Engineering
IGES	Initial Graphics Exchange Specifications
IPT	Integrated Product Team
IRR	Internal Rate of Return
ISV	Independent Software Vendor
KBD	Knowledge-Based Design

Acronym	Meaning
KBE	Knowledge-Based Engineering
KF	Knowledge Fusion
M&S	Modeling & Simulation
MDA	Multi-disciplinary Design Analysis
MDO	Multi-disciplinary Design Optimization
NPV	Net Present Value
NVH	Noise Vibration and Harshness
OEM	Original Equipment Manufacturer
PD	Product Development
PDP	Product Development Process
PLM	Product Lifecycle Management
PM	Program / Project / Product Manager
PMU	Profit Margin Unit
QED	Quick Exhaust Design
RFQ	Request For Quotation
SI	Spark Ignition
STEP	STandard for Exchange of Product model data
SUV	Sport Utility Vehicle
SW	Software
THK	Thickness
TQM	Total Quality Management
TTM	Time To Market
UG	Unigraphics

1. INTRODUCTION

1.1. Motivation for the Work and Thesis' Value Proposition

1.1.1. Current Status of Automotive industry and Future Outlook

"The industry's massive volatility represents the shudders of a dying business model that has created an era of high sales volume and negligible profits... Today we have high sales volume and low profitability. We have to change. My view is the change is under way".

David Cole – President of the Center for Automotive Research (Automotive News, 01/15/2003 – for the full presentation, see [1]).

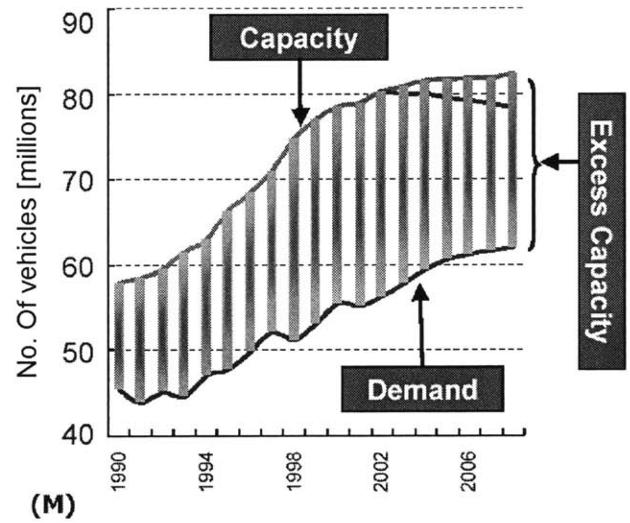


Fig. 1: Global Yearly Light Vehicle Capacity and

As scaring perspective as it may appear, this is indeed the sharp depiction of the profitless prosperity that the automotive industry is facing these years, with stock valuations and cash available for re-investment that are at historic lows.

Industry analysts and observers (for some reviews see [2]) identify overcapacity (20-25%) and high fixed costs the main drivers of this downturn, Fig. 1 [3] . Furthermore, within the next ten years, they predict consolidation to three or four global OEMs and a 25% consolidation of capacity and companies across the supply base. Accenture, for example, estimates that the number of Tier-One suppliers will decline to around 4,000 by 2010 [4] , Fig. 2.

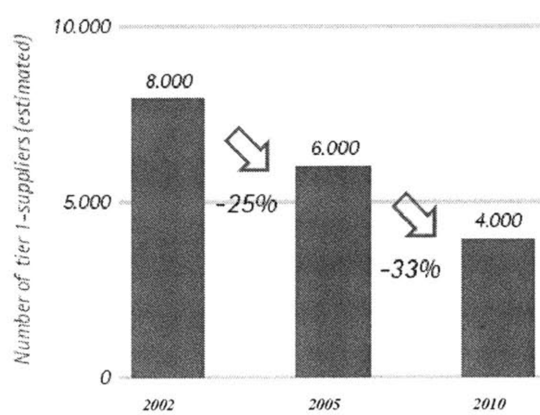


Fig. 2: Continuous supplier consolidation

Automotive companies are strained by multiple tensions.

Data reveal that since the 1960's the number of basic vehicle segments has grown from 4 to more than 15. The industry's top five manufacturers alone are expected to introduce nearly 160 new models and facelifts in the U.S market from 2003 through 2007 (Fig. 3 [5]).

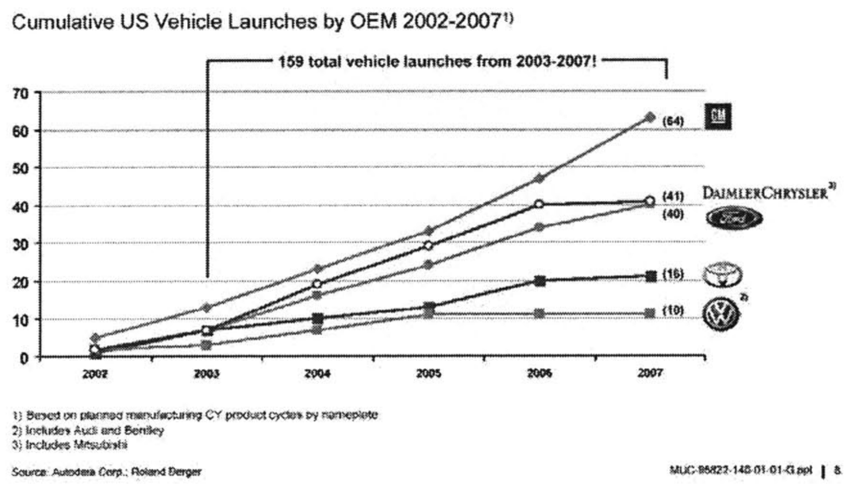


Fig. 3: Proliferation of models – heavy burden on development resouces

The number and complexity of new-model features also continues to climb. The electronics cost content of an average car in the 1970's was less than 10 percent. It's expected to top 40 percent by the year 2010. Additionally, regulatory requirements also have a major impact on product development. Rising fuel economy, safety and environmental standards put additional pressure on OEMs product development.

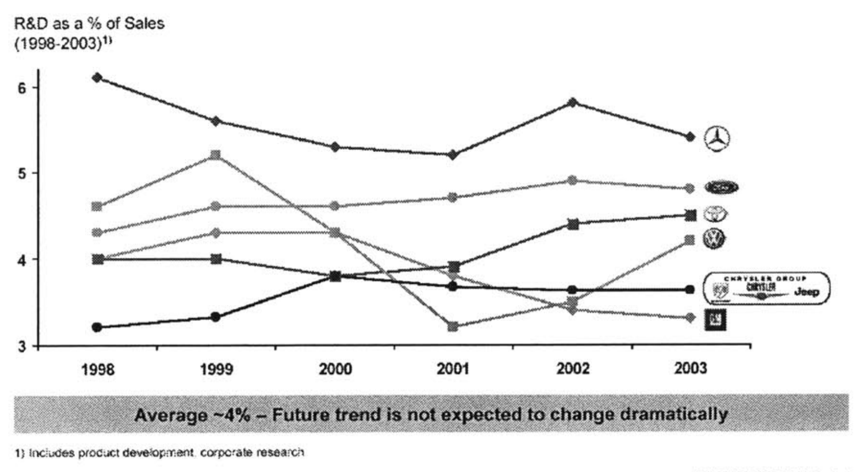


Fig. 4: R&D budgets remain flat

While costs and complexity have risen sharply in recent years, the price of an average vehicle has remained virtually unchanged since 1993. In parallel, since 1998, R & D budgets as a percent of sales at five of the industry's top OEMs have remained virtually unchanged at an average of 4% of sales, Fig. 4.

However, pressures to reduce costs and to increase systems' contents are not the only forces that shake automotive companies. Other forces are challenging the status quo and pushing for breakthroughs. The first is the reduction in the time to market (TTM).

Over time, the product development lead-time of car manufacturers has been constantly shrinking (Fig. 5), linked to shorter and shorter TTM. From the 60 months of about ten years ago, to the current range of 46-36 months, to the stretch target of 18 months touted in a recent announcement by GM [6] . Drawing a TTM data a 'speed to market curve', Fig. 6, a technology S-curve emerges that would indicate that, in the next ten years, the car industry will move at a much faster pace than today.

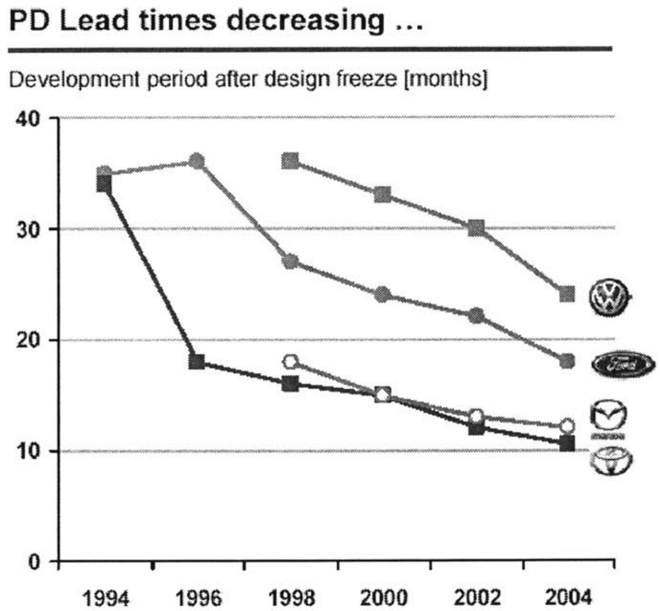


Fig. 5: PD Lead time constantly decreases

In addition to increasing operational margins, coping with systems of increasing complexity and reducing the development time, there is an increasing need for flexibility, the possibility to quickly change from one design to the other with different performance metrics to adapt to market or company strategy changes.

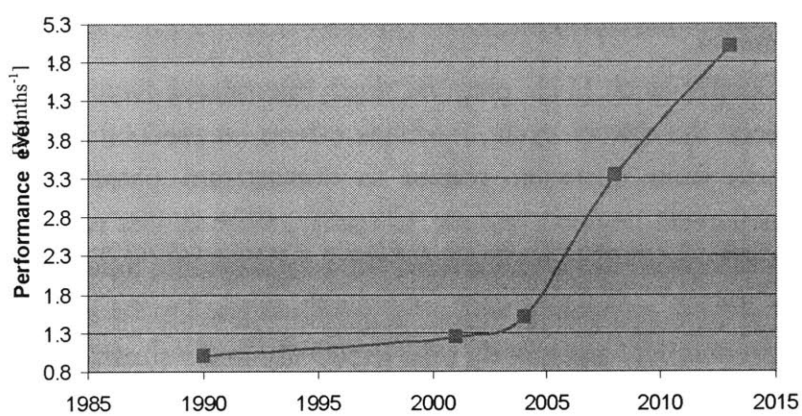


Fig. 6: Speed to Market

1.1.2. Doing more with less: the risk of cost cutting

> *When a business's income and expenses are out of balance, it's a symptom. You can no more cure your business by cutting expenses than you can cure a fever by dropping the patient in an ice-water bath* [7] .

Up to now, in order to sustain their business in this context of overcapacity, high fixed costs and intense competition both OEMs and suppliers' have reacted (somewhat irrationally) asking their suppliers for immediate cost reductions.

Cost reduction, however, has several negative implications. First of all, it further squeezes companies' margins and increases the risk of supply. Secondly, as a report of AMR Research points out [8] , targeting cost to improve margins throws manufacturers into what General Electric's Jeff Immelt calls "commodity hell"- where companies find themselves competing on price alone. But, what's worse, it might put the company in a state of permanent emergency with quality reduction and long development leadtime at no actual appreciable reduction in product development costs. This is due to the occurrence and to the subsequent spreading of a pathological organizational dynamic phenomenon called fire fighting [9] [10] , which is defined as the unplanned allocation of resources to fix problems discovered late in a product's development cycle.

The understanding of fire fighting is intimately linked to the understanding of project dynamics.

Cooper et al. [11] pinpoint three interrelated factors related to the dynamics of a single project: the rework cycle, feedback effects on productivity and quality impacts, and knock-on effects from upstream phases to downstream phases. Customarily, conventional project management neglects rework. In reality, more or less rework emerges in any project. At least part of rework lies undiscovered for a considerable time, and after its discovery, it is rushed to completion, competing with other work assigned to the specialists in question. Feedback effects on productivity and quality refer especially to the situation where there is managerial corrective action after deviation from the plan. Bringing more resources, using overtime or exerting schedule pressure will usually reduce productivity and quality. Reduced quality will, in turn, lead to more rework. When a project consists of several phases, the availability and quality of upstream work can impact the productivity and quality of downstream work. Thus, the rework cycles and feedback effects in one phase extend their influence to the next phases. When multiple projects are running, as in a normal company, the problems on one project knock then onto the following project in a never-ending spiral.

Repenning et al. [10] demonstrate the existence of a tipping point above which the negative effects of rework, productivity and quality loss and knock-on dynamics amplify and spread whereas they die out below this point. In analogy with models of infectious diseases where the tipping point represents the threshold of infectivity and susceptibility beyond which a disease becomes epidemic, in product development systems there exists a threshold for problem-solving activity. When crossed, fire fighting will emerge, grow and spread rapidly from a few isolated areas to the entire development system, due to a sort of domino effect from one project to the other.

Fire fighting is a self-reinforcing phenomenon: once the system has entered the fire-fighting zone, escape is difficult. As illustrated by Brooks' Law (1995) about the Mythical Man-month [11] , in fire-fighting situation the progress is not proportional with the effort. This is because resource utilization efficiency drops. As a result development cost rise and leadtime lengthen, which induce management to further attempt to increase the pressure. The fire-fighting syndrome persists because few actions can push the product development system back over the tipping point into a virtuous cycle of improved process execution [10] .

Moreover, as Repenning and Henderson point out, when the company is in a fire-fighting mode, all the resources are utilized to fix problems and to manage the growing rework, which, being immediate needs, are perceived as more urgent and compelling. Resources spent in fire-fighting are drained from medium-long term innovation and research, with the result of emptying the product pipeline (Fig. 7, [13]) and undermining effective management of the development system [14] . Extreme time pressure not only stops medium and long-term research but kill also vital incremental innovation. The need to freeze the design avoiding high rewards / high-risk paths pushes developers to follow known development routes and ending in products of standard performances

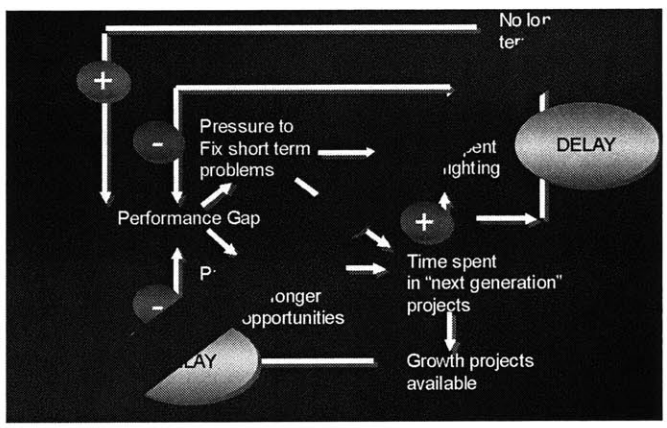


Fig. 7: Overload may destroy long term peformance

Why fire-fighting occurs? And why, resources have such an important role? One contributory cause for this situation, discussed by Dörner [15] , is the cognitive limitation of humans as decision-makers and their inability to cope with complex and dynamic situations. Dorner isolates four factors. First, the question is about complexity, the existence of many interrelated variables. Second, we have to deal with dynamic systems. It is not enough to manage the system a single moment, but over time. Third, the system is to some extent intransparent; we cannot see all we want to see. Fourth, ignorance and mistaken hypotheses prevail. We usually do not know all relationships between the variables. Dörner goes on to explain the generic causes of mistakes people make when dealing with complex systems:

- Slowness of thinking
- Small amount of information that can be processed at any one time
- Limited inflow capacity of the memory
- Tendency to protect the sense of competence
- Tendency to focus on the immediately pressing problems.

Resources limitation created by cost cutting, if not adequately compensated by more powerful tools that allow the correct management of the dynamic complexity, exalts the negative modes of project dynamics and lowers the tipping point, thus making easier for fire fighting to burst.

In addition, the "better-before-worse" behavior created by fire fighting coupled with basic human tendencies for immediate rewards and intuitions further leads decision makers down the wrong path.

1.1.3. Beyond cost cutting: new paradigms for product development

> *"When you are face to face with a difficulty, you are up against a discovery."*
>
> – Lord Kelvin

To stay competitive in a changing market, automotive companies need to continuously increase their operational efficiency. This means increasing the added value of resource utilization in engineering, manufacturing, and distribution processes – both internally and along the supply chain.

A major leverage factor, as Liz Lempres, director in McKinsey's Boston office, notes [16] , is the process, which in most companies has become an inflexible sequence of activities, like a production line. Because it is inflexible, it is disconnected from marketplace changes that may determine the fate of new products. The solution is to inject more customer-related information into the process and to make it flow better. By transforming a rigid process into a more dynamic and information-based one, companies can quicken the pace of development and improve a product's odds of success.

However, to do so, they will have to implement basic changes in the way they make and time product-development decisions. By improving the quality, timing, and synthesis of information throughout the development cycle, companies can free themselves from prescheduled project time lines and formalized process steps and manage their resources and work flows more flexibly. They can keep their product options open longer, act on market information later, and reduce the delays, bottlenecks, rework, and wasted effort inherent in today's assembly-line product-development process.

Both managers and scholars increasingly realize the central role that product development plays in creating competitive advantage. Product Development is governed by the classic four forces of cycle time, product cost, development expenses and reduced product performance or feature set, Fig. 8 [17] that trade off with each other. Improving PD should aim to achieve, at the same time, lower time to market, competitiveness through innovation and to lower costs due to change, either because of the need to fix failures or because of changing customer requirements.

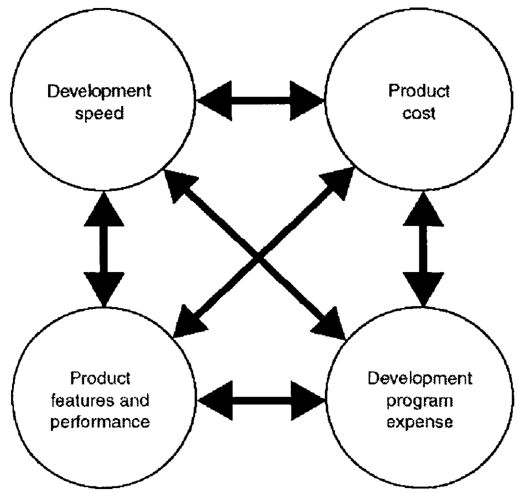


Fig. 8: the iron gate of Product Development

Product-development cycles have improved over the past two decades by following a more disciplined and rigorous process. However in the recent years, the pace of improvement seems to have levelled off, and companies are unable to meet the rising demand for better and more frequently launched new products aimed at narrower customer segments.

The conflict of opposing forces, the compelling need of nowadays to do more with less, sets the premises for growth, change, and progress. For example, new approaches to mass customization via product platforms are pursued. We believe that the auto industry is preparing for a huge technology revolution over the next few years, especially in PD.

Significant, permanent changes with holistic structural interventions and appropriate tools are needed to create a fire resistant product development system. Novel approaches are expected to have bigger benefits if:

- They are applied to the early phases where the leverage is higher,
- They dramatically reduce the time taken to perform design iterations.

In what follows each of features will be briefly discussed

The crucial role of the early phases of PD

Experience of manufacturers in many industries has shown that 80% of the total time and cost of product development are committed in the early stages of product development, when only 5% of project time and cost have been expended (Fig. 9 [18]). This is because in the early concept stages, fundamental decisions are made regarding basic geometry, materials, system configuration, and manufacturing processes [19] . Further along in the cycle, changes get harder to make. Essentially, the time and cost to correct problems increase ten-fold with each step of the product development cycle [20] : concept definition, detailed design, prototype manufacture, prototype testing, and production, Fig. 10. So a relatively minor change that would have cost a few dollars if made in the concept definition stage, could end up costing hundreds of thousands of dollars in the production stage, or millions if flawed products are shipped.

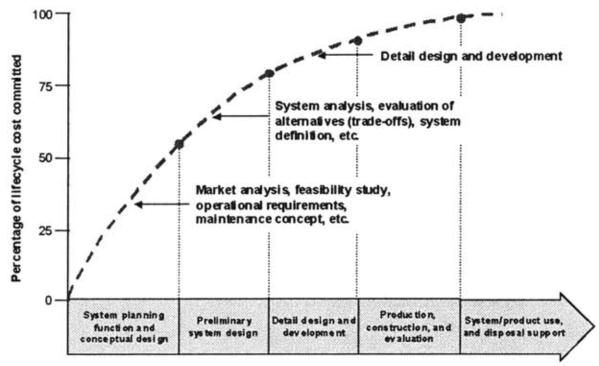


Fig. 9: Lifecycle cost committed Vs. Project Phase

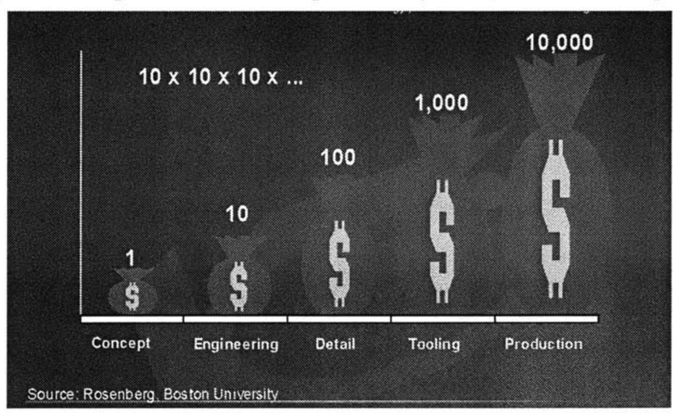


Fig. 10: The cost of change

Despite the crucial importance of this delicate phase, surprisingly in the industry it is not given the attention it deserves. This has, as a consequence, a lot of failure-trial-fix loops and development costs dominated by failure recovery actions (Fig. 11 - [21]).

Consequently any PD improvement will be more beneficial if allows better exploration and resolution of the system's trade-offs in the early stages of design definition.

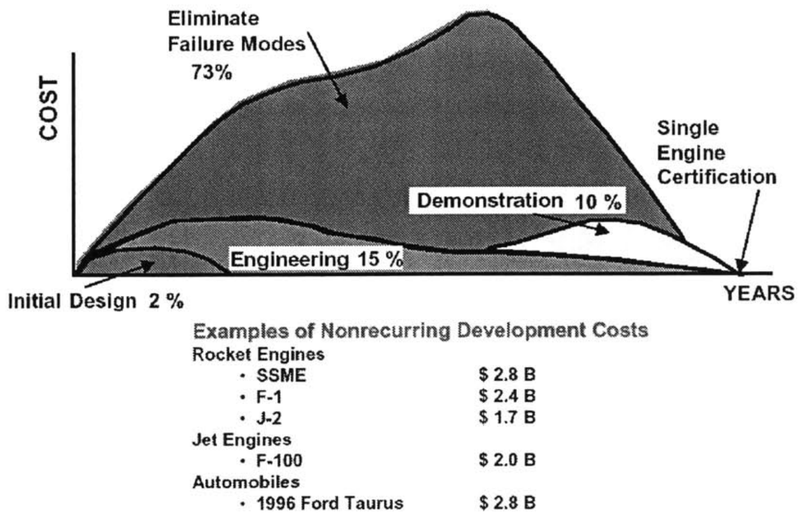


Fig. 11: Product Development Costs: dominated by rework

Design iterations

Product Development is inherently iterative and this property makes it particularly difficult to keep it under control. Delays in schedule and budget excedances are common. The combination of the interdependency structure and dynamics delays cause development problems (issues) believed to be solved (closed) to re-appear (reopen) at later stages of development.

As an anonymous product development manager at an automobile manufacturer said, *"We just churn and chase our tail until someone says that they won't be able to make the launch date."* [22]

Iterations occur because of inherently conflicting trade-offs for which it's very difficult to find a balance and they usually represent a major portion of the product development leadtime and cost.

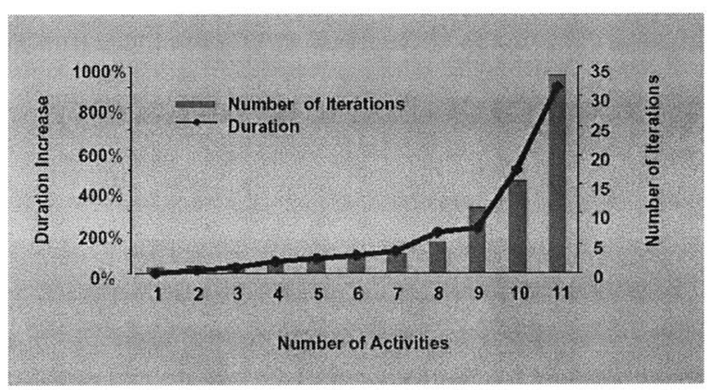


Fig. 12: Iterations grows exponentially with product's and organizational complexity

Iterations duration increases exponentially with the number of interrelated activities and, consequently, with product / organizational complexity (Fig. 12, [23] , [24]).

The modest effort that is usually deployed in the concept design phase leads to a limited number of design iterations whereas they could be more rewarding. Consequently, the design space is poorly explored and / or some design aspects are neglected. This poor job in the concept design sets the seeds for the rework that will surface downstream in the project, causing longer and more expensive design loops.

Hence, an improvement in PD should allow performing more design iterations, particularly in the concept and preliminary design phase.

CAx tools are extensively used in these early stages to virtually simulate the performance of the product. Impressive advances are being made in the application of M&S in design. However, M&S is believed to still be in its infancy. M&S tools suffer several limitations, among which we can enumerate the following:

- They are complex, and each tool requires a specifically designed data set to enable its operation.
- Creation of the data sets and models that enable accurate simulation is costly and time-consuming.
- Difficulty of calibrating the models against real experimental data
- Interoperability is a huge issue
- The models that now exist are in most cases incomplete, in that they do not always support good decision processes with accurate bounding of risks and uncertainty.
- Current models are unable to allow the evaluation of design alternatives across different disciplines and that prevents a great opportunity for cost effective, robust designs that meet all product requirements.

The new PD process should also overcome these limitations, particularly interoperability.

1.1.4. The Enhanced Product Development Framework

Given this scenario of strong forces that push for cost and leadtime reduction and to increased flexibility, having identified the concept / preliminary design as the highest returns phase where to intervene and having isolated an important source of waste, namely design iterations, we can now proceed to formulate a roadmap for a solution.

We envision an approach to product development with superior performance the one which, assuming a system's perspective, considers the product in its all performance dimensions, minimizes the disruptions by establishing optimum communication channels along the interdependence lines and allows fast execution of design iterations.

This Enhanced Development Framework is therefore a development process that has to, simultaneously, take care of all targets and constraints and deal with several different analysis disciplines in an integrated environment for concept evaluation, optimisation, verification, and multiphysics integration. It should also enable the developers to reach overall best solutions and very closely survey the different conceptual limitations.

We argue that Multi-disciplinary Design Optimization appears well positioned to become the core of this environment. To exploit its full potential, MDO needs to be executed in an Integrated Concurrent Engineering platform where seamless data transfer among simulation packages occurs. The greater the number of product performance dimensions and the deeper is the interrelation among them, the higher the advantages expected of this methodology will be.

The EDF with its high performance ICE platform is projected to enable a very fast execution of design iterations in the early phases of PD. Consequently, many solutions can be

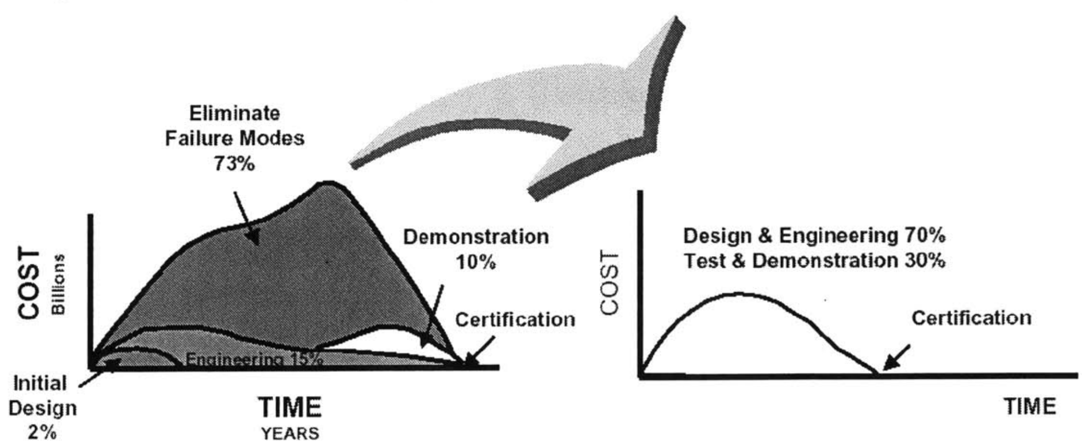


Fig. 13: Computerized Engineering Vision

investigated at a relatively low cost. In addition, the MDO approach leads to evaluate each solution from all the relevant performance attributes standpoints. The combinations of the two factors, i.e. comprehensive design exploration and holistic perspective, is expected to reduce dramatically the cost of rework, thereby leading to product's value maximization, Fig. 13 [21]

Automakers search for ways to increase their speed, agility, situational awareness and ability to innovate and, in consequence, improve their competitive position. Following our industry and PDP analysis, MDO in an ICE framework is viewed and proposed as a mindset and a methodology to cope with the challenging demands that the market is imposing on automotive manufacturing firms, helping the transition in the product-development and engineering processes from a "test, analyze, and fix" paradigm toward an industry-ideal "design-right-first-time" for value maximization.

1.2. Objectives

Given the previously illustrated value proposition, the present work has the resulting set of goals:

- Explore Product Development improvement research
- Investigate the application of the Multi-disciplinary Multi-objective development paradigm in the automotive industry
- For a specific system, an exhaust system maniverter for passenger cars, evaluate potential benefits and issues of the application of a MDO approach and of a system perspective in engineering simulation in terms of:
  - Cost reduction
  - Product development lead time reduction
  - Flexibility
  - Innovation (non-traditional solutions)
- Perform a cost benefit analysis to verify that a project targeted to develop such a new framework is characterized by a solid business case
- Outline an implementation roadmap in a business setting investigating the technical and organizational challenges involved.

1.3. Operative Approach

1.3.1. Introduction

To prove the feasibility and robustness of the MDO approach in an ICE platform and evaluate the challenges of the implementation of the EDF as well as the potential rewards, a prototype of the proposed environment is built around a specific application. As a particular system an exhaust system manifold or, more precisely a maniverter (manifold+converter) for passengers cars, is chosen (see Fig. 14 for an example). The choice of this system is driven by multiple reasons. First and foremost, an exhaust maniverter is intrinsically a highly multi-disciplinary system: in fact, its design is governed by fluid dynamics requirements (pressure losses, engine tuning), heat management issues (gas temperature drop, radiated heat) as well as structural constraints (resonance frequencies, thermal stresses, vibration induced stresses), not to mention, packaging, manufacturability and cost. Second, the system is of a favourable medium-size complexity and therefore amenable to be managed in a limited resources-limited budget framework such as of a thesis project and yet it is not too simple to make the development trivial. Last, but not least, this is the type of systems which the writer has first hand experience and can have direct access to related information.

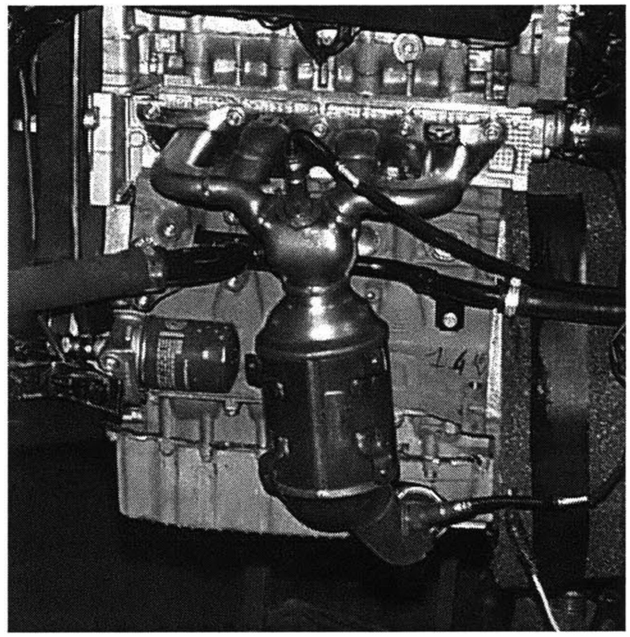


Fig. 14: example of a maniverter system

In building the prototype EDF, several guiding principles have been followed:

- The ICE platform has been set-up with a modular structure with minimum interaction among the different software packages. This architecture ensures, at least conceptually, independence from the individual software and has several benefits:
    - Possibility to replace easily one software package with another one, depending on the company strategy or available skills
    - Possibility to use the best software package for the specific task
    - Possibility to add at any time additional modules to widen the scope of the performance attributes evaluation.

- Single platform / single node execution. To mitigate interoperability issues all the software programs have been installed on a single computational platform. In addition a laptop has been chosen to favour the data and information exchange among the different people that contributed to the effort and that worked in different geographic locations. Since the chosen platform is not capable of computationally intensive calculations, analyses that require high computing power have been either discarded or low computational cost alternatives pursued. As an example, CFD, which is usually utilized in the development of maniverter to compute the pressure drop and flow distribution in front of the catalyst, was soon excluded from the analyses suite because of its computational requirements. As a competitive alternative, a 1-D fluid dynamic code was used.
- Emphasis is placed more on the characteristics of the process as a whole than on the accuracy of the results. The prototype environment created in the present work is intended to be a demonstrator of the advantages of the MDO approach and of the ICE platform and to verify the business / organizational implications of a potential implementation. Therefore breadth was emphasized rather than depth. Absolute accuracy of CAE results in the respective domains was not pursued, but care was only put that the trend showed by the calculus was in the right direction. For the same reason, the geometry was not modelled with all the details to keep the number of parameters within a manageable range.
- MDO as a business tool. Taking a managerial perspective, the demonstrator was verified as a tool to improve some crucial business processes such as trade-off evaluation, decision-making and customer relationship
- Incremental approach. The framework was built incrementally to keep the complexity always at a controllability level. Complexity was raised progressively along two dimensions: application complexity and framework complexity. The complexity of the application was raised from a simple pipe with fixed centreline and variable cross sections, to a simplified maniverter with constant and circular cross section but variable piperun and finally to a maniverter system with several design variables.

1.3.2. Methodology

Given the project goals and the system to be analyzed, the following steps were taken:

- Scope definition
- Architecture / platform layout and essential pre-requisites list
- Hardware / software requirements
- Partnership building
- Hardware / software acquisition
- Literature search
- Project execution, i.e. building the EDF
    - Simple pipe
    - Full manifold
- Cost-Benefit Analysis
- Implementation layout:
    - Skill set re-definition
        - Project Managers
        - Functional members
    - Change management

The following paragraphs briefly describe each of those steps.

Scope Definition

Even though this phase was iterative, a preliminary list of analyses and performance aspects to be evaluated has been laid out at the outset, using the experience in product development of the writer. Among those that were identified from the beginning, there are:

- Mass characteristics
- Structural behavior
- Fluid Dynamic and thermal behaviour
- Emissions performance
- Engine Performance
- Cost

In subsequent phases, this list was then refined and tailored to the available options. Structural behavior was then restricted to the eigenfrequencies, excluding, on the other side, thermal induced stresses and forced vibration induced stresses. Fluid dynamic section was de-scoped from a full 3D Computational Fluid Dynamic (CFD) analysis because of too much computational expense, while it was enhanced by the calculation of the temperature in front of the catalyst, in addition to the calculation of the pressure drop. These restrictions are not considered fundamental limitations of the platform. Given the modular architecture (see 3.10) of the ICE platform, the dropped modules can be added at any time, should the model be ready and the hardware able to withstand the workload.

While outlining the specific analysis domains, a series of dimensions along which a system is assessed and the related performance metrics has been worked out. If, for some of them, the choice came naturally (e.g. mass, first resonance frequency, etc), for others, more subjective and articulated, a specific synthesis work was required (e.g. engine performance).

A detailed description of the specific analysis done, the related methodologies adopted and the resulting models are given in Chapter 3.

ICE PlatformArchitecture / platform layout

Having identified the individual design domains, the next step was selecting a robust and yet flexible architecture that was to link them in a consistent framework. The trade-off was between integral and modular.

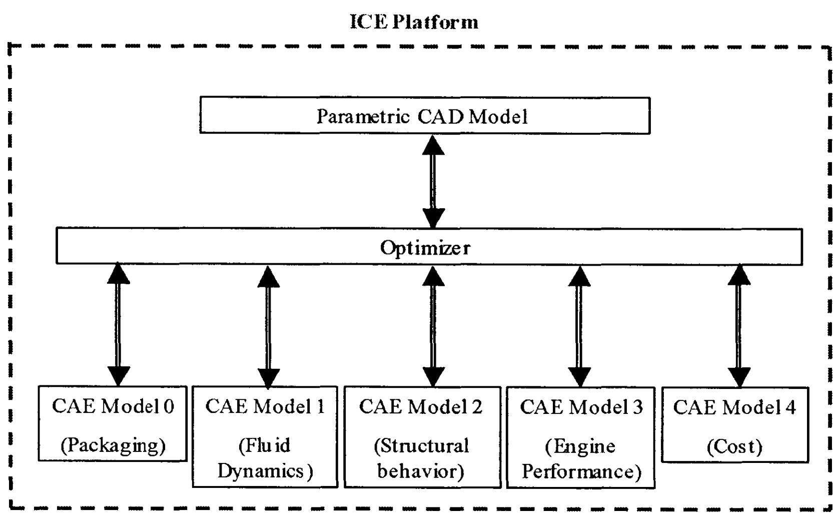


Fig. 15: the Integrated Concurrent Engineering Architecture

Integral would have meant a deep interaction between the different software packages. In this sense, there are several attempts by different software vendors. The usual situation that is encountered is that CAD or CAE vendors, specialized in one domain with a flagship product, are trying to widen their area of action around that product offering an extension of their basic capavbilities.

No single CAE platform, however, exists that is able to incorporate seamlessly different codes of different vendors. This was one of the main motivations for the choice of the modular architecture, with the optimizer to act as a data transfer bus. In this bus architecture a standardized and minimum set of data is exchanged among applications and the transfer back and forth is handled uniquely by the optimizer. This approach works well in the case of a maniverter because the main interaction between disciplines is via system's geometry. In other applications (e.g. aero-structural design), with deep multiple interactions between coupled disciplines, a different approach might offer superior performances.

As mentioned, a modular architecture has got other advantages: 1) one CAE module can be replaced by another one, provided that the interfaces with the bus are the same; 2) if, at any time, an additional CAE module becomes available, it can be easily integrated and only the interfaces with the bus (and not with all other packages) need to be defined.

In addition to the software architecture, the hardware architecture was also defined. In particular, two main trade-offs were investigated: single platform - distributed execution and workstation / desktop – laptop. Each of them has its own pros and cons (Fig. 16): while distributed computing offers superior computing performance, the single platform is amenable of a simpler implementation. A workstation and a desktop are, again, superior in terms of computer power, but a powerful laptop may be competitive and offer unique mobility characteristics.

	Single Platform - Desktop	Single Platform - Laptop	Distributed Execution
Simplicity of Implementation	☺	☺	☹
Robustness	☺	😐	😐
Computing power	😐	☹	☺
Mobility	☹	☺	😐

☺ Good
😐 Sufficient
☹ Bad

Fig. 16: Hardware Architecture Options

The final choice has been to develop the application on a single computer and on a laptop. Since, in fact, it was anticipated that the development of the application would have been shared among different individuals and companies, portability was an essential feature that would allow the interchange of information. The single computer, single operating system, made easier the implementation.

Hardware / software requirements analysis and partnership building

Having identified the disciplines and the analyses involved, the following step was to identify software candidates for the different computational jobs and the hardware/operating system.

For each of the parts of the ICE platform at least three different options were considered and two of the market leaders were contacted for collaboration, Tab. 1.

Architecture Component	Vendor	Software Name
Optimizer	Engineous	iSIGHT
	MIT	DOME
CAD	UGS PLM Solutions	Unigraphics

Architecture Component	Vendor	Software Name
	Dassault Systemes	CATIA
Structural Analysis	MSC	Patran / Nastran
	Ansys Inc.	Workbench / Ansys
Fluid dynamic analysis	AVL	Boost
	GTI	GTPower
	Ricardo	Wave

Tab. 1: Software Packages Options for the ICE platform Analyses Modules

The request to each and every vendor was to provide a temporary license of the required software and the support of a dedicated application engineer for its customization for the specific application.

All of them declared their availability to provide a temporary software license and some of them offered some sort of engineering support. At the end, the team was formed with the partners that proved proactive and particularly supportive and whose software demonstrated suited for the ICE platform development and for integration with the other packages. The companies that eventually were admitted in the team are highlighted in Tab. 1.

In addition to those, the collaboration of two more companies was necessary to cover two important areas:

- Parametric CAD model generation - Autostudi
- CAD geometry update - Centro Ricerche Fiat

The complete architecture and distribution of responsibilities and competence is represented in Fig. 17:

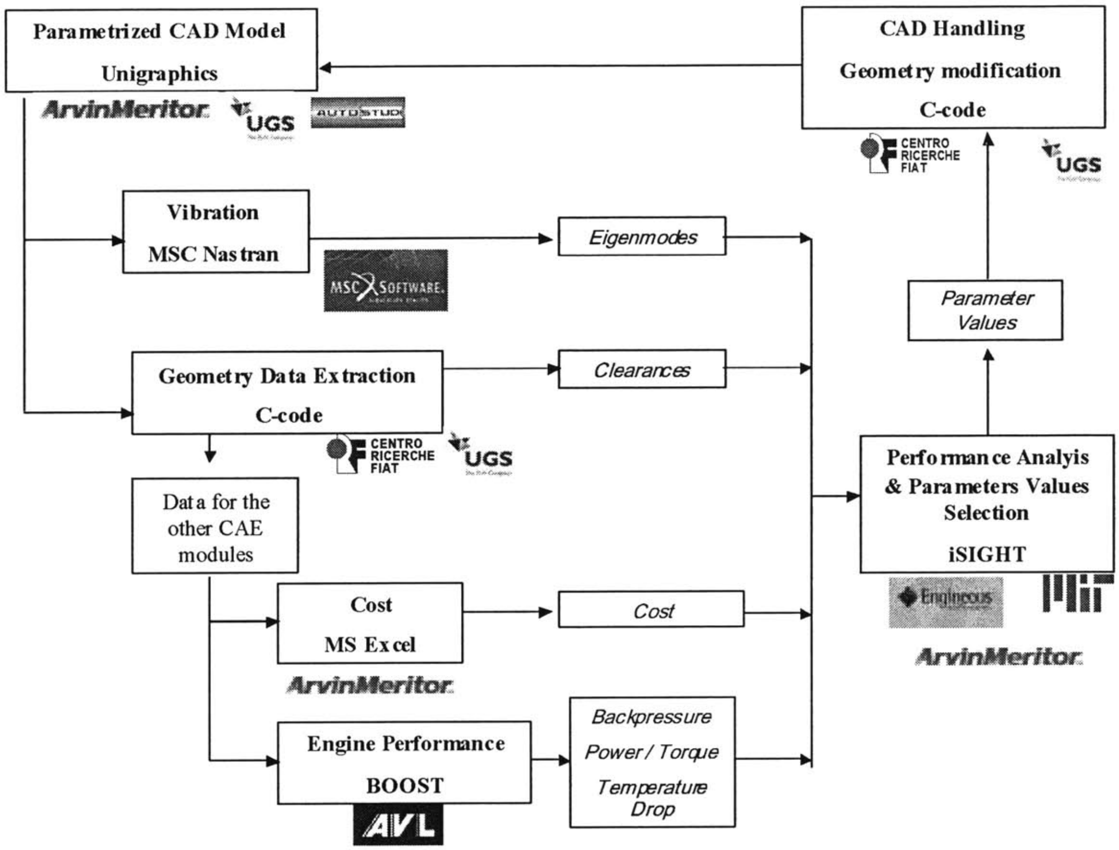


Fig. 17: ICE Platform Architecture and Team Members Coverage

As far as the hardware was concerned, given the budget limitations, a notebook was used with reasonably high specification, a Pentium IV 1.8 GHz, with 1MB RAM and 15GB HD. Microsoft Windows 2000 Professional SP4 proved to be a convenient, stable and reliable operating system environment for all software packages.

Project Stages: Simple Pipe and Full Manifold

Given the complexity of the framework and the maniverter application due the number of interfaces involved, an incremental approach was used. The project was sub-divided into two main phases: Phase 0 and Phase I/II.

To test the software working conditions and interoperability issues, in Phase 0 the first prototype of the EDF was built around a simple geometrical application: a pipe of fixed centreline but variable cross sections and thickness (Fig. 18). This was used as a trainer. In this phase, the routines for the automatic analyses were developed, the response in terms of reliability and execution time of the different codes was tested, the pros and cons of different design space exploration and multi-disciplinary optimization algorithms were verified and the complexity of post-processing of multi-dimensional data was tackled.

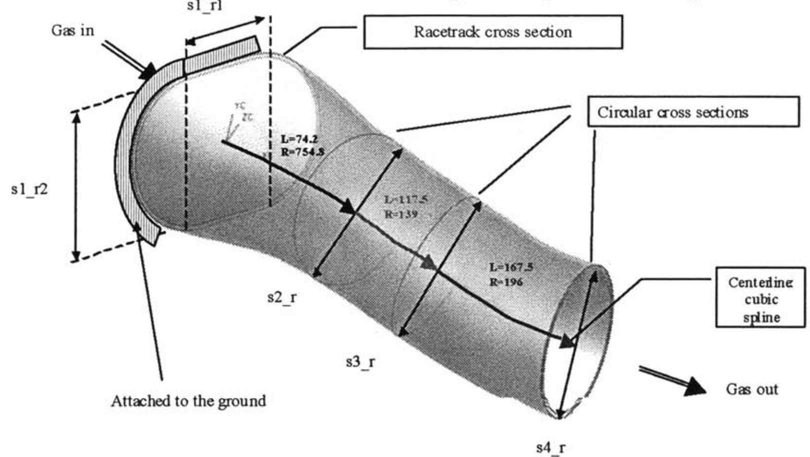


Fig. 18: The Simple Pipe

Partly overlapping with the end of the phase 0, the Phase I/II consisted in building the EDF around a complete maniverter application, in a simplified form but representative enough of a real life application, Fig. 19.

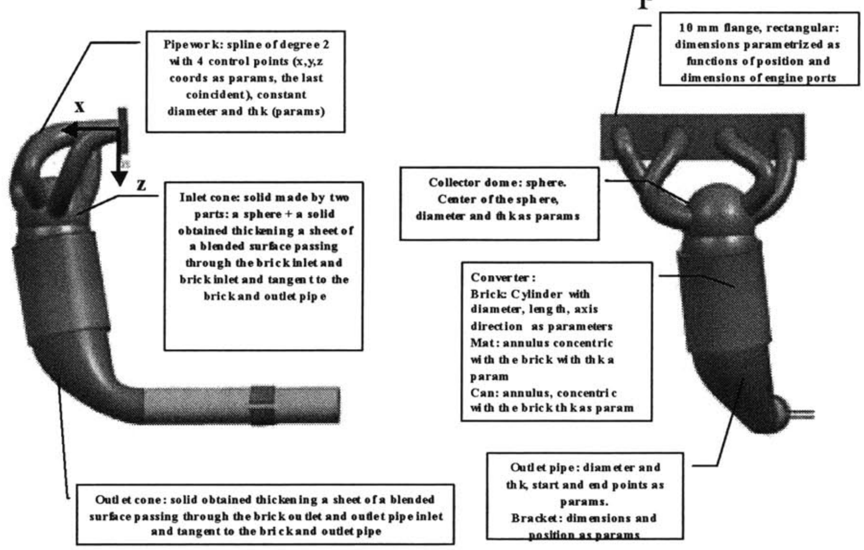


Fig. 19: The Full Maniverter

The incremental approach proved successful because, as expected, the raised complexity added a totally new set of issues and led even to rewriting important portions of already developed software. If some problems hadn't been solved in the previous Phase 0, the overall complexity would have been overwhelming.

Literature search

Since the beginning and in parallel with the activity of building the EDF, a literature survey was made that lasted throughout the period of the work. The survey had several aims:

- Understand the research activity in product development improvement
- Evaluate the application of MDO to product development: current challenges and perspectives.
- Learn from specific existing examples on maniverters
- Learn optimization / design space exploration techniques
- Understand post-processing issues

Results are presented in Chapter 2.

Cost-Benefit Analysis

Given the results deliverable by the EDF, for the same maniverter development application the economic benefits stemming from the lowered costs, lowered development time and increased agility and innovation are estimated and balanced against the development costs necessary to bring the tool from the current status of demonstrator to a level where it could be used for normal business activity.

Implementation layout

Finally a roadmap for implementation is outlined. An approach of insertion of this new tool in a manufacturing firm is given, together with the analysis of the major implications, particularly from the organizational viewpoint.

1.3.3. Structure of Thesis

After Chapter 1 where the outline of the business scenario and the drivers for the present work are analyzed, in Chapter 2 a literature review of the major areas touched by the present work is presented. For sake of clarity, it is divided in the following Sections: PD process analysis and improvements, MDO (general status and benefits in terms of lead-time/cost reduction), exhaust system related MDO application, MDO enablers, 3+ dimensions visualization techniques.

In Chapter 3 the demonstrative EDF is described in detail. Starting from the application description and the outline of the architecture, the individual modules are explained in detail with particular emphasis to their limitations and their interfaces with the other modules.

In Chapter 4 the EDF is put at work. Simulation set-ups, the details of the runs and main results are presented for both the simple pipe and for the maniverter application and insights are drawn. At the end all the insights are summarized.

Chapter 5 outlines a path for the further development of the tool from the current status of demonstrator to a level where it can be used in a business setting. The cost benefit analysis is presented to compare the current product development process and the proposed new product development paradigm and to show that the business case of the development project is solid. It then discusses the technical and organizational challenges of the EDF implementation in a real environment.

Chapter 6 is the summary of the whole work: it collects all the insights gained during the activity and couples them with a forward looking perspective of the automotive industry, product development and computers evolution presenting a forecast of product development computerization as a standard paradigm for the future. Recommendation for action ends the work.

2. LITERATURE REVIEW

2.1. Introduction

In this Section the knowledge gathered during the project-long literature search is synthesized.

The material is grouped in four Sections:

- Product Development Process Improvement Research
- MDO methods
- MDO key enablers
- Examples of applications of the MDO approach

2.2. PD Process Improvement Research

> *"The goal for any product development improvement effort should be to conceive human-centered design processes that result in efficient, effective, user acceptable system interfaces that are simple to train, use, and maintain."*

Given its crucial importance, it is not surprising that the design of effective product development processes has received considerable attention.

Initiatives and pilot project targeted at developing world-class product creation process that is more flexible, adaptable, dynamic and low cost are multiplying both in Academia and in the industry. A complete survey is beyond the scope of this work, but we want anyway to describe some examples of the research activity running in different contexts.

- **Evolutionary Product Development**
  - Various competing solutions
  - Continuous advancement
- **Shortening of control cycles**
  - Relocation of efforts (and gain of insights) into early phases
  - Short fast iteration loops
  - Early assessment and feedback of results
- **New processes of organisation and management**
  - Self-organisation and coordination
  - Short paths of decisions

Fig. 20: Rapid Product Development as new Approach

Two are the common components among the different works: the heavy reliance on Computer Aided tools to shorten the virtual design loops and the careful shaping of the underlying organizational structure and culture, Fig. 20.

Three works on new approaches for product development process are presented: a proposal from the Fraunhofer Insitute of Technology, and two pilot projects experimented by BMW and Ford of Europe.

2.2.1. Fraunhofer Institute Perspective

A Fraunhofer Institute research [25] identifies rapid product development the main paradigm of the next ten years. Three main key enablers to that are also identified: many competing solutions, short/fast iteration loops and a self-controlling organization (Fig. 21).

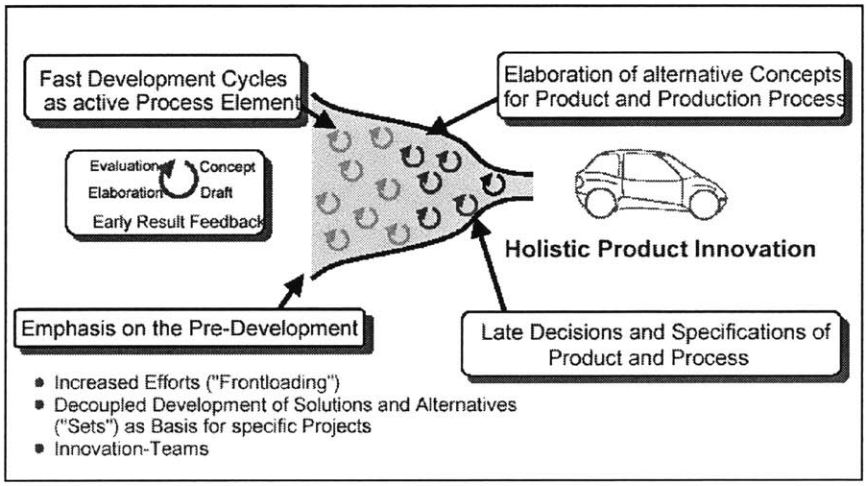


Fig. 21: Evolutionary-Iterative Development

More specifically a scenario is conceived where many solutions coexist until late in the project and they are continuously evaluated in light of the project evolution. This operational mode carries the great advantage of the flexibility: having many options in the solutions portfolio, depending on the particular needs over time, one solutions might be chosen because it offers superior benefits than another one. However, to carry on the development of multiple solutions simultaneously, fast development cycle is mandatory.

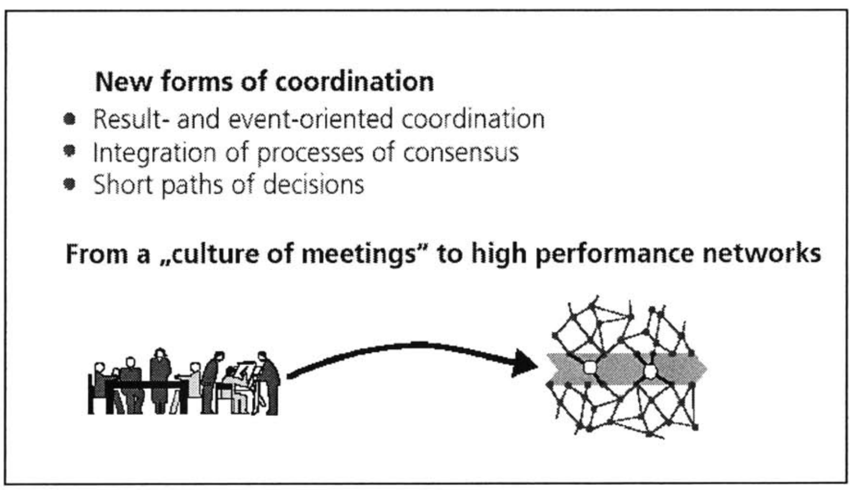


Fig. 22: New Forms Of Coordination

The other essential element of the Fraunhofer research is the holistic or system perspective: the product is evaluated against all its performance attribute measures so that iterations deriving from downstream processes are avoided.

From the organizational standpoint, an unprecedented change is proposed: from a hierarchical, rigidly controlled structure to a networked, self-adjusting, self-governing structure, Fig. 22. If we want to make an analogy with an aircraft system, it's like moving from an intrinsically stable aircraft to an intrinsically unstable aircraft. The first offers increased safety and controllability at the penalty of a slow change in direction, the second guarantees a superior manoeuvrability but at a price of a "real-time brain" that controls and adjust the configuration at every instant.

2.2.2. BMW Pilot Project

In 2002 BMW has initiated a pilot project, called The Digital Auto Project, with the aim of

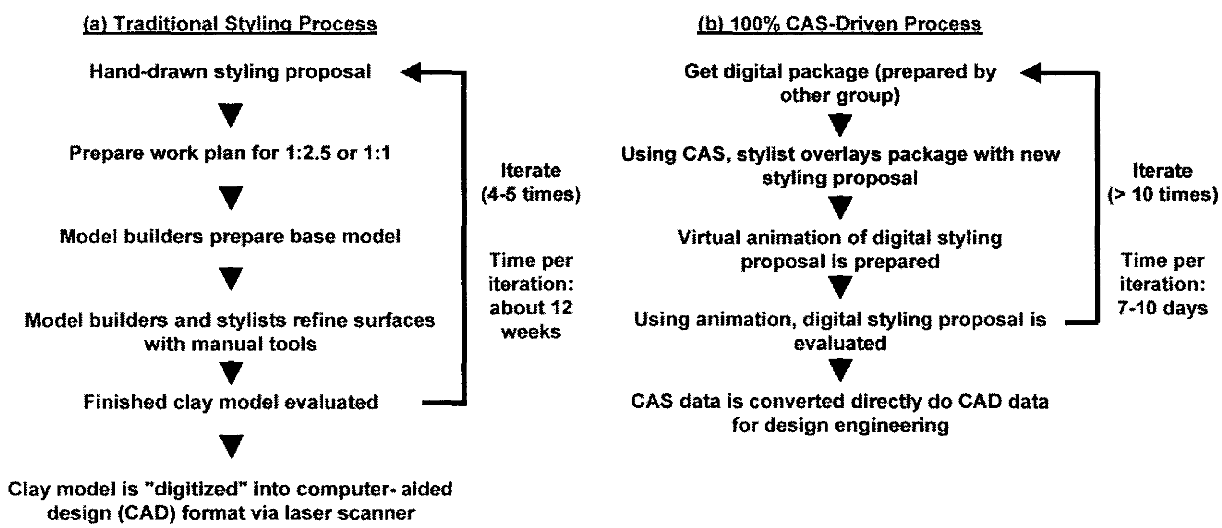


Fig. 23: The BMW Digital Auto Project

slashing by 50% the development time [25] . This project is based on three basic working principles: 1) increased parallelisation of design tasks, 2) elimination of some design tasks such as physical prototyping and 3) quicker completion of the remaining design. In Fig. 23 the traditional development process is compared with the new approach.

The intensive use of the Computer Aided Simulation (CAS) tools and the fast communication channel allow the time per iteration to be amazingly squeezed to 15% of the original. This allows the number of iterations to be increased (from 4, 5 to 10+), thereby improving product's quality and value, and still achieving a significant reduction in the overall development lead-time (50%).

2.2.3. Ford C3P

The Ford Motor Company, in 2003, celebrated its 100 anniversary but it's not content to rest on its history as an innovator in vehicle design and production. In fact the company is constantly striving to narrow the gap between concept and production. The Ford Europe team decided that the only way to get to design quicker was to fundamentally change its design, engineering and manufacturing processes [26] . As John Sullivan, Chief Engineer for Body product Development Ford Europe, said "We can't focus on taking one or two days out of the process; we need to take out months".

The response was the creation of an integrated package of computer-aided design, computer-aided engineering, computer-aided manufacturing and a comprehensive product information database. The computer-aided toolset, which is the latest version of the C3P product development environment (see Fig. 24 [27]), allows designers and engineers to create a new vehicle largely in a digital environment. The broad capabilities of C3P allow Ford engineers to reduce the number of hard prototypes created during the engineering process. The sophisticated array of computer tools allows engineers to build virtual prototypes of their digital designs and to "test" their function entirely in the digital environment.

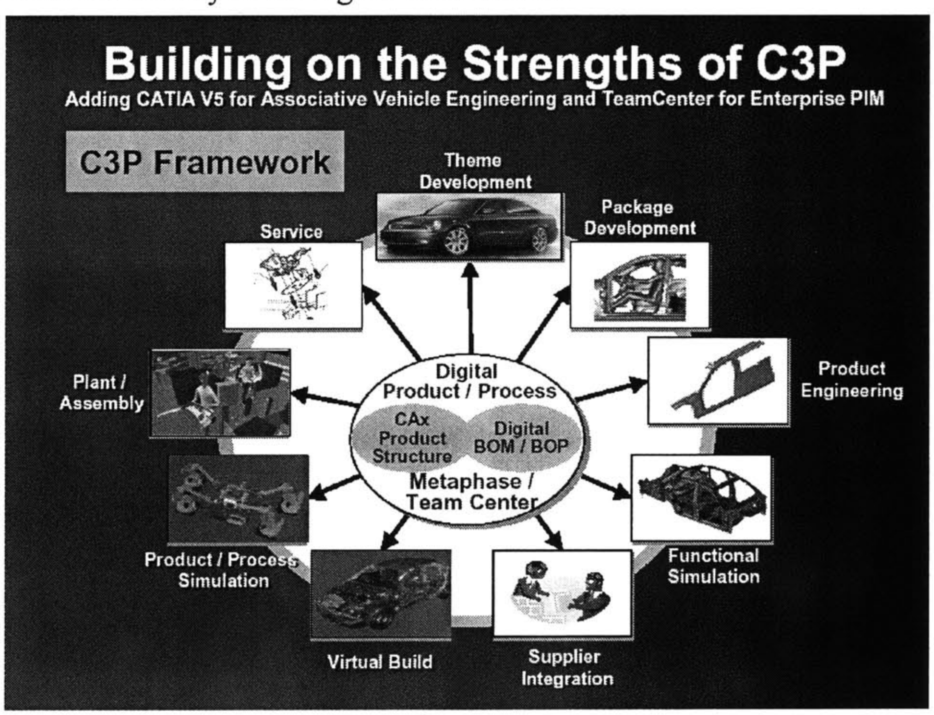


Fig. 24: The Ford C3P Framework

With this environment, the three months that used to take to go though an analysis cycle, from the time a design is clear, have been slashed to ½ day.

The parallel design / analysis approach has led to 25% reduction in staff, delivering significant cost savings.

2.3. Multi-disciplinary Design Optimization

2.3.1. Multidisciplinary Technology Overview

Practically all engineered and manufactured systems, such as automotive vehicles and aircrafts as well as many consumer products, experience interactions among various physical phenomena and between various components of the full system. These interactions make the system a synergistic whole that is greater than the sum of its parts. Taking advantage of that synergy is the mark of a good design, but the web of interactions is difficult to untangle.

The complexity of these interactions combined with the need to partition the design work into subtasks that can be executed simultaneously in order to compress the project time gave rise to the conventional practice of dividing the detailed design work into specialty areas. This decomposition is usually centered on a physical phenomenon, such as structural deformations or fluid flow, or on a hardware subsystem, such as a vehicle's suspension system.

This reductionist approach, however, decouples only *temporarily* the subsystems that were originally linked. As expected, when reassembling them, the influences of the coupling links will manifest themselves again and, if they are negative for the system, a design iteration will stem: it's common experience that designs are passed between product teams and or departments several times until the differences are minimized and a mutually acceptable solution is found.

We argue that Multidisciplinary Design Optimization (MDO) is an appropriate approach to tackle the complexity of modern and future product development. A detailed illustration of techniques, methods and algorithms is beyond the scope of this work and can be found, among papers and books dedicated to the subject, in [28] - [32] . In addition, the interested reader can access the numerous online resources, a list of which is presented in the Appendix 7.5. In what follows the fundamentals are given as a background.

Introduction to Multidisciplinary Design Optimization

Multidisciplinary Design Optimisation can be defined as a formal methodology for the design of complex coupled systems in which the synergistic effects of coupling between various interacting disciplines/phenomena are explored and exploited at every stage of the design process. Some other popular definitions for MDO are:

- A methodology for the design of complex engineering systems and subsystems that coherently exploits the synergism of mutually interacting phenomena.
- Optimal design of complex engineering systems which requires analysis that accounts for interactions amongst the disciplines (or parts of the system) and which seeks to synergistically exploit these interactions.
- "How to decide what to change, and to what extent to change it, when everything influences everything else."

In a MDO general framework (see Fig. 25, [33]), a system is defined by a series of design variables (design vector x) and characterized by a set of performance attributes (objective vector J).

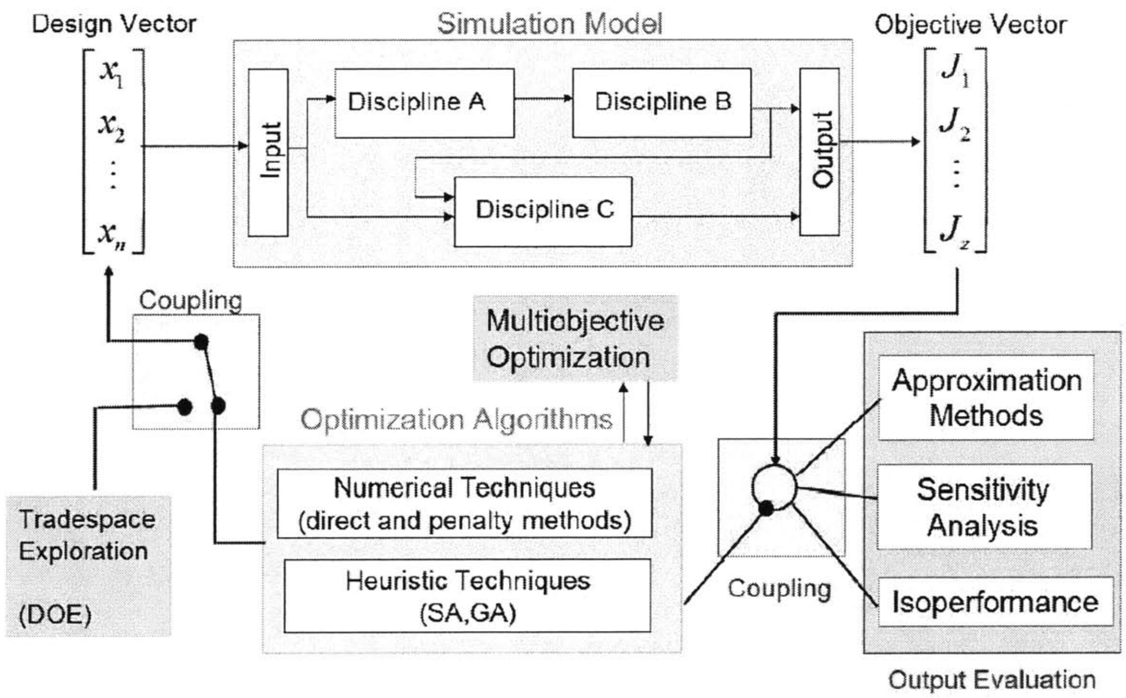


Fig. 25: General MDO Framework

Performance attributes are computed through a series of models. Appropriate algorithms allow the analysis of the relationship between design variables and performance attributes.

Types of Analysis

Typical analyses involve design space exploration, single objective optimization and multi-objective optimization.

Design Space Exploration is a collection of statistical techniques providing a systematic way to sample the design space. It is useful when tackling a new problem for which very little about the design space is known and it is often used before setting up a formal optimization problem to identify key drivers among potential design variables, to select appropriate design variable ranges and to set up achievable objective function values. A list of the main techniques is summarized in Fig. 26 [33] . For a comprehensive review and comparison, see [34]

TECHNIQUE	COMMENT	EXPENSE (l=# levels, n=# factors)
Full factorial design	Evaluates all possible designs.	l^n - grows exponentially with number of factors
Orthogonal arrays	Don't always seem to work - interactions?	Moderate – depends on which array
One at a time	Order of factors?	$1+n(l-1)$ - cheap
Latin hypercubes	Not reproducible, poor coverage if divisions are large.	l - cheap
Parameter study	Captures no interactions.	$1+n(l-1)$ - cheap

Fig. 26: DoE Techniques Overview

In single-objective optimization, the system is "optimized" in order to find the design solution, which is characterized by the maximum or the minimum of a particular performance attribute.

The multi-objective optimization, on the other hand, usually results in a set of optimal solutions, which lie on the trade-off hyper-surface between the different conflicting criteria. These non-dominated solution points are called Pareto optimal solutions. They constitute a set where every element is a problem solution for which further improvement in one of the performance attributes requires sacrifice in at least one of the other attributes. As such, any one of them is an acceptable solution and can be considered "optimum" in some respect. Once the set of optimal solutions is identified, the designer has the freedom of choosing one solution out of many possible alternatives based on experience, prior knowledge and other criteria or constraints particular to the current design problem. One way to simplify the multi-objective optimization problem is to create a linear combination of the objectives choosing a priori a weighting factor for each objective function; then the process becomes a single-objective optimization. The outcome of this simplified process will largely depend on the vector of weights used in the linear combination. More advanced approaches use Adaptive Weighted Sum Optimization.

Optimization Algorithms

As far as algorithms for optimization are concerned we can distinguish two different broad categories: gradient based algorithms and heuristic techniques.

Gradient-based algorithms are designed to search the minimum or the maximum of an objective function $J(x)$ using some information about its gradient. Various techniques and implementations are possible, with different level of complexity and varying computational effort. A list is presented in Fig. 27, [33] .

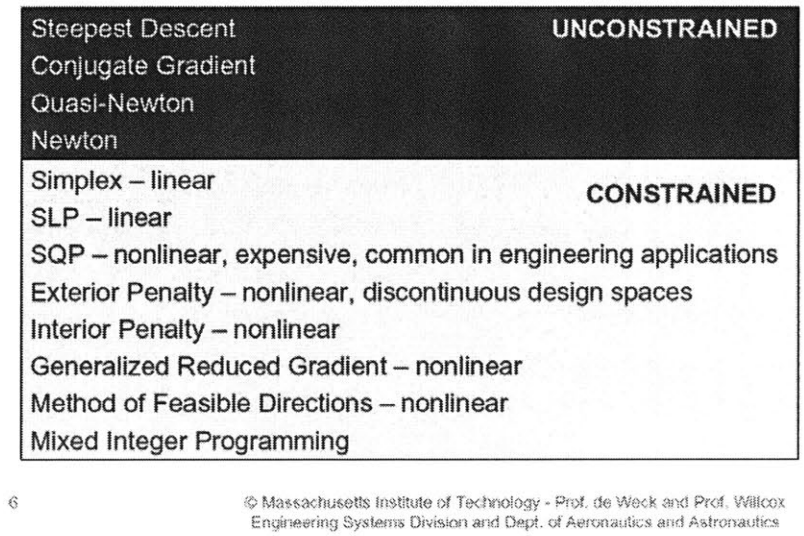


Fig. 27: Optimization Algorithms

In the last decade, a different category of algorithms has evolved which is commonly known with the name of heuristic algorithms. They facilitate solving optimization problems that were previously difficult or impossible to solve. These tools include Evolutionary Computation (mainly Genetic Algorithms), Simulated Annealing, Tabu search, Particle Swarm, etc. Genetic Algorithms are, by far, the most popular ones. They are stochastic search methods that mimic the metaphor of natural biological evolution. Genetic Algorithms operate on a population of potential solutions applying the principle of survival of the fittest to produce better and better approximations to a solution. At each generation, a new "population" is created by the process of selecting individuals according to their level of fitness in the problem domain and breeding them together using operators borrowed from natural genetics. This process leads to the evolution of populations of individuals that are better suited to their environment than the individuals that they were created from, just as in natural adaptation. Genetic Algorithms model natural processes, such as selection, recombination, mutation, migration, locality and neighborhood. They work on populations of individuals instead of single solutions. In this way the search is performed in a parallel manner.

Genetic Algorithms differ substantially from more traditional gradient-based search and optimization methods. The most significant differences are: 1) Genetic Algorithms search a population of points in parallel, not a single point; 2) they do not require derivative information or other auxiliary knowledge; only the objective function and corresponding fitness levels influence the directions of search; 3) they use probabilistic transition rules, not deterministic ones; 4) they can provide a number of potential solutions to a given problem and the final choice is left to the user.

Approximations

In MDO, computer simulation codes are discipline-specific, composed in different languages (e.g., Fortran, C, Java), distributed, both geographically and on different computer platforms and computationally expensive due to fidelity of modelling and need for accurate results. Sometimes, computationally expensive computer simulations and/or analyses are replaced by surrogate models, which are fast, simple approximations. As for the Design Space Exploration and for the Optimization Algorithms, also for approximation functions a vast series of techniques exist that go from a simple polynomial interpolation to more accurate methods of representing localized behaviour, such as radial base functions or kriging models [35] .

The evolution of MDO: from optimization of product performance to business

MDO research has blossomed in the 20 years since the first Multidisciplinary Analysis and Optimization symposium, held in April 1984 at NASA Langley Research Center, and has evolved since then as a new discipline that provides a body of methods and techniques to assist engineers in moving engineering system design closer to an optimum. Parallel to the development of the above methodology, a number of software packages have been created to facilitate integration of codes, data, and user interface leading it its emergence as a tool for mainstream application in product and process development.

Design of engineered products, in fact, can only be done in the context of the producing enterprise and the market in which the product must exist. Traditional design optimization has been limited to design decisions about engineering performance. Product success for both producer and user clearly depends on other requirements, including production requirements, marketing, and investment strategies, collectively referred to as enterprise-wide design. In an effort to bring design optimization into a more central position within the enterprise, and thus increase its value and impact, there is increased effort in augmenting the engineering physics models of performance with models from production, economics, investment science and marketing.

The aerospace industry has been applying optimization in some form to multidisciplinary design problems since its inception. Other industries, such as automotive, electronics and computers, transportation and energy/power generation and distribution, followed.

2.3.2. Available MDO software tools

With increasing acceptance and utilization of MDO in industry, a number of independent software vendors developed the software frameworks that facilitate integration of application software, data, and user interface, along with various MDO- related problem- solving functionalities. A non-exhaustive list is provided of key ISVs that specialize in engineering process integration and MDO (Tab. 2), as well as the traditional CAE and CAD vendors that support some functionality toward design optimization (Tab. 3).

	Product	Comments
Engineous Software	iSIGHT	Originated from GE, CR&D, MIT
Phoenix Integration	ModelCenter	Originated from VPI
Technosoft	AML	Originated from USAF
Altair Computing	HyperOPT, OptiStruct	Focused on structural design
EASi Engineering	ST-ORM	Originated from CASA Spain
LMS International	LMS OPTIMUS	Automotive industry
Samtech	BOSS/Quattro	
Esteco	modeFRONTIER	Originated from EU project

Tab. 2: ISVs Specializing in MDO and Process Integration Applications

	Product	Comments
MSC.Software	MSC.Nastran SOL 200	By VMA Engineering
Ansys Inc	ANSYS	Design sensitivity analysis
HKS Inc	ABAQUS/Design	Design sensitivity analysis
LSTC	LS-OPT	Response surfaces based optimization
ESI	PAM-OPT	Numerical Optimization
Mecalog	RADIOSS	Design sensitivity analysis
Vanderplaats R&D	GENESIS	
PTC	Pro/ENGINEER, Pro/MECHANICA	
SDRC	I-DEAS	

Tab. 3: Traditional CAD and CAR ISVs with Design Optimization Functionality

2.3.3. MDO as a tool to harness engineering complexity

> *In order to understand anything, you must not try to understand everything.*
>
> Aristotle

The span of immediate memory imposes severe limitations on the amount of information that we, as humans, are able to receive, process, and remember. It's now over forty years that Miller published his study [36] where he reported, among his findings, that a magical number 7 (+/-2) is the maximum number of sensory inputs a person can handle without error. During this time, it has been studied extensively by both psychologists and other sensory scientists and, despite the extensive challenges, it still remains valid in its generic terms.

On the other hand, looking at the evolution of engineering systems, we can distinctively see that, over time, our creative ability has resulted in their essential improvement. At the same time, however, also complexity that arises from the systems' entangled interconnections has increased, to such an extent that it now far outstrips our intuitive mental capacity for dealing with it. As if product's complexity was not enough, modern communication systems provide each of us overwhelming mountains of information, much of which is unorganized, not relevant, redundant, or inaccurate; and thus, may well provide more confusion than clarity.

The engineering design process is recognized as a two-sided activity as illustrated in Fig. 28 [29] .

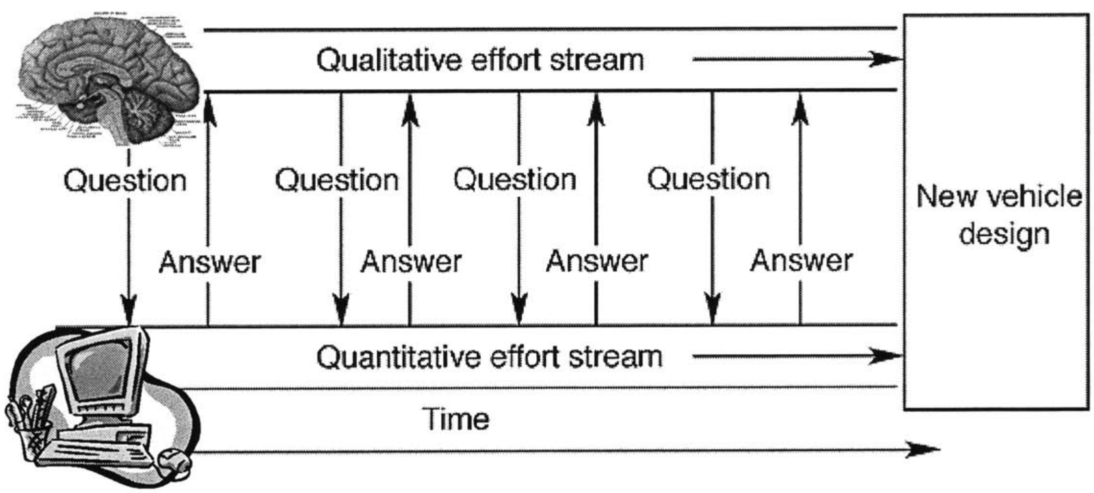


Fig. 28: Parallel, qualitative, and quantitative efforts in design

It has a qualitative side dominated by human inventiveness, creativity, intuition and synthesis capabilities. The other side is quantitative, concerned with generating numerical answers to the questions that arise on the qualitative side. The process goes forward by a continual question-answer iteration between the two sides.

If at first it may appear that design optimization is a means to replace the engineer and his or her expertise in the design loop, this is certainly not the case. In fact, any design optimization application cannot infer what should be optimized, and what are the design variables – the quantities or parameters that can be changed in order to achieve an optimum design. The MDO methodology discards the "push button design" idea in favor of a realistic approach that recognizes the role of human mind as the leading force in the design process and the role of mathematics and computers as indispensable tools. This approach is consistent with the creative characteristics of the human brain and the efficiency, discipline, and infallible memory of the computer.

Depending on the complexity of the system, the MDO-based environment of the future can be thought centered on one or a core team of senior Design Engineers that may rise at the role of Product Managers. It hides the complexity of the product inside its coherent and quickly adaptable structure presenting to the users a simplified and yet comprehensive picture of the product and of its performances. To facilitate its use the MDO process it's thought as interactive and permitting the engineers to formulate its design problem in real time as the design issues become clear. Specifically, the MDO process should be flexible enough so that the problem formulation, applied constraints, and the fidelity level of simulation can all be specified by the product team. An environment that offers visibility to the process, permitting the team to monitor progress or track changes in the problems dependent or independent variables will be beneficial. The environment could be distributed to reflect the nature of today's design projects.

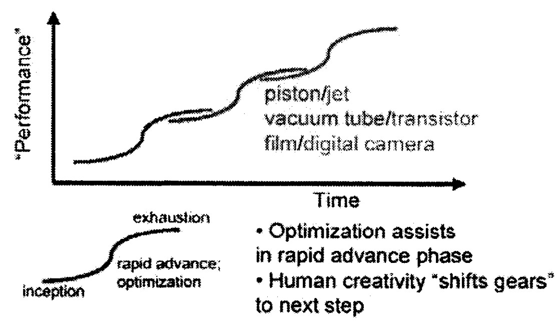


Fig. 29: Technology Progress Sigmoidal Staircase

Whatever the hardware and software infrastructure may be, best results will come if complemented by an effective team organization [37] . In fact, MDO will assist in the design process while, all along, the control would remain squarely in the hands of the design team. Any savings in terms of human resources could then be best used to create new solutions Fig. 29.

2.4. MDO enablers: the ICE Platform in a HPC environment

For the MDO approach to be successful, its processes will need to be executed in an environment that supports:

- The ability to easily access remote analysis tools as well as easily bringing together multiple analysis tools into an integrated system analysis while hiding the details of data management from the user. This includes linking data between different analysis components on different platforms.
- Meta-computing consisting of a collection of high performance machines that can provide the aggregate computing powers necessary for solving large-scale, multidisciplinary optimization problems.

In the following Sections, we will analyze the fundamental reasons why these are key enablers and current issues and future perspectives.

2.4.1. ICE: the solution to software interoperability

> *"There is little question that the interoperability problem is a significant one. It consumes tremendous resources that could be more productively deployed elsewhere. It inhibits the achievement of broad corporate and national goals. It jeopardizes quality and safety of manufactured products by allowing error to persist in the design and production process."* [38]

Many product development processes rely on a frequent exchange of information among different stakeholders. Organizations pass information back and forth between internal engineering and manufacturing groups, as well as with supply chain partners. However, translating geometric models of complex products is an imperfect, error-prone process that reduces the content and value of the information to the lowest common denominator. Repairing or recreating product data can impose uncertainties, errors and delays in progress from concept through manufacturing – delays that, in today's highly competitive marketplace, can mean the difference between winning and losing business.

A 1999 study commissioned by NIST reported that the U.S. automotive sector alone "wastes" one billion dollars per year due to design data incompatibility and the need to re-enter the same data multiple times in different systems. For all sectors, estimates of $20 to $40 billion have been put forth [39] Studies have, for example, shown that in the most advanced "all-digital" aircraft design efforts, engineers manually execute about one million data transfers at a cost of many millions of dollars.

If data exchange is a cumbersome, time and cost consuming part of a relatively low clockspeed development process, in a much faster clockspeed environment such as an MDO one, the drawbacks are so exacerbated that they make design optimization impossible. The information flow between the computational tools is an essential enabling factor when making use of an automatic optimization procedure. Any MDO approach must therefore be executed in an Integrated Concurrent Engineering (ICE) hardware/software environment.

Generalizing, an environment may be called integrated if the tools that belong to this environment are able to share and exchange information. In other words, several tools compound an integrated environment if the results produced using one of them are suitable to be used for some of the others.

To make such an integrated environment, a holistic analysis of the CAx/IT architecture must be performed at the outset in order to enable the various tools to interoperate seamlessly. All the steps of the analysis loop must be examined, all the inputs required by each step must be listed and proper processes much be put in place to retrieve the required data from the previous analysis steps and adequately transform and assemble them in a suitable format.

While this requirement may pose additional challenges to the MDO approach, it is a mandatory requirement for it.

2.4.2. High Performance Computing

A simulation based design process almost invariably relies heavily on complex computer analysis codes and simulations (e.g., FEA and CFD). These time-consuming and expensive analyses are repeatedly invoked during optimization making the design exploration and multidisciplinary design optimization time very long, if not prohibitive.

One solution to make the problem tractable is to use Approximation Models (also referred to as Surrogate Models). These are, in essence, approximations of the output with simpler functions. Since these approximate models are inexpensive to evaluate for a new set of data or values assigned to design variables, we can afford to evaluate approximate responses many more times without having to worry about the computational resources. Surrogate models, however, have several shortcomings. They usually require, to be created, the execution of the high fidelity models a considerably large number of times. In addition, the necessary simplification that is implied in the surrogate models may cause misleading information to be introduced in the optimization process. This is particularly true in case of highly non-linear complex problems, such as shock waves in supersonic flow, resonances in lightly damped structures, etc. As a result the entire MDO can be seriously compromised.

Whenever possible, the use of the high fidelity models is recommended. In this case, the investigation of high dimensionality for optimization and the complexity and expense of the underlying analyses require, for practical turnaround of MDO solutions, the use of High Performance Computing, with servers with a large number of processors and multiple levels of parallelism (coarse and fine grained parallelism) to deliver high throughput computing.

If this can represent a roadblock now, it will be mitigated and eventually removed as time goes by. Over the past decades, in fact, we've observed a continuous and amazing progress in the computing performance and the years to come are believed to see a similar progression.

Hereafter, we present just some glimpses of computers evolution to hopefully show that MDO has an open avenue ahead.

Gordon Moore, in 1965, just four years after the first planar integrated circuit was discovered, observed an exponential growth in the number of transistors per integrated circuit and predicted that this trend would continue. Moore's Law, the doubling of transistors every couple of years, has been maintained since then, and still holds true today, Fig. 30 [40] . Expectations are that it will continue at least through the end of this decade. This translates roughly into a 1% performance increase every week.

On 11 July 1977 the CRAY-1A, serial number 3, was delivered. It could execute over 80 megaflops. The system cost was $8.86 million ($7.9 million plus $1 million for the disks).

The supercomputer weighted 5.5 tons, arrived in two refrigerated electronic vans, and needed more than 30 construction workers, engineers, and helpers to move it into the computer room.

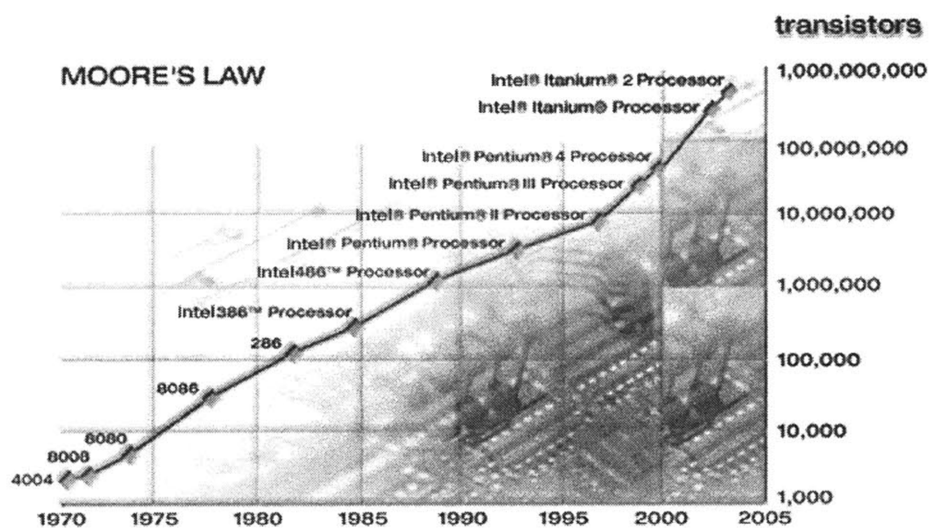


Fig. 30: Moore's Law

It typically had 8MB of RAM and 2.5GM HD.

Only slightly more than 20 years later these performances were surpassed by a Pentium IV laptop, for a cost of $1000 and a weight of 1.5 kg.

The progress over the years has been really impressive. A view of performance escalation is given in Fig. 31 [41] .

(Entries for this table began in 1991.)

Year	Computer	Number of Processors	Measured Gflop/s	Size of Problem	Size of 1/2 Perf	Theoretical Peak Gflop/s
2000	ASCI White-Pacific, IBM SP Power 3	7424	4938	430000		11136
1999	ASCI Red Intel Pentium II Xeon core	9632	2379	362880	75400	3207
1998	ASCI Blue-Pacific SST, IBM SP 604E	5808	2144	431344		3868
1997	Intel ASCI Option Red (200 MHz Pentium Pro)	9152	1338	235000	63000	1830
1996	Hitachi CP-PACS	2048	368.2	103680	30720	614
1995	Intel Paragon XP/S MP	6768	281.1	128600	25700	338
1994	Intel Paragon XP/S MP	6768	281.1	128600	25700	338
1993	Fujitsu NWT	140	124.5	31920	11950	236
1992	NEC SX-3/44	4	20.0	6144	832	22
1991	Fujitsu VP2600/10	1	4.0	1000	200	5

Fig. 31: Top Computers Over Time for the Highly-Parallel Linpack Benchmark

The DOE/IBM BlueGene/L beta-System is today the fastest computer in the world, with its record Linpack benchmark performance of 70.72 Tflop/s ("teraflops" or trillions of calculations per second), Fig. 32. It is closely followed by the Columbia system built by SGI and installed at the NASA Ames Research Center clocked in at 51.87 Tflop/s. The Earth Simulator, with its Linpack benchmark performance of 35.86 Tflop/s, had held the No. 1 position for five consecutive editions of the listing and is now shown as No. 3. [42]

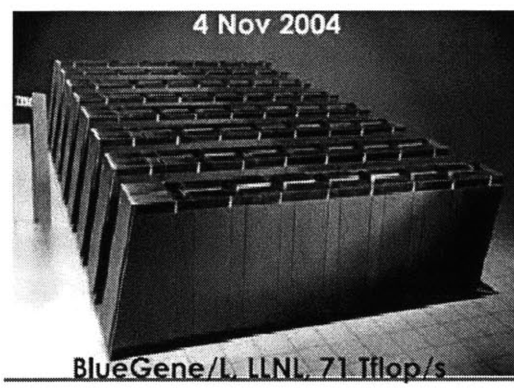


Fig. 32: Blue Gene/L

The largest planned Blue Gene/L machine, which is scheduled for delivery to Lawrence Livermore National Laboratory (LLNL) in California in early 2005, will occupy 64 full racks, with a peak performance of 360 teraflops. The next generation of Blue Gene/L will scale to over 1 petaflops and the third generation will span up to multiple petaflops (see Fig. 33, [43]).

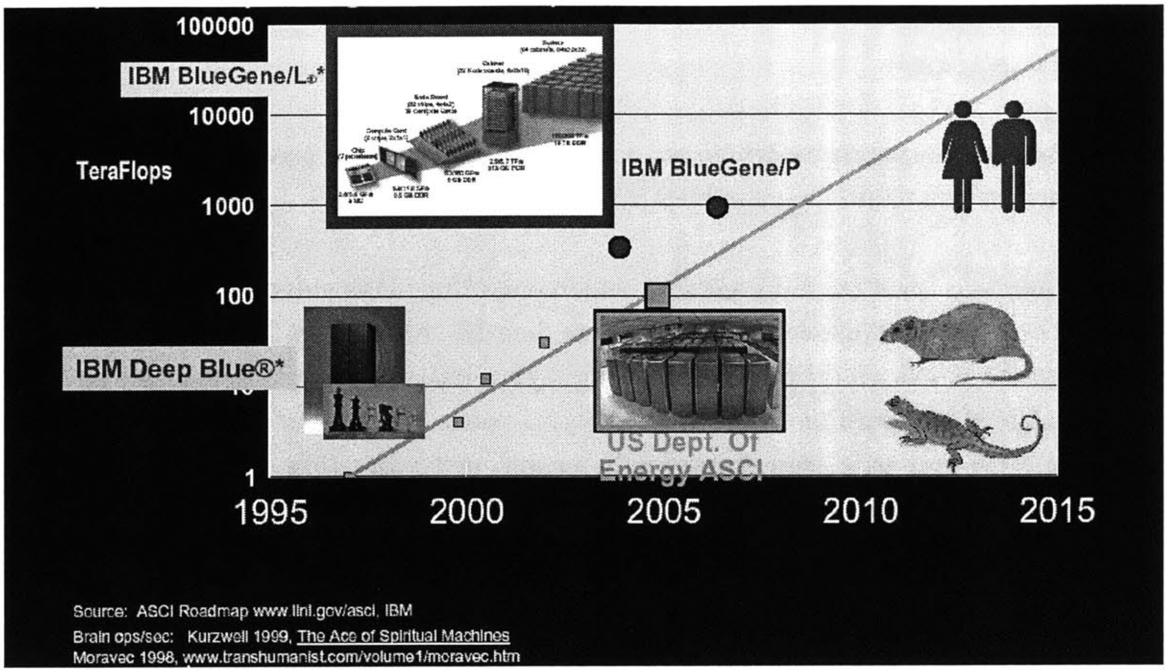


Fig. 33: IBM Supercomputing Roadmap

Similar plans come from Cray Inc., one of the historical leaders in supercomputing [44]

The U.S. Government's Accelerated Strategic Computing Initiative ASCI program (a driving force behind supercomputing advances for more than five years, has set the goal of achieving petaflop-level supercomputer performance by the year 2010.

As a highly reliable, easily scalable and cost effective alternatives and relatively new to the supercomputing realm of traditional supercomputers, cluster-based computers are also emerging. Clustered computers are comprised of multiple computers that are linked together via high-speed networks to form a single system. The collective system leverages its many computer processors to achieve supercomputer speeds. Clusters work by the "divide and conquer" philosophy. Complex computer calculations are divided into many parts. Individual nodes of the cluster are each sent a different part of the problem to solve. Once a node crunches its numbers, the results are combined with the answers provided by the other nodes to produce an aggregate solution to a request.

Cluster computers just work with normal desktops equipped with Pentium processors. Studies forecasts [45] that a Pentium 4, (which is now, in 2004, capable of few gigaflops), or its equivalent should deliver:

2005: 40 gigaflops

2010: 200 gigaflops
2013: 600 gigaflops

Several conventional processors might be harnessed even by a private individual to bring computations into the teraflops realm by 2013.

A TeraFlop was once the Holy Grail of supercomputing. This milestone was first achieved only in 1997 by Cray Computer at a cost of more than $80 Million. In 2001 there were only 12 computers in the whole world that ranked over 1 Teraflop and these systems cost on average greater than $20M per teraflop. In 2003, after only 2 years, the cost per teraflop has plummeted to less than $1M and will drop dramatically in the next years (Fig. 34, [43]), so teraflop computing is going to be soon low-cost.

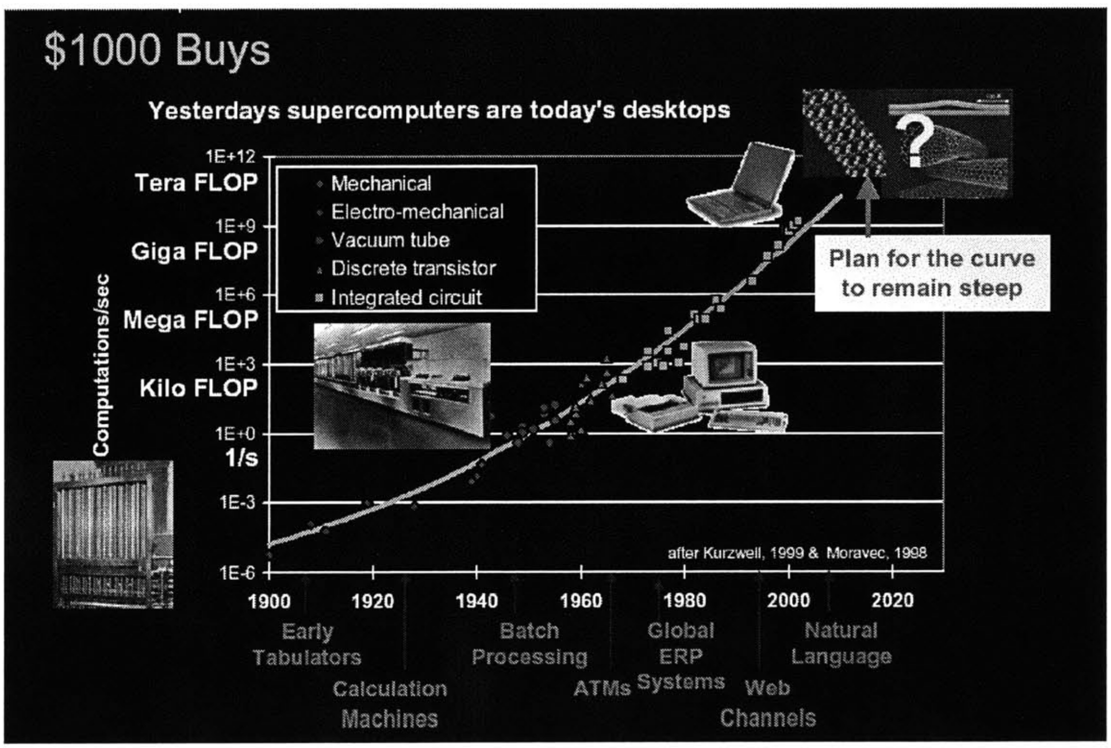


Fig. 34: Evolution of FLOPs

This will enable new opportunities in MDO, because it will mitigate the trade-off between model fidelity and breadth of design spaces searching which has been an important limiting factor in the past.

2.5. Examples of application of MDO as a design tool

The combination of simulation and optimization, essentially unheard of in practice a decade ago, is much more accessible today, thanks in large part to the development of commercial optimization software designed for use with existing simulation packages. The increasing levels of high capability and cost effective HPC is contributing towards the widespread usage of high fidelity simulation models and tools as well as newer methods and technologies within the manufacturing industry.

In this Section, some recent examples of MDO practices in the automotive industry are given, with reference to generic automotive applications first and then, specifically for exhaust systems.

As a general comment, we note that the application of MDO techniques is not widespread yet and it's generally not part of the mainstream development process. In addition, only few disciplines are usually considered, two or three at most, and economics is not among them.

2.5.1. Automotive Industry

A Ford Experience: MDO of a vehicle system for safety, NVH (noise, vibration and harshness) and weight [46]

The focus of this work is on an automotive vehicle system design optimization for safety and NVH. As far as safety is concerned, the following conditions are evaluated:

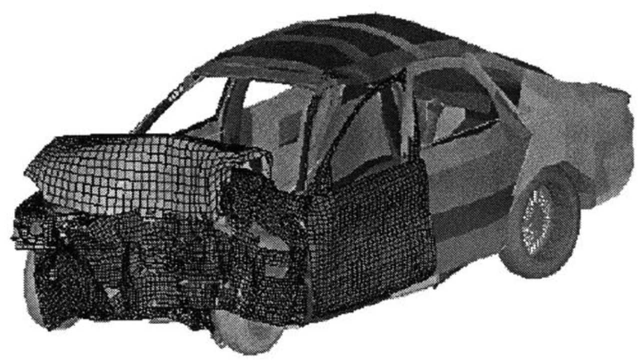


Fig. 35: CAE model for Frontal Crash

- Frontal Crash. The vehicle crashes into a rigid 90 degree fixed barrier with the speed of 35 MPH (Fig. 35). The key safety performance measures in the full frontal crash include occupant Head Injury Criteria (HIC) and Chest G. Full frontal crash is commonly used to

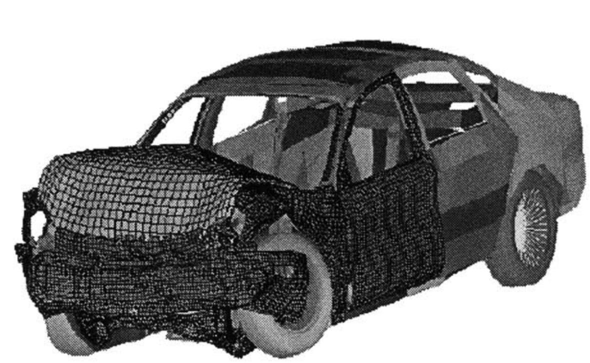


Fig. 36: CAE Model for 50% Offset Frontal Crash

design and validate the vehicle front structures. Federal Motor Vehicle Safety Standards 208 (FMVSS) clearly specifies the safety regulations and test configuration. The regulation states that the HIC and Chest G injury numbers have to be within 1000 and 60g. The design targets for the full frontal impact in this study are not only to satisfy FMVSS 208 regulation but also to comply with corporate guideline. In this work, the occupant HIC and Chest G numbers are targeted to be less than 450 and 45 respectively.

- 50% Frontal Offset Crash. The vehicle is set to crash into a 90-degree fixed rigid wall with 50% offset (Fig. 36). The impact velocity is 40 mph. The design target for toe board intrusion is set to be less than 10 inches.
- Roof Crush. Vehicle roof crush is a federal mandatory requirement intended to enhance passenger protection during a rollover event. The test procedure is defined in FMVSS 216. In roof crush simulation, the ram normal speed is set to be 7.5 MPH. As described in the FMVSS 216, the force generated by vehicle resistance must be greater than 5,000 lbs. (22,240 N) or 1.5 times the vehicle weight, which ever is less, through 5 inches of ram displacement (Fig. 37). In this study, the roof crush resistant force is set to be 6,000 lbs.

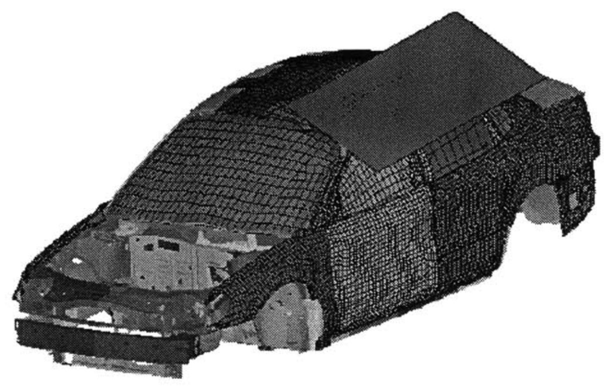


Fig. 37: CAE Model for Roof Crash

- Side Impact. For side impact protection, the vehicle design should meet the requirements for the National Highway Traffic Safety Administration (NHTSA) side impact procedure (Federal Motor Vehicle Safety Standards 214) or European Enhanced Vehicle-Safety Committee (EEVC) side impact procedure. The dummy performance is the main concern in side impact, which includes head injury criterion (HIC), chest V*C's

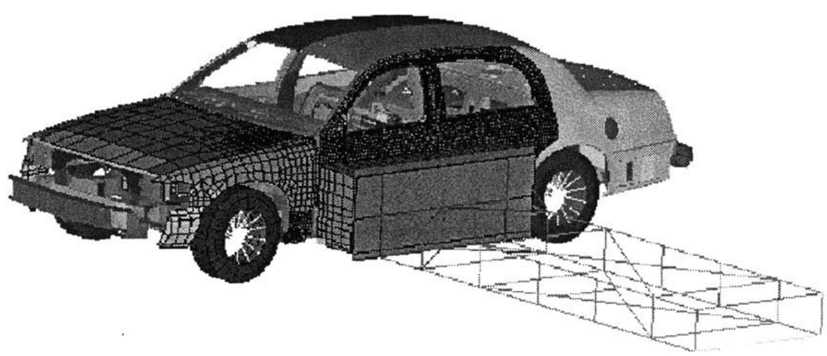


Fig. 38: CAE Model for Side Impact

(viscous criterion) and rib deflections (upper, middle and lower). These dummy responses must at least meet EEVC requirements (Fig. 38). Other concerns in side impact design are the velocity of the B-Pillar at the middle point and the velocity of front door at B-Pillar. For side impact, the increase of gage design variables tends to a get better dummy performance. However, it also increases vehicle weight, which is undesirable. Therefore, a balance must be sought between weight reduction and safety concerns. The objective is to reduce the weight while satisfying safety constraints on the dummy. The dummy safety performance is usually measured by EEVC side impact safety rating score. In the EEVC side impact safety rating system, the safety rating score depends on four measurements of the dummy: HIC, abdomen load, rib deflection or V*C, and pubic symphysis force.

- NVH: The torsion frequency for the Body In Prime free-free normal mode is set to increase by 5% from the baseline 26.5 to 27.8 Hz. The upper bounds for static torsion and static bending displacements are chosen as 3.4 mm and 0.9 mm, respectively, i.e., 10% improvement from the initial design.

Different models are used for different purposes so that the quality of the simulation results is high and the cost is at minimum. The optimization problem is involving the "disciplines" of NVH and Safety and it's set as follows:

Given the set of vehicle system design variables X, find X in order to minimize the Weight of the Vehicle System Structure, while satisfying:

- NVH: Static torsion & bending displacements Frequency (Mode3) $26.65 < \omega_3 < 29.32$ Hz
- Crash Characteristics:
    - Frontal Crash: Dummy HIC (Head Injury Criterion), Dummy Chest G, Probability of severe injury
    - 50% Frontal Offset Crash: Intrusion at several key locations
    - Roof Crush: Maximum resistance force
    - Side Impact: Displacements at several key locations, Viscous Criterion, Bounds on the design variables, X and Z

In this MDO task, the NVH discipline has 19 local design variables while the safety disciplines combined have 25 local design variables. In addition, 10-system design variables are common to both the NVH and crash disciplines, that gives a total of 54 design variables, which are primarily sizing (thickness) variables and spring stiffness.

The above MDO problem is solved using a variation of the OMDAA (Optimization by a Mix of Dissimilar Analysis and Approximations) method and a Sequential Quadratic Programming optimizer is used to solve the numerical optimization.

A commercial Optimization package is used (ModelCenter) for integrating the different tools/component (including spreadsheets).

System	Attribute	Baseline	Target	CYCLE 1 Solution	CYCLE 2 Solution
NVH	Frequency - Mode3 (Hz)	26.5	$27.8 \leq f3 \leq 29.3$	27.8	27.9
	Torsion disp1 (mm)	3.8	≤ 3.4	3.39	3.38
	Torsion disp2 (mm)	-3.8	≥ -3.4	-3.41	-3.40
	Bending disp (mm)	-0.97	≥ -0.9	-0.90	-0.90
FRONTAL CRASH	HIC	500	≤ 450	378	426
	Chest G	42	≤ 45	35.9	36.9
	Ptotal (%)	10	≤ 10	6.8	7.5
OFFSET CRASH	Intrusion 1 (in.)	11.2	≤ 10	8.9	9.0
	Intrusion 2 (in.)	10.8	≤ 10	9.9	10.0
	Intrusion 3 (in.)	10.9	≤ 10	9.6	9.5
	Intrusion 4 (in.)	10.1	≤ 10	8.7	8.7
	Intrusion 5 (in.)	10.5	≤ 10	9.1	8.8
ROOF CRUSH	Resistance Force (kN)	34.7	≥ 27	35.2	32.8
SIDE IMPACT	VC 1	0.48	≤ 0.54	0.48	0.47
	VC 2	0.51	≤ 0.54	0.53	0.52
	VC 2	0.54	≤ 0.54	0.55	0.54
	Disp2 (mm)	26.3	≤ 27.2	25.9	25.8
	Disp3 (mm)	27.2	≤ 27.2	26.7	26.6
SYSTEM	Weight (kg)	1740.5	Minimize	1740.8	1739.1

Tab. 4: Safety & NVH MDO Results

The MDO problem results are provided in Tab. 4 for 2 cycles of the optimization process. The initial design is an infeasible design with NVH and Safety constraint violations of over 10% from the target. The final design is feasible without any adverse impact on the system objective, weight of the car body.

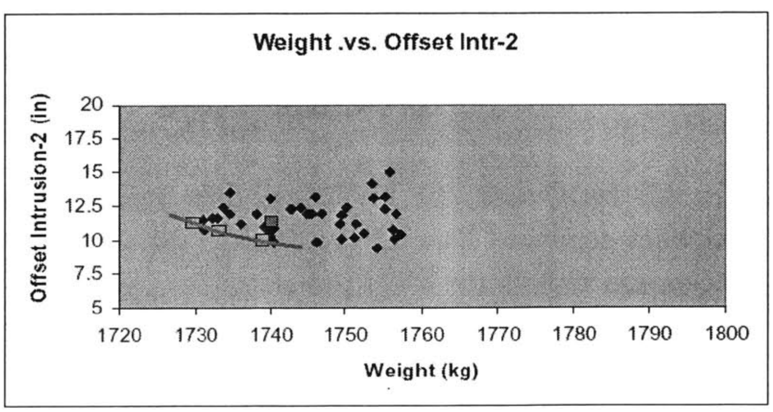


Fig. 39: Weight Vs. Offest Intrusion

Since many different criteria are involved disciplinary trade-off information was drawn for deciding how best to balance the various criteria to arrive at the most desirable design. Certain trade-off analyses are performed for the system objective with respect to the active design constraints and the results are used in constructing Pareto optimal curves and surfaces.

Fig. 39 shows the trade-off between the vehicle weight and offset crash response – intrusion - that is an active constraint in the MDO problem. The dots represent the sample set of design points using the Latin Hypercube sampling method. Fig. 40 provides the trade-off between the same offset crash response – intrusion and an active NVH response (torsion displacement).

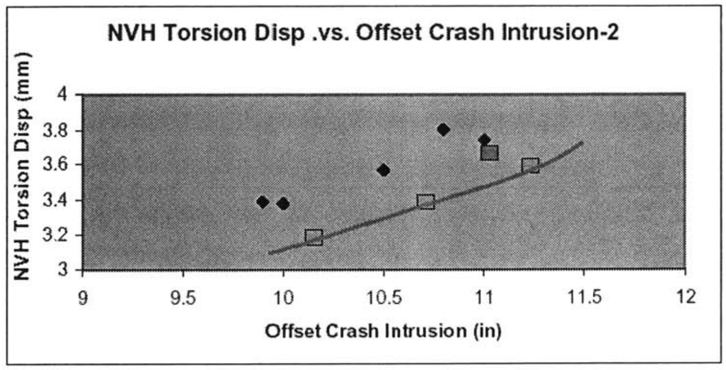


Fig. 40: Torsion Displacement Vs. Offset Crash Intrusion

The MDO problem with NVH and multiple safety systems (frontal, offset, roof and side impact) would require close to 3 years of elapsed computing time on a single processor of the type of an Origin 2000 server. On the Origin 3800 server, with 256 processors, these 3 years of elapsed computing time is compressed to less than 2 days using a combination of fine and coarse grain modes of parallelism.

Fig. 41: The Ferrari 360 Modena

Aerodynamic optimization procedure at Ferrari [47]

On a Ferrari 360 Modena (Fig. 41), Ferrari tried for the first time the vehicle external aerodynamics optimization which had, as objective, to minimize the aerodynamic drag while maximizing the downward force, taking a series of constraints into account, mainly style (max 3 cm displacement from the baseline

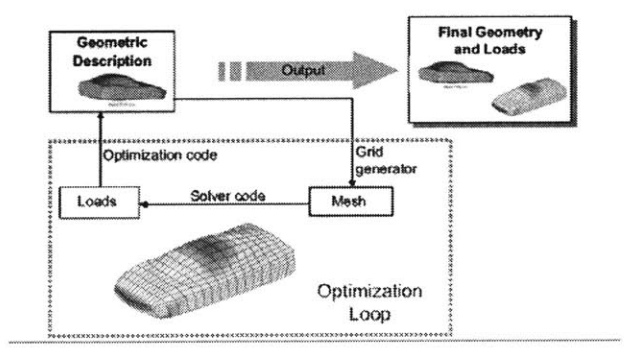


Fig. 42: Scheme of the aerodinamic optimization procedure

styling shape) and technological (i.e. manufacturing feasibility).

Since the evaluation of the aerodynamic loads must be repeated many times, the aerodynamic solver must be "inexpensive" as regards computational time, but yet sufficiently accurate. The conventional code for the solution of the Navier-Stokes equations was then discarded and a simplified modified "potential" method was worked out. The results from the potential method were verified against experimental data and parameters were tuned for the application.

One day of computational time was enough to achieve an improved design. A prototype has been built and tested in the wind tunnel. The results are presented in Tab. 5 and show a consistent improvement in the design performance.

	Baseline geometry	Optimized Geometry
Cz	0.176	0.141
Cx	0.185	0.181
Fz (load variation @ 280 km/h)	0	-28 kg (equivalent to a 0.16 x 1.5 m wing)
Fx (load variation @ 280 km/h)	0	-3.5 kg (equivalent to 3 hp more)

Tab. 5: Validation Results: Wind Tunnel Experimental Data

What's worth noting is that, after the successful application of the MDO process on the 360 Modena, the aerodynamic optimization procedure has been introduced in the standard design process of any new Ferrari car.

Performance-Cost Tradeoffs for Engine Manifold Surface Finishing [48]

In this work, the link between manufacturing process and product performance is studied in order to construct analytical, quantifiable criteria for the introduction of new engine technologies and processes. Knowing, in fact, the trade-off between the cost of the new process and the realizable profit stemming from improved performance enables a proper business decision.

The Abrasive Fluid Machining (AFM) technology for finishing the inner surfaces of intake manifolds is studied. AFM

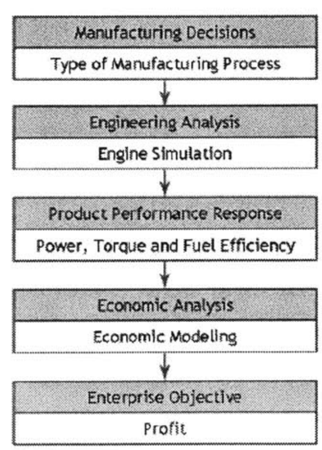


Fig. 43: AFM Evaluation Flowchart

process employs a viscoelastic medium impregnated with grit to smooth inner surfaces of metal parts. This process is very effective in reducing the roughness of cast-iron or cast-aluminum components, such as inner surfaces of engine manifolds. Improved finish leads to better performance through reduced flow losses and improved engine volumetric efficiency, i.e. better filling of engine cylinders with fresh charge. An additional benefit is reduced variability between cylinders and thus more accurate engine calibration.

The basic assumption is that choosing such a manufacturing process and increasing product performance in turn impacts product's demand. The firm's profitability is then used as a criterion for decision-making. Fig. 43 shows a flowchart depicting different analyses.

The engine used in this study is a V6 2.5L Spark-Ignition (SI) engine with four valves per cylinder. The air intake manifold directs the flow of air from the throttle body to the intake valves. The intake manifold selected for this study is made of aluminum alloy. As shown in Fig. 44, air flows into the manifold through a single large orifice, and is then divided into twelve "runners" that lead to the intake valves.

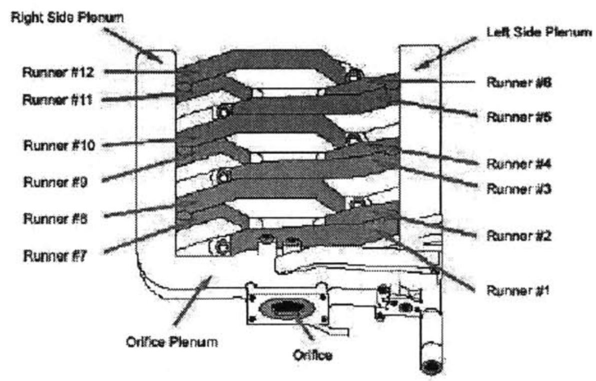


Fig. 44: Sketch of the Air Intake used for AFM Evaluation

Measurements and statistical analyses were done to characterize the surface finish and the related roughness. Gt-Power, a 1-D fluid dynamic code, was used to compute the engine performances.

An initial sensitivity study is performed on the effect of the AFM process to engine performance. Design of experiments is used to assess variation of power, torque and fuel economy caused by variation of surface roughness in the runners, plenums and orifice of the manifold with and without AFM. To avoid computational burden, the sensitivity study is performed with three input variables: roughness of all runners, roughness of plenums and roughness of the orifice. Latin Hypercube sampling was used to generate a 20-point sample. Tab. 6 shows performance enhancements gained by applying AFM technology. Both power and torque have been improved by about two percent while BSFC improvements were negligible; hence the focus of the analysis is on power.

	μ	σ
Runners Surface Roughness (*μm*)	82.6%	69.4%
Plenums Surface Roughness (*μm*)	86.7%	76%
Orifice Surface Roughness (*μm*)	85.3%	85.8%
Power (*hp*)	2.03%	0.014%
Torque (*ft-lb*)	2.03%	0%
BSFC (*lb/hph*)	0.29%	0%

Tab. 6: Effect of Surface Finish on Engine Performances

A model, which translates the engineering trade-off(s) to a microeconomics or/and financial optimization problem, has then been developed.

Following Hazelrigg [49] , engineering decisions affect product performance attributes, which in turn affect the demand of the product. In the specific case, the surface roughness decision for the intake manifold of the engine influences horsepower, a product characteristic observed by the consumer, and hence it would affect product demand. More precisely, it can be shown that horsepower (*HP*) / vehicle weight (*w*) ratio is the relevant system attribute. The demand is also affected by price. However, to focus on performance influence, we assume the firm will keep the price constant.

Elasticity of demand q with respect of the increase in performance over vehicle weight $E_{\frac{HP}{w}} = \frac{\%\Delta q}{\%\Delta HP/w}$ is then computed, relying on the work of Berry [50] . A change in quantity translates then to a change in revenue.

The average total cost of the AFM process is known and estimated at $5 per horsepower gained per car.

Having modelled revenue and cost, profit can then be calculated, which will be used as the criterion for decision-making.

The optimization problem is then set up as follows:

Maximize: Profit
with respect to: surface roughness

Under the assumption that no part dominates the engine performance, the surface roughness at each manifold location is equally weighted. Therefore, the decision variable is the average sand surface roughness for all runners, orifice and plenums. Upper and lower bound values for power, torque, BSFC, and surface roughness are set.

The DIvided RECTangles (DIRECT) optimization algorithm is used. DIRECT can solve mixed-integer nonlinear programming problems and locate global minima efficiently without derivative information, when the number of variables is small, as in this case. DIRECT starts at the center of the user supplied design space, divides it in rectangles and evaluates the objective function at the centre points of these rectangles. Based on the objective function value and the characteristic dimension associated with each rectangle, DIRECT selects which rectangles to further divide until it reaches the specified number of function evaluations. This ensures that the entire space is searched in sufficient

	Compact Car	SUV
Profit ($)	-204,647	313,551
Roughness (*μm*)	125.21	6.96
Power (*hp*)	163.98	167.04
Torque (*ft-lb*)	137.8	140.3
BSFC (*lb/hp-h*)	0.4897	0.4882

Fig. 45: AFM MDO Analysis Results

granularity in order to explore more promising areas in more detail.

The optimization problem is solved for the two extremes in an automotive manufacturer's fleet: compact car and sport utility vehicle. Profit margins per unit (PMU), namely selling price per unit minus average total cost per unit, are given in Fig. 45.

The decision model suggests application of the AFM process only in the SUV segment as opposed to both. This is the interpretation of the results recommending 90% surface roughness reduction for the SUV engine manifold and a negligible reduction for the compact car manifold. This result is no surprise. Although demand for acceleration is higher for compact cars the current profit margin level does not motivate the firm to innovate. In case of the SUV segment, the firm will offset the increased average total cost with higher profits.

In this case, MDO has been used as a decision support tool.

2.5.2. Exhaust System Related Applications

Examples of MDO can be also found with relation to exhaust system design. Three of the most recent and interesting ones are presented below.

Multidisciplinary optimization of the aero-acoustics and of the structural dynamic characteristics of an exhaust system [51]

In this work, a method and an integrated virtual design system enabling multidisciplinary optimization of the aero-acoustic characteristics, the backpressure characteristics and the structural dynamic response of exhaust systems for a five cylinders, four-stroke engine, Fig. 46, is performed.

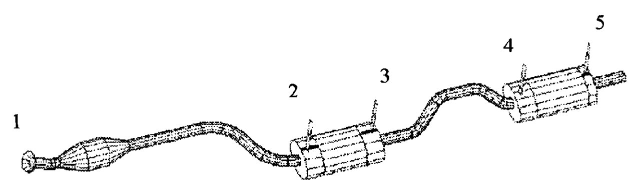


Fig. 46: The Exhaust System Model

The two numerical disciplines used are the FEM for structural dynamic analysis and Computational Fluid Dynamics for the aero-acoustic analysis and pressure analysis. The FEM code MSC.Nastran and the CFD code Wave by Ricardo are used for the analyses, I-DEAS is used for the modelling and iSIGHT for the optimization tasks.

The engine torque fluctuation under driving conditions excite some of the exhaust system eigenmodes to resonances and the generated forces excite the car body where the exhaust system is suspended. This could influence the ride comfort of the vehicle through imposing undesirable vibrations. The focus here is in improving the ride comfort by minimizing the vibrations at certain positions in the car body. The FEA strategy is to perform a normal modes extraction and afterwards a modal frequency response analysis.

The acoustic characteristics of the exhaust system are also of great importance as the sound pressure level generated from a car is regulated through laws and low noise is a competitive factor in the automotive industry today. Another important factor in exhaust system design is the backpressure, which is the static pressure exerted by the exhaust system on the engine and should generally be kept as low as possible. Wave uses a 1-D flow assumption and simulates the dynamics of pressure waves, mass flow and energy losses in ducts, offering CPU cost savings compared to 3-D codes.

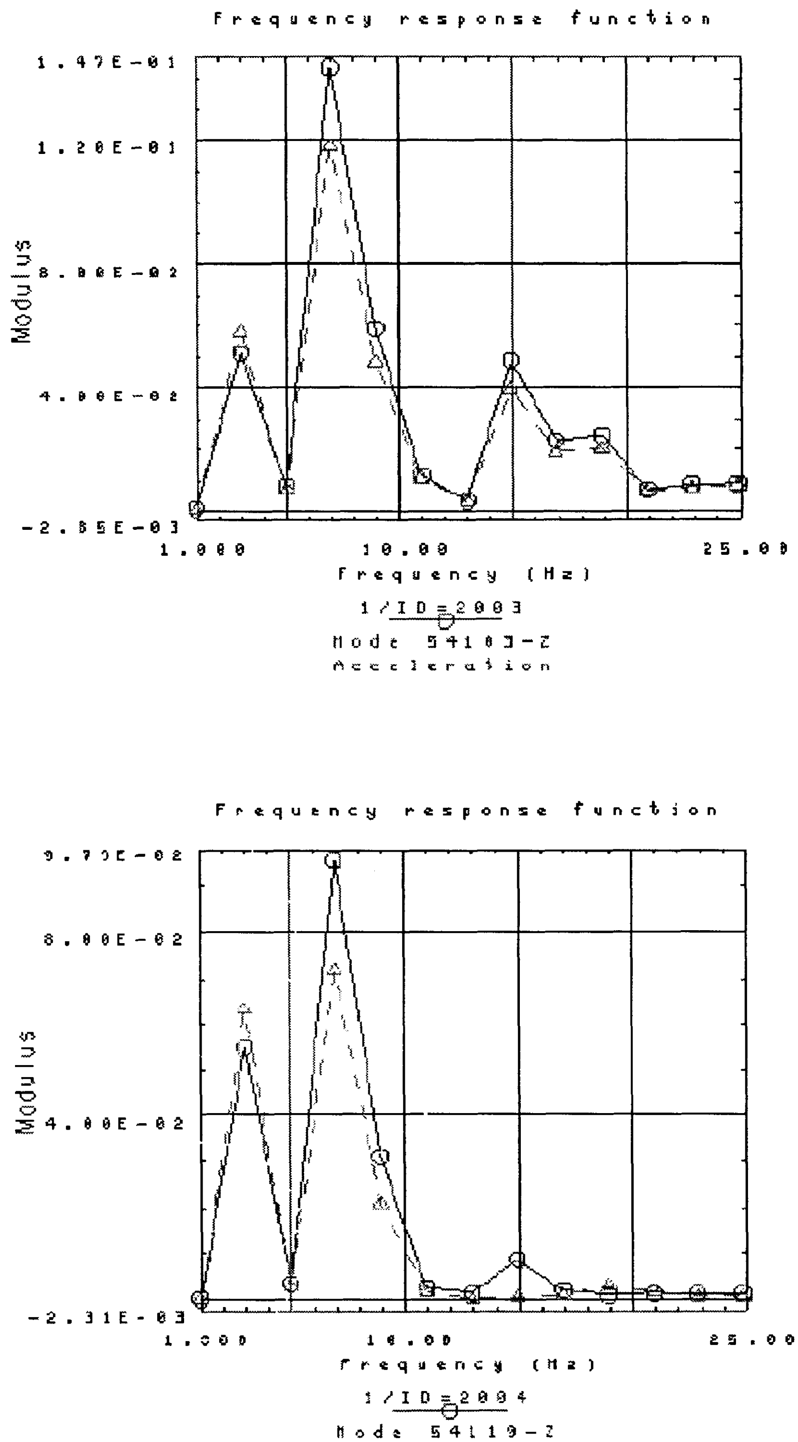


Fig. 47: Acceleration Results at points 3 (above) and 4 (below)

One parameterized geometry model is

used as the base for work in all disciplines. The solid geometry is described by geometric design variables. When those are changed, the simulation models are updated automatically. Special developments have been made to automate the steps and the data transfer in the optimization loop. The width, height and length of the first and second muffler and the position of the four suspension points are chosen as design variables.

In this work the Modified Method of Feasible Directions (MMFD) is used as the optimization algorithm. The algorithm uses gradients and handles non-linear problems.

The objective is to minimize the dynamical forces at the suspension points, given a certain sound pressure level at the exhaust system tailpipe. As the exhaust system has to be fitted to the car body, there exist several constraints on the geometrical design variables.

A selected set of results of the optimization (acceleration levels at two suspension points) is presented in Fig. 47. The solid lines represent the baseline, while the dashed ones represent the optimized levels.

The optimized geometry has smaller muffler volumes and the second suspension point is moved away from the centre line of the exhaust system. The result is that the dynamical forces decreased in all suspension points except in one.

Acoustic Optimization of Exhaust Systems – the experience of ArvinMeritor [52]

As a first tier supplier to the automotive industry, ArvinMeritor is exposed to the continued drive to shorten development times for new vehicles. Representative engines and vehicles may not become available until late in the project leaving less time for the traditional build and test development cycle. To support the shorter development times there has been a move to make extensive use of CAE predictive tools.

Tools are developed to simultaneously design for tail pipe noise, system backpressure, system weight and sound quality. Two cases are presented.

<u>Muffler Design.</u>

When optimizing a muffler design there are many design features that may be modified, such as pipe diameters, internal baffle locations, the perforation characteristics, the presence or absence of absorption material and so on, Fig. 48.

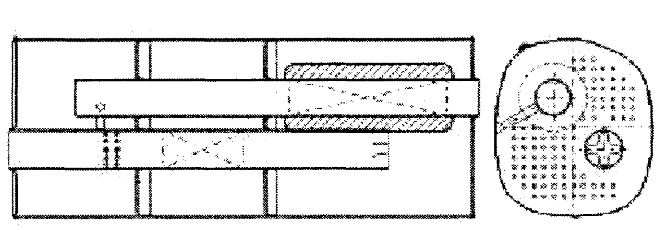


Fig. 48: Sketch of Muffler Internals

For the case study project, the

objective was to optimize a baseline muffler design attenuating an intrusive idle noise resonance (16th to 21st order for an I4 petrol engine). A target noise reduction was set based on an objective of achieving an improvement at 200 Hz of around +15 dB. This target level was arrived at following a subjective noise assessment of an artificially filtered idle noise recording where it was found that a 10 dB improvement in the idle noise resonance would be acceptable.

The acoustic modelling software that was used to predict the acoustic performance for each muffler was LAMPS. A software, named QED (Quick Exhaust Design) was developed by ArvinMeritor to perform optimum search computations. This software is able to automatically write and solve the acoustic models by interfacing with LAMPS. Controls were put in place to maintain backpressure and muffler volumes at the baseline level.

Optimization is achieved via genetic algorithms. Designs are solved in batches (generations) then compared against some fitness function. Each population contains a fixed population size, which can vary from as few as twelve up to hundreds. The fitter designs may proceed to the next generation but weaker designs do not. Consequently, the average population fitness tends to increase from one generation to the next.

The design generated by the QED software was calculated overnight running on four PCs in parallel solving some 120,000 LAMPS simulations in total.

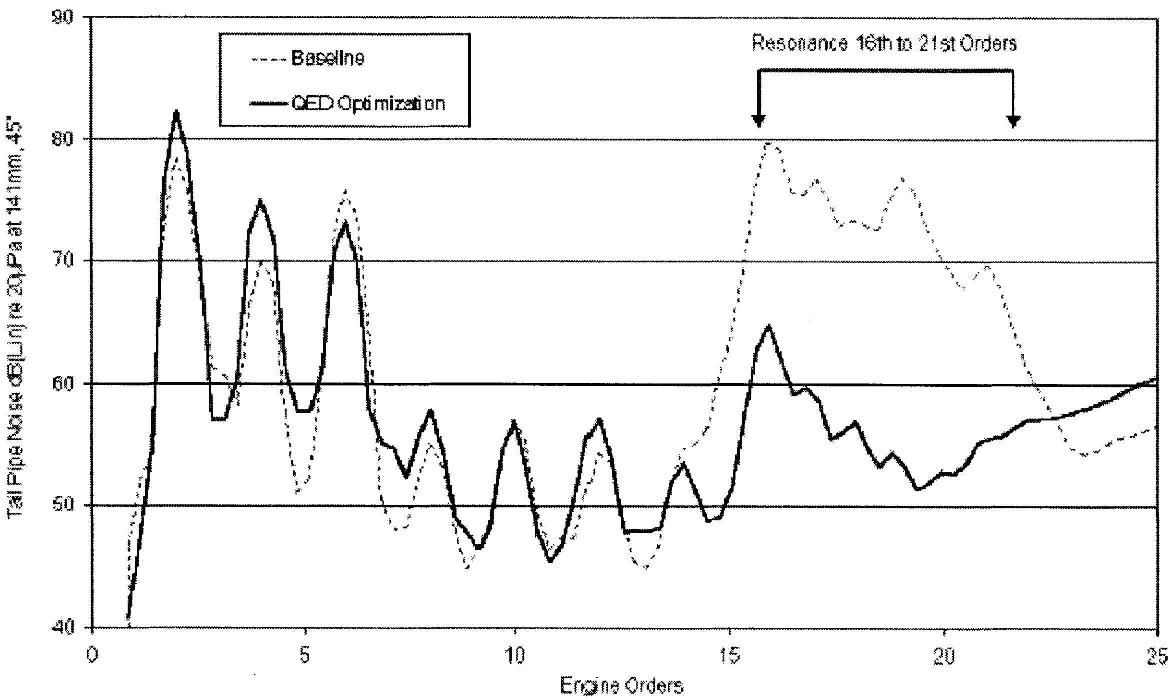


Fig. 49: Tailpipe Noise Results: baseline and optimized muffler

The muffler coming out of the optimization process was then built and tested on the vehicle to measure idle noise and wide-open throttle accelerations. The results from the idle noise test are shown in Fig. 49. The problem resonance (16^{th} to 21^{st} orders) is clearly visible in the baseline system. The optimized design is around 15 dB lower in the resonant order range, and a subjective assessment of the idle noise concluded that it was now acceptable.

Manifold Optimization Tuning

The subject of this optimization case is tuning a 4-into-1 exhaust manifold on an I4 petrol engine to obtain a sporty sound. For a smooth, even, power delivery from all cylinders, the primary pipes of an exhaust system are usually designed to have the same length. This produces an exhaust note dominated by 2^{nd}, 4^{th} and 6^{th} orders often called the even orders. For a sporty sound it has been shown to be necessary to increase the proportion of 3.5 and 4.5 orders. The goal set for this optimization was therefore to minimize the difference between second order versus 3.5 order and second order versus 4.5 order. For this analysis, the four primary pipe diameters and lengths were allowed to vary independently across the limits of bore = 28 mm to 60 mm and length = 230 mm to 470 mm.

These eight parameters were built into a test array and the different manifold configurations analyzed in WAVE using a Design of Experiments (DoE) technique. The 8 dimensional DoE matrix was created using the Latin Hyper Cube across 300 test points. This resulted in 300 manifold designs with a random distribution across the design space. Each manifold was solved in WAVE at three engine speeds of 1600, 3600 and 5600 rpm (wide-open throttle). To solve the test array took 45 hours of processing time. The data obtained were used to generate a Response Surface Model (RSM) to be used in the subsequent optimization phase

Following the building of the RSM, a search for an optimum arrangement within the design space, which best suits some target function is made. The project objective was to increase the 3.5 engine order (3.5E) relative to 2^{nd} engine order (2E). The optimization function was weighted heavily towards the 1600 and 3600 rpm since flow noise can mask the contribution

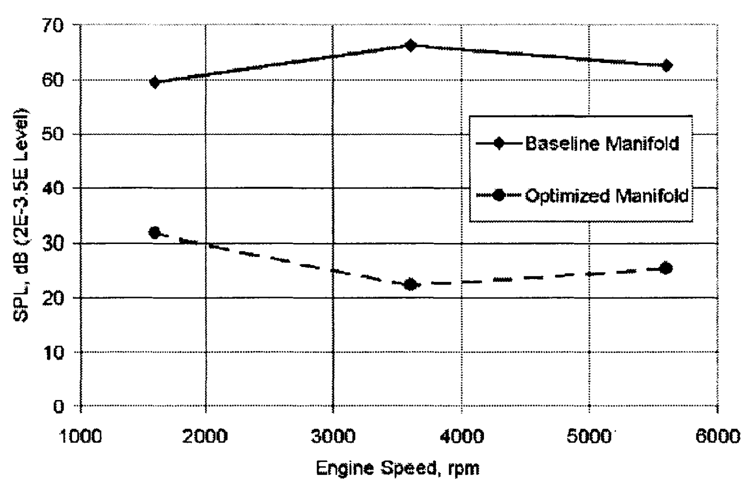


Fig. 50: 2nd-3.5 Engine Order Sound Pressure Level difference: baseline and optimized manifold

from engine orders at high engine speeds (wide-open throttle). The commercial code OPTIMUS was used to perform the optimization using a hill descent search strategy.

The comparison of the tg_35E function predicted by WAVE for the baseline and optimized designs is presented in Fig. 50. The baseline model shows a 60-65 dB difference between 2E and 3.5E since the equal primary length are extremely effective at suppressing half order noise. However, the optimized manifold order balance shown in Fig. 50 demonstrates some 30-40 dB improvement when compared against the baseline.

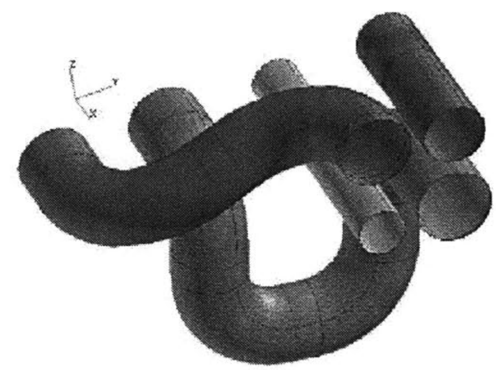


Fig. 51: Optimized Manifold Layout for Sporty Sound

The manifold geometry for the optimized design is illustrated in Fig. 51. No restraint was placed on the engine power or the balance of power between cylinders since the objective of this analysis was to maximize sporty sound only.

The output from WAVE is a time domain pressure pulsation, which has then been converted into a sound file for subjective appraisal. The difference between the baseline and optimized designs is immediately obvious when listening to the sound files, with the sporty manifold having a definite rumble or uneven nature, which conveys a powerful or sporty character to the exhaust note.

Exhaust Manifold optimization for engine power and catalyst inlet temperature [53]

Car engines today are required not only to have more engine power, but also to be more environmentally friendly. Exhaust gas should be kept at high temperature in the exhaust pipe especially at low rpm conditions because the catalyst located at the end of the exhaust pipe will absorb more pollutant in high temperature conditions. Exhaust gas should also be led from the piston chambers to the exhaust manifold smoothly to maximize the engine power especially at high rpm conditions.

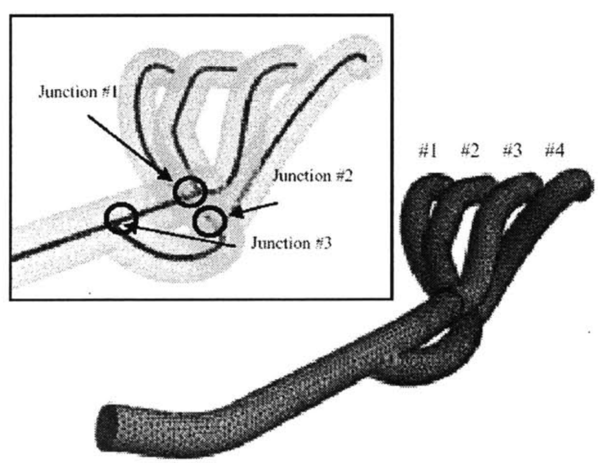


Fig. 52: The initial manifold shape and design variables as junction positions on pipe centerlines

In this work, the high power engine of a sports car is considered for multi-objective optimization to increase the engine power as well as to reduce the environmental impact. The objective functions considered here are to maximize the gas temperature at the end of the exhaust pipe at 1,500 rpm and to maximize the charging efficiency at 6,000 rpm, where the charging efficiency indicates the engine power.

The initial manifold shape is taken from an existing engine with four pistons as shown in Fig. 52. Topology of the merging configuration is kept unchanged.

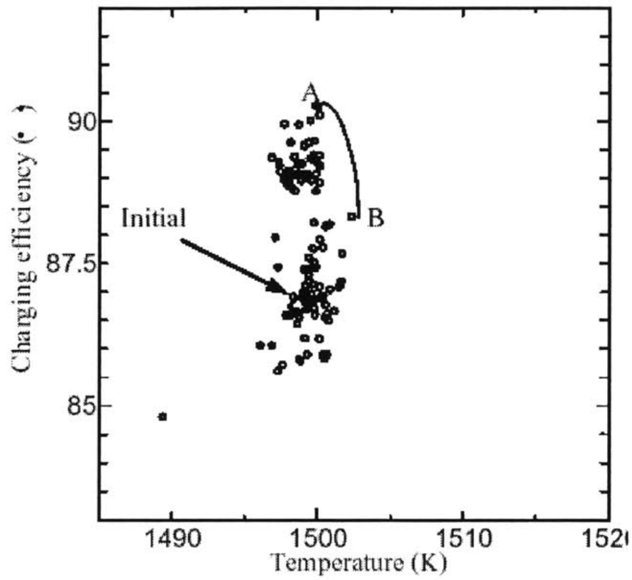
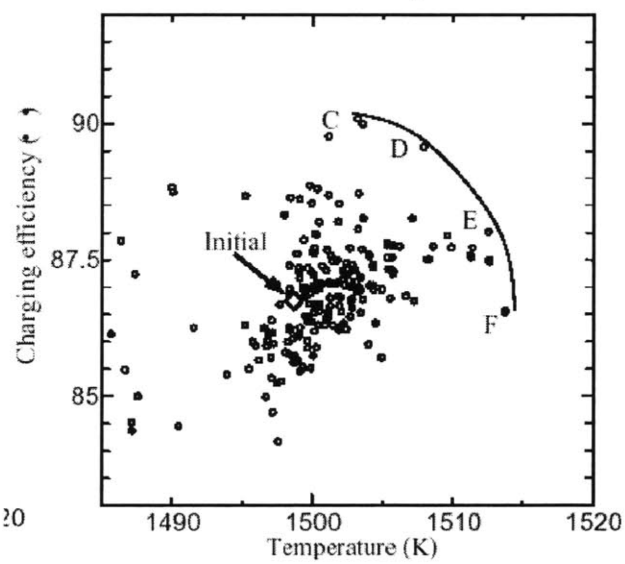


Fig. 53: Solutions plotted in the objective function space; Case 1, merging points optimization

The pipe shape travelling from the port #2 to the outlet is also fixed. Three merging points on the pipe centerlines, junctions #1-3, are considered as design variables. Pipe centerlines of #1, 3 and 4 are then deformed similarly from the initial shapes to meet the designed merging points. This method allows the automated grid generation for arbitrary merging configuration defined by the pipe centerlines.

A Genetic Algorithm was used as the optimization algorithm.

This study considered two design cases. The first case assumes a constant pipe radius for all pipes; therefore only three merging points are to be designed. In this case, the population size was set to 32 and the evolution was advanced for 25 generations. In the second case, the pipe radii of the entire exhaust manifold are considered a design variable because the pipe radius is known important for the performance of the exhaust manifold from experiences in industry. The pipe radius will change from 83% to 122% of the original radius. In the second case, three merging points and the pipe radius are to be designed



g. 54: Solutions plotted in the objective function space; Case 2, merging points and pipe radius optimization

simultaneously. In this case, the population size was set to 64. The evolution was advanced for 29 generations.

In Case 1, Pareto solutions were found as shown in Fig. 53. Many solutions achieve much higher charging efficiency than the initial geometry. These results suggest that the merging points are effective design variables to improve in the charging efficiency that indicates the engine power. However, the improvement in the temperature remained marginal.

In Case 2, Pareto solutions were found as shown in Fig. 54. Improvements in both objective functions were achieved. The Pareto front also confirms the trade-off between the two objectives. This result suggests that the pipe radius is effective to maximize the temperature at the end of the exhaust manifold.

3. BUILDING THE ENHANCED DEVELOPMENT FRAMEWORK

3.1. Introduction

To prove the effectiveness of the product development computerization featuring multi-disciplinary analysis, we built a prototype of an EDF. After a testing phase of the tool, we simulated to be in an OEM setting and be charged to develop a maniverter for a specific application, recording the performances of the tool in terms of development lead-time and product attributes levels.

We chose, for this demonstrator, a real application that ArvinMeritor was asked to develop in 2001 by Fiat and that, for several reasons, technical and commercial, ArvinMeritor did not have the opportunity to bring to market: a maniverter for the Fiat Fire 1.4 16V engine. After three years of development, the project was actually stopped in 2004.

The three years development, the many solutions tried and the different issues emerged during the project are assumed as a representative sample of what could happen in the course of a development project. If we the EDF available for development at that time, would the project have evolved differently?

This chapter starts with the illustration of the maniverter as a system and of its design requirements. Then, after a brief description of the Fiat Fire 1.4 16V engine, overview of the Enhancement Development Framework is made. The architecture is presented first and each of the modules is examined in detail afterwards. The Chapter continues with the implementation of the architecture, where how the different modules are assembled in a coordinated whole is shown. At last the capabilities of the framework and the post-processing techniques of its output for decision-making purposes are described.

The tests of the tool and the actual results are, on the other hand, a subject of Chapter 4.

3.2. Application: the IC engine exhaust system maniverter

3.2.1. Background

Any Internal Combustion (IC) engine is equipped with an exhaust system. A typical example is shown in Fig. 55.

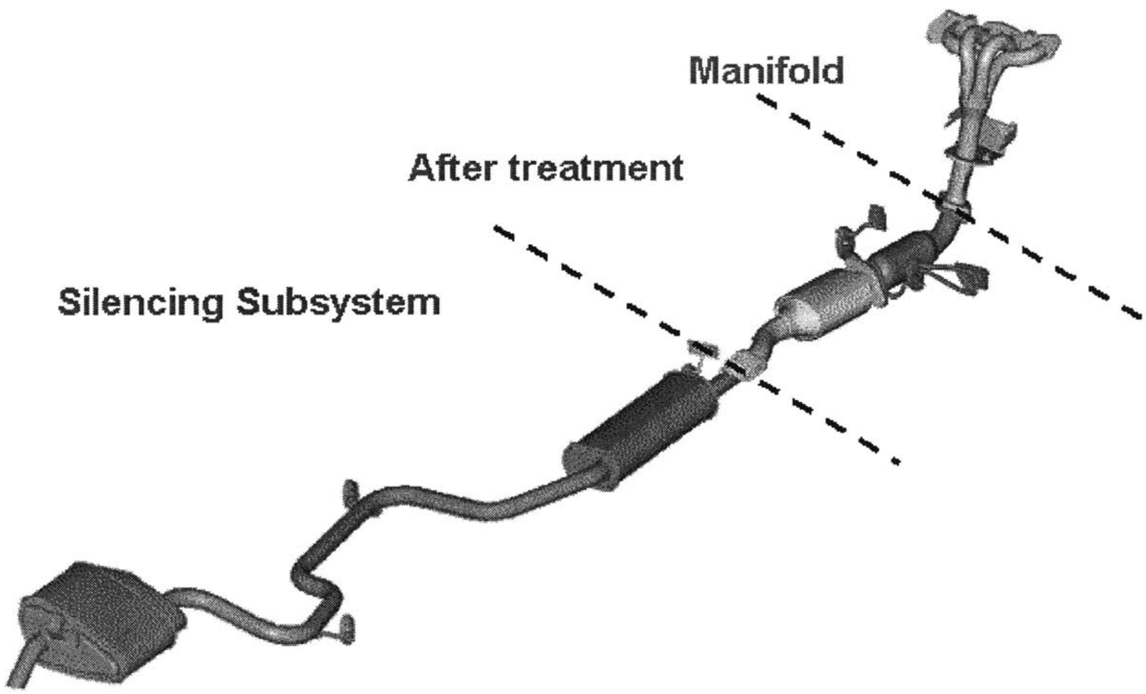


Fig. 55: An Example of a Full Exhaust System

The exhaust system carries exhaust gases from the engine's combustion chamber to the atmosphere. Exhaust gases leave the engine in a pipework, travelling through an after-treatment sub-system, which often consists of a catalytic converter, and then through a silencing sub-system before exiting through the tailpipe. Chemical reactions inside the catalytic converter change most of the hazardous hydrocarbons and carbon monoxide produced by the engine into water vapour and carbon dioxide, while the muffler attenuates the noise produced by the engine.

The conventional muffler is an enclosed metal tube packed with sound-deadening material. Most conventional mufflers are round or oval-shaped with an inlet and outlet pipe at either end. Some contain partitions to help reduce engine noise futher.

The exhaust manifold, in particular, is the first stage of the exhaust system. It conducts the exhaust gases from the combustion chambers to the exhaust pipe. Some exhaust manifolds are made from cast iron or nodular iron, while others are made from stainless steel or heavy-gauge steel. In this work, we will assume that the manifold is made from stainless steel, which is the technology that ArvinMeritor masters.

The exhaust manifold contains an piperun for each exhaust port in the cylinder head, and a flat machined surface on this manifold fits against a mating surface on the exhaust port area in the cylinder head.

Some exhaust manifolds have a gasket between the manifold and the cylinder head.

The exhaust passages from each port in the manifold join into a common single passage before they reach the manifold flange. An exhaust pipe is connected to the exhaust manifold flange. [54]

Sometimes, a catalytic converter is moved upstream from the traditional underfloor position and is placed just after the point where pipes coming out of the engine ports join. This particular position is selected in order to achieve a reduction in the converter warm up time after the engine is cranked-up and consequently to speed-up the start of pollutant conversion. In this case, quite often the term "maniverter" (manifold+converter) is used (for an example, see Fig. 56.

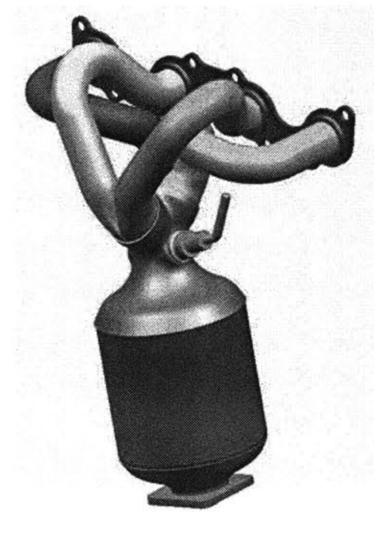


Fig. 56: An Example of a Maniverter

The design of an exhaust system maniverter is the result of a complex trade-off among different and equally important requirements:

- Exhaust gases should be kept at a high temperature in the exhaust pipework especially at low rpm conditions when engine starts because, in higher temperature conditions, the catalyst will "light-off", i.e. start converting pollutants, earlier
- For a non-sporty application, such as the one in study, the engine torque curve should be as even as possible, reaching the maximum value at the lowest rpm and maintaining it as long as possible throughout the working rpm range.
- The engine should have the highest possible torque level and, consequently, power for a given rpm. This translates in two requirements for the maniverter: the pressure drop of the gas through the ductwork should be minimized and the manifold should be "tuned". Tuning is achieved when the manifold pipework is such that the pressure waves that originate at the exhaust valves, after propagating through it, are reflected back and come at the exhaust valves as a depression, aiding the scavenging of the engine and therefore increasing the power. Tuning is particularly effective for a naturally aspirated engine and it's often possible only at a certain rpm.
- The manifold should be designed in order for the sound emitted at the tailpipe to have a characteristic "color". This is particularly important for sporty vehicles, which are required to exhibit a characteristic rumble.
- Manifold system natural frequencies should not lie in the excitation frequency range of engine vibrations. If that happened, in fact, the manifold could resonate and generate

unpleasant noise. With time, vibrations would transform in fatigue failure. For a 4-cylinder 4-stroke engine, it can be shown that the forcing vibration frequency in Hz is given by rpm/30 (the so-called second-order frequency).

- In any case, the manifold structure should maintain a sufficient stiffness to avoid localized resonances and, consequently, unacceptable radiated noise
- Thermal stresses arising from the thermal expansion that occurs when the manifold heats-up should be kept lower than the yield stress of the material, otherwise the plastic strain that occurs when the manifold expands will soon degenerate in a crack.
- Similarly, stresses generated from the vibrations induced by the engine should be below the fatigue limit of the material at the temperature working conditions, otherwise a fatigue failure is expected.
- The exhaust system manifold should be fitted in the available space in the engine compartment and sufficient clearance for assembly tooling access in the production plant should be ensured.
- The manifold surface temperature and distance from the surrounding components should be such that the latter, particularly those components made by plastics or rubber, are not exposed to a temperature that exceeds the maximum working limit allowed by material properties
- The manifold mass should be as low as possible to enhance the fuel consumption characteristics of the vehicle, its driveability and CO_2 emission
- Manifold pipework should be designed to allow the gas stream to impinge on the surface of the converter with a flow velocity distribution as even as possible to improve emissions reduction and moreover, to ensure a longer converter life
- Similarly, flow velocity of the impinging gas should not exceed a certain threshold level above which the converter damages
- The manifold geometry (pipework, catalyst inlet and outlet cones, etc) has to satisfy the requirements of manufacturability and assemblability with the available equipment of both the OEM and of the supply chain firms.
- Last but not least, manifold cost should allow the exhaust system manufacturer having a competitive price in the marketplace, while preserving or enhancing its product margins.

Each of those requirements put particular strain on the design and may drive different design solutions. The interaction between the effects on the performance attributes of the different design choices is not, generally speaking, evident and experience is usually the only valid guide for a cost-effective successful design.

The following table lists the several different disciplines[1] that are involved in the design of a maniverter and the engineering issues they manage:

Engineering disciplines	Development issues	Departments
Geometry	Clearances, mass and manufacturing feasibility	CAD Dept.
Structure mechanics:	Vibrational behaviour, thermal and vibration-induced stresses, radiated noise	CAE Dept.
Fluid dynamics	Pressure drop, manifold tuning, flow distribution on the converter brick, max flow velocity, airborne noise	Fluid Dynamic group
Heat management	Gas temperature in front of the catalyst, for the radiated heat to the surrounding components	Thermal group
Acoustics	Tailpipe noise, radiated noise	Acoustic Dept.
Costing	Maniverter cost	Cost engineering

Tab. 7: Maniverter Engineering Issues and related Disciplines

In the common practice the Program Manager handles the different issues separately with each department and tries to find the right compromise between the different needs.

3.2.2. Fiat Fire 1.4 16V engine

In the present work we will assume that the maniverter is tailored to a specific application and engine: the Fiat Fire 1.4 16V. However, results are generalizable to any 4-cylinder engine.

In this Section, few descriptive information about the engine are presented.

The Fire engine, which started to equip Fiat vehicles (Punto, Stilo and others) in 2003, offers a cylinder capacity of 1368 cc and a 4 cylinder in line configuration with a bore of 72 millimetres and stroke of 84 mm. The four valves per cylinder are driven directly by two overhead camshafts, Fig. 57.

The power unit was developed with particular attention to performance and fuel consumption. Volumetric efficiency has been optimised throughout the service range due to painstaking fluid dynamic development studies on the entire intake and timing system. The

[1] An engineering discipline is a branch of the engineering knowledge. To be qualified as a discipline, though, it must possess the following six basic characteristics: a focus of study, a world view or paradigm, a set of reference disciplines used to establish the discipline, principles and practices associated with the discipline, an active research or theory development agenda, the deployment of education and promotion of professionalism. [55]

result is a power output of 70 kW (95 bhp) at 5800 rpm and a maximum torque of 13.0 kgm at 4500 rpm.

Manufacturer: Fiat Type: S-4 Wet sumped DOHC 16 valves total 4 valves per cylinder Bore x stroke: 72.00mm × 84.00mm Bore / stroke ratio: 0.86 Displacement: 1368 cc 83.48 cu in Compression: 11.00:1 Fuel system: MPFi Aspiration: Normal Catalytic Converter: Y Max. output: 94.3 PS (93.0 bhp) (69.4 kW)@5800 rpm Max. torque: 128.0 Nm (94 lbft) (13.1 kgm)@4500 rpm Coolant: Water Specific output: 68 bhp/litre

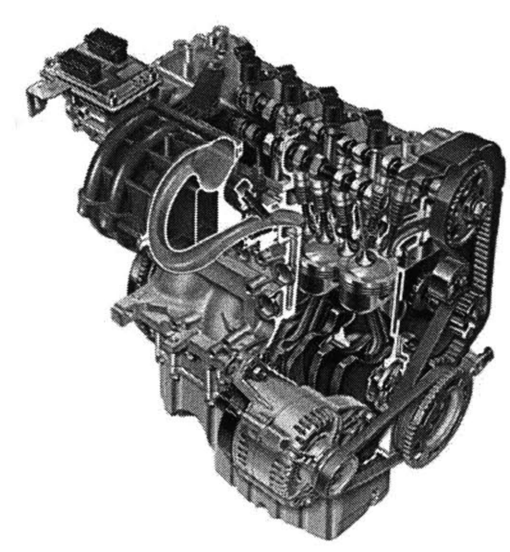


Fig. 57: The Fiat Fire 1.4 Engine

This performance is obtained also thanks to an electronic throttle valve control system known as a drive by wire system. The 95 bhp 1.4 unit uses new engine control unit management software. This torque-based system represents the cutting edge in its field. Its strength lies in being able to manage all actions through a single co-ordinator block that operates according to one basic parameter, i.e. the driver's torque requirements expressed through the accelerator. When translated into a physical torque value, these demands (including the demands of external systems such as the ABS) may be coordinated even before the main engine control parameters have been converted (advance, throttle position, injection time etc.) with the huge benefit of meeting needs with extraordinary accuracy and within a very short time period. Not to mention the fact that this system exploits a single standard of communication between the various systems and functions that all speak the lingua franca of drive torque. This allows a higher level of handling than with current systems while also reducing polluting emissions levels. The system also guarantees maximum integration with all the other devices such as ESP and Cruise Control.

Another specific feature of the new 95 bhp 1.4 16v Fire is the increase in compression ratio and the high torque values at low speeds, qualities that have allowed fuel consumption to be kept low. This aim is also achieved through the tuning of the cutting edge engine control system that succeeds in cutting fuel consumption as far as possible while still maintaining handling, performance and low emissions. As far as emissions are concerned, the 95 bhp 1.4

16v already meets Euro 4 legislative requirements. This is due to a catalytic converter located in the engine compartment (and welded to the exhaust emission manifold flange) that reaches high temperatures within a shorter time period and thus reduces emissions even while the engine is warming up. To minimise the environmental effect, the new engine is also equipped with a returnless fuel system that eliminates fuel recirculation within the tank and thus reduces vapour formation.

High-performing, thrifty and clean: the 95 bhp 1.4 16v Fire unit backs these qualities with outstanding acoustic comfort. Firstly, a barycentric power unit mounting system has been adopted to achieve reaction forces with zero offset and thus minimise the transfer of engine vibrations to the body. The acoustic comfort offered by the new engine is also enhanced by:

- An aluminium crankcase base with cast iron main bearing caps cast together;
- The development of an aluminium oil sump that is connected directly to the crankcase base and gearbox to increase the flexural and torsional rigidity of the entire power unit and thus reduce vibrations;
- The use of a damper with setting specially adjusted to damp vibrations with torsional resonance in the crankcase and flywheel system;
- Lastly, the adoption of an optimised piston skirt profile on which is deposited (screen-printed) a molybdenum bisulphate coating that allows piston/liner mating clearances to be pared to the minimum possible during production. This reduces noise produced by secondary movement of the piston in the cylinder (piston slap).

3.3. Hardware / Software Platform for ICE environment

Decision has been made to built the ICE platform on a laptop with Pentium 4 Processor, 1GB RAM and 15GB HD with Microsoft Windows 2000 Professional SP4. Several are the reasons for this choice. Among the most important are:

- Today a Pentium processor supplies enough computing power for many applications, even for CAD or CAE packages, traditionally run on powerful Unix workstations
- A laptop enhances mobility and communication

3.4. Scope Definition

Compared with other MDO approaches, our work brings some novelty because it has the aim to create a tool intended to be used not by specialist, but by design engineers in mainstream development.

A tool for manifolds development, ideally, should include all the design aspects illustrated in 3.2.1. That's why we strived to include as many aspects we could. Indeed, we carefully scoped the activity in order to create a framework which is adequately representative of the real environment but whose complexity is not so high to impede any progress. For that purpose, we excluded aspects that were either of minor importance, or for which explicit knowledge did not exist within ArvinMeritor or that would require the use of too computationally intensive calculations incompatible with the selected hardware platform. In addition, a deliberate decision was made to use commercially available software.

Details of the design aspect included and excluded in the prototype EDF are presented in Tab. 8.

Design aspect	**Included**	**Motivation for exclusion (if excluded) or software used (if included)**
Catalyst Inlet temperature	✓	1-D Fluid Dynamic code (AVL BOOST)
Torque evenness	✓	1-D Fluid Dynamic code (AVL BOOST)
Max torque	✓	1-D Fluid Dynamic code (AVL BOOST)
Min Backpressure	✓	1-D Fluid Dynamic code (AVL BOOST)
Tuning	✓	1-D Fluid Dynamic code (AVL BOOST)
First natural frequency	✓	Finite Element Code (MSC.Nastran)
Fit in the available space	✓	CAD package (Unigraphics NX2)
Thermal induced-stresses	×	Computationally too expensive
Vibration induced-stresses	×	Computationally too expensive
Temperature of surrounding components	×	Secondary aspect
Mass	✓	CAD package (Unigraphics NX2)
Flow distribution on the converter surface	×	Computationally too expensive
Max flow velocity in the converter	×	Computationally too expensive
Manufcturability and assemblability	×	Usually based on experience. No software model available.
Sound quality	×	Secondary aspect
Radiated noise	×	Secondary aspect
Cost	✓	Spreadsheet (Microsoft Excel)

Tab. 8: Design Aspects Included and Excluded in the Prototype EDF

3.5. Architecture Definition

A model is a symbolic device built to simulate and predict aspects of behavior of a system.

Having identified the design aspects that we wanted to include and the individual disciplines softwares that are able to handle them, the next step was defining:

- How to insert them in a platform designed for automatic execution
- How they would interface with each other
- How the user would interact with the platform.

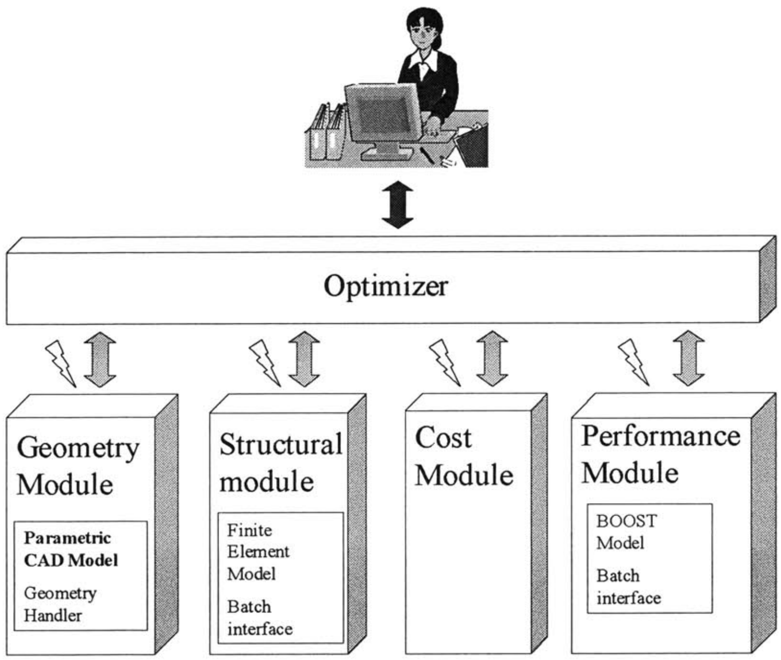


Fig. 58: Overview of the ICE platform Architecture:

⇳ data flow, ↯ client-server relationship

This set of architectural choices was partly driven by the selection of the code that forms the glue of the different disciplines packages, tying them together in a coordinated whole: the optimizer, Fig. 58. As mentioned in 1.3.2, iSIGHT from Engineous Software Inc. was selected for this role.

A detailed description of the EDF architecture will be given in 3.10, however we would like to outline here the essential features of the underlying ICE platform.

Through iSIGHT, a process flow of the different tasks (that correspond, in Fig. 58, with individual modules) is made, somewhat replicating the process in a normal product development environment: for example a CAD model is generated first and the structural modes analysis is performed afterwards.

Each task, generally speaking, requires an input and provides an output. Input and output are exchanged between the optimizer and the individual packages in the form of ASCII text files. Very little coupling exists between one module and the other.

From the data exchange standpoint, the platform architecture is therefore of a bus type, where the data bus is provided by the optimizer, Fig. 58.

The user interacts only with the optimizer.

From the execution standpoint, tasks are executed in a pre-determined sequence, as illustrated in Fig. 59. The user specifies the application (i.e. engine type and overall constraints) and the targets for the system performance attributes.

The product development definition starts with a selection of a baseline geometrical configuration and through the series of analysis the desired performance attributes are calculated. At the end of the analyses, the calculated attributes are compared with the target. If on target, the process ends, otherwise a new product definition is generated which has the potential of having performances closer to the specified target.

The loop features no iterations between the different modules because all Performance Modules depend on the Geometry Module, as evident from the dependency matrix shown in Fig. 60. This characteristic would enable also, in principle, parallel execution of the different Performance Modules.

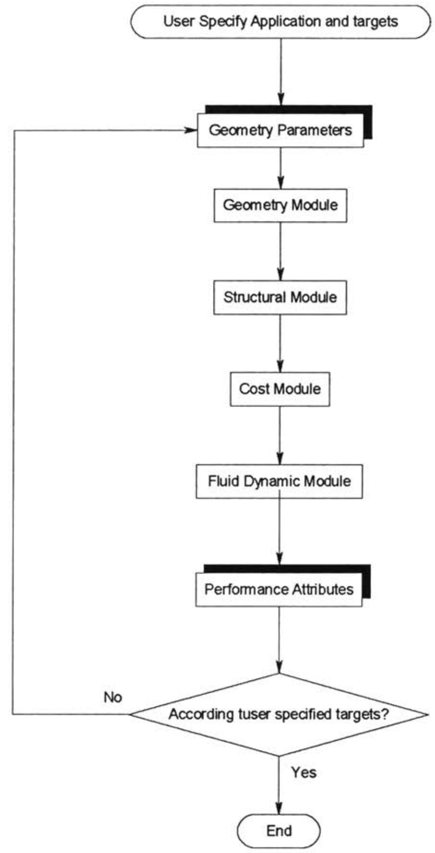


Fig. 59: Flowchart of the design loop execution

Task Name	Level		1	2	3	4	
Geometry Module	1	1					1
Structural Module	2	2	1				2
Cost Module	2	3	1				3
Fluid Dynamic Module	2	4	1				4
			1	2	3	4	

Fig. 60: Dependency matrix of different ICE modules

Two are the key elements that make this process possible:

- Optimization and simulation algorithms which build the correlation between design variables and performance attributes
- A product which is defined by a set of design variables

These are the essential features of the framework. The rest of the chapter is devoted to the description of the individual modules first and then to a detailed illustration of their integration.

As an introductory comment, we note that individual software packages, in general, are purposely designed for user interaction. Since we had to automate all the processes running all the calculations in batch mode, some customized development proved necessary. Each model, except for the cost model, an Excel spreadsheet for which iSIGHT has a direct interface, is composed of two parts: the model itself and the interface that handles the model and enables the batch execution, Fig. 58. It's the latter which interacts with directly with iSIGHT .

3.6. The Geometry Module

The Geometry Module is indeed the foundation of the ICE platform. All the analyses are based, in fact, on a unique product definition.

After a product conceptualization phase where appropriate simplifying assumptions are made to downsize design complexity to a manageable level, a parametric model is built which is able to represent a variety of maniverter configurations with a handful of parameters, the design variables.

A decision was made to utilize a commercial CAD package among the market leaders and, specifically, Unigraphics NX2 from UGS PLM Solutions. Special software was developed to handle the geometry regeneration in batch given a set of parameters and to extract some required data from the model.

This Section, after an introductory part on the status of the CAD technology and a digression on parametric modelling and on some of the relevant features of Unigraphics instrumental for the activity, describes in detail the product conceptualization, the parametric model and the geometry handler code for batch execution.

3.6.1. The Computer Aided Design

The inception of the CAD (Computer Aided Design) methods dates back to 1960 [56] however, after a decade of ferment, it's only in the 70s that they started to grow in popularity. Companies started to adopt CAD systems during the 80s and in the 90s the industry definitively took off[2].

After some decades, the use of CAD systems for geometrical modelling representation is mature and well established [57] . No dominant CAD standard exists and few players share the marketplace while many others have perished along the way.

---

[2] For a synthetic and yet well written history of CAD, consult http://accad.osu.edu/~waynec/history/lesson10.html

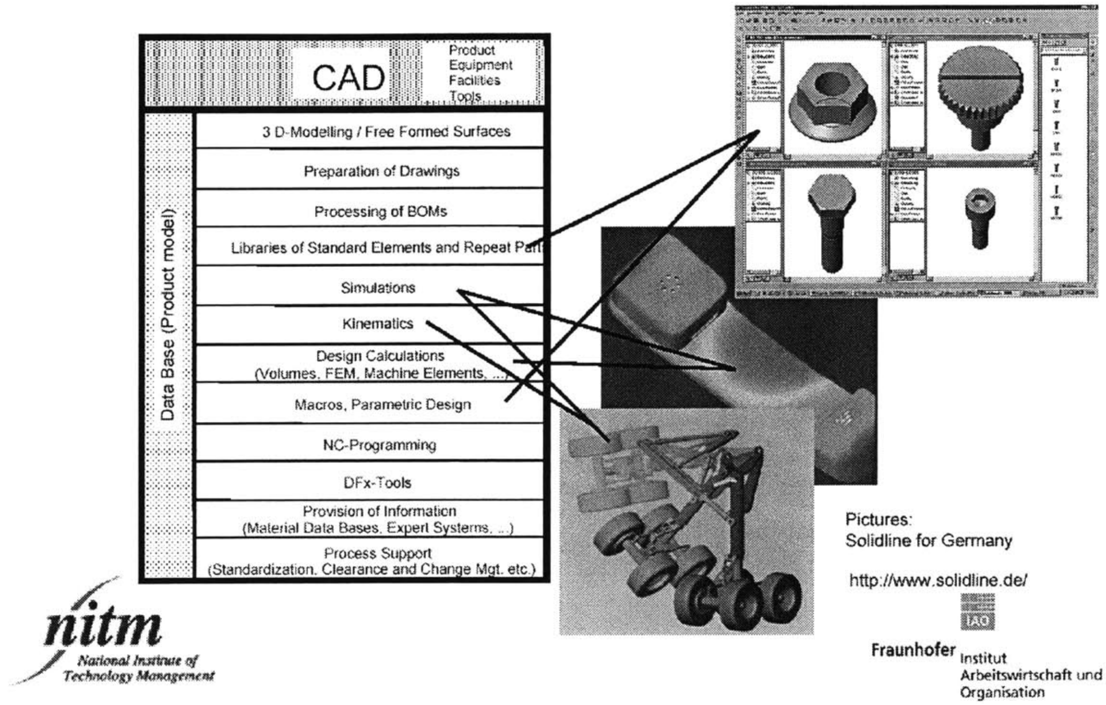


Fig. 61: CAD Functionalities

CAD capabilities have been increasing over time and span all the design activities, Fig. 61. In the Concept Design and Preliminary Design, the flexibility of CAD tools allow inexpensively liying down different solutions that could be subsequently analyzed to quickly get an estimation of the performances of the object. In the Detail Design phase, CAD models are refined to include manufacturability requirements and to allow more accurate analyses. CAD includes the possibility to link the 3D models of the parts to a Bill Of Material (BOM), thus favoring the interchange of information with the costing and manufacturing engineers. CAD models can then be transformed into CAM (Computer Aided Manufacturing) models for the final realization of physical prototypes and to verify the assembly in the production plant.

The use of CAD tools allows to reduce the number of physical parts that are built and consequently to reduce the leadtime and cost.

All CAD packages offer an intuitive Graphical User Interface (GUI) which improves the learning curve. However, despite continuous efforts to make them easier to use, CAD programs remain complex (and, some argue they are getting even more complex thanks to the also never-ending effort to add new features and ship new releases). In addition, quite often the ease of obtaining sensible geometries progressively disables the critical judgement of the designer on the goodness of the result.

3.6.2. Essential CAD capabilities: parametric / associative modelling and API

If CAD is nowadays an assessed tool of the modern industrial and research environment, the ways of using it can be profoundly different. One fundamental difference is between parametric and non-parametric – sometimes called explicit – modelling.

In parametric modelling, the dimensions of geometric entities are defined by parameters, i.e. numerical values or expressions. A particular instance of the geometry is obtained by specifying the values of the parameters. The user can change interactively parameter values and the CAD system will perform the geometry update following the geometry relationships defined during the model creation phase.

In an explicit model, vice versa, the dimensions of geometric entities are set when the entity is created and cannot be changed afterwards. The only alternative to change the entity is to actually delete it and to regenerate it with the new dimensions.

Parametric modelling offers superior flexibility; however, it is by and large more time-consuming than explicit modelling. The structure of the parametric model and the degrees of freedom of the geometry as well relationships among the parameters must be thought of at the outset. The complexity of their interrelation can soon become overwhelming. If product changes are expected to be only minor or infrequent, it may be more convenient to redo part of the model instead of building a delicate and complicated parametric model. Explicit modelling is, in fact, usually quicker and easier, and the skill set required by the CAD operator is usually of a lower profile.

Parametric and explicit modelling are not two separate worlds. Hybrid models can also be possible, with one part modelled parametrically and the other explicitly. Moreover, a parametric model could be easily transformed automatically in its explicit version. The backward way, i.e. from an explicit to a parametric model, is, on the contrary, not viable, if not for oversimplified shapes.

Parametric modelling is intimately linked to the concept of associativity. One element is associated to another if changes in the first element are automatically reflected on the other. For example, if a spline is defined by its poles, it's an associative spline if, given a change in the position of the poles, the spline modifies accordingly. CAD packages usually have the same geometric entity in an associative or non-associative form. For more information, refer to [58]

A parametric associative model is a key requirement for the ICE platform. What is needed, in fact, is a product fully defined by a set of values, the parameters, which can reproduce different product configurations in dependence of parameters' values.

Unfortunately parametric CAD tools are still not robust enough for representing complex geometries in a multidisciplinary environment. However, they are usually reasonably good to handle geometries of medium complexity.

When building a parametric model, of paramount importance is the early phase, when the relationships among the features are defined. Since all the elements are intimately intertwined, in fact, when the model is built, it becomes very difficult to make any changes without partly destroying the whole model. In addition, great care must be put in checking that the geometry can be regenerated with all combinations of the parameters within the validity ranges.

In these phases, the experience of the modeller is crucial. Modern CAD packages offer several ways to create the same geometric entity. Depending on the chosen, the resulting geometry may be more or less robust to parameter changes.

In addition to the capability of parametric / associative modelling, essential for the ICE platform is the presence in the CAD package of an Application Programming Interface (API) layer. The API is a set of routines, protocols, and tools for building software applications. API makes it possible to develop a program by providing all the building blocks. In the ICE, the geometry update according to parameters values as well as any data extraction (e.g. the mass) must be done in batch by a separate program. The API functions allow access by a program to the same functionalities that are available interactively. Currently the UG API provides access to over 4000 NX2 internal functions.

3.6.3. A particular CAD tool: Unigraphics by UG PLM Solutions

For this work the CAD tool Unigraphics NX2, by UGS PLM Solutions Inc., has been chosen (in short UG). Unigraphics is one of the leaders in the CAD industry. It shares the market with CATIA by Dassault Systems, Pro/Engineer by Parametric Technology and AutoCAD by Autodesk.

The greatest supporter and user of UG is undoubtedly General Motors.

As all the major CAD tools, Unigraphics is a complex and articulated package that allows managing the entire lifecycle of the product. It is composed of several modules, each of which provides a particular functionality: from the 2D drawing to solid modelling to the modelling of surfaces to assemblies' management.

Complex CAD tools require a vast amount of experience for a good result. And indeed the available experience has been the key driver for the choice of this product among the others, both the experience in parametric solid modelling and in the programming interface.

Hereafter is a non-exhaustive list of UG features that were particularly appreciated during the realization of the geometry module:

- "light" package, suitable for medium performance hardware such as a laptop
- fast execution of even complex operations
- efficient solid modelling
- intuitive navigation in the design tree

- possibility to manage parameters in a spreadsheet
- good associativity

Unigraphics NX2 is also one of the very few (another one is CATIA) which incorporates Knowledge Based Engineering (KBE) tools. KBE is currently in its infancy but it has the potential to change the way products are designed. As Evan Yares noted in Engineering Automation Report (July 2002), products and parts "are always designed based on functional requirements, but CAD products have historically been oriented toward designing based upon geometric requirements. Knowledge-driven automation tools provide a way for engineers to translate functional requirements into a geometric model capturing and manipulating engineering knowledge".

The Unigraphics KBE application, called Knowledge Fusion (KF), contains tools for capturing and manipulating engineering rules and design intent, so that they can be added into the design process. The rules extend beyond a purely geometric nature, and may involve engineering calculations, such as non-geometric physical properties, analysis results, sensitivity, processes, and much more. Some of its functionalities were used in the geometry handler module. Not only can engineers specify the requirements and rules that will drive the creation of the product, but designers also are free to make geometric model changes from within the CAD system – just as they normally would – and still have a model that is completely consistent with and associatively linked to the engineering rules. Central to Knowledge Fusion is the ability to capture NX entities, and represent these in the Knowledge Fusion language. A user can then easily extend the feature, adding knowledge about material, behavioral, or other characteristics. The resulting Knowledge Fusion feature works exactly the same as a native NX feature, but incorporates all of the additional information.

Because of its properties, KBE tools may the basis of a smarter CAD model to be used in next generations ICE platforms.

For more info on Knowledge Fusion, see [59]

3.6.4. Product Conceptualization

A maniverter is a complex product, which is tailored to a specific engine and vehicle. As such, many are the variants that could be found in the market. As an example, in Fig. 62, a gallery of manifold recently developed by ArvinMeritor is presented.

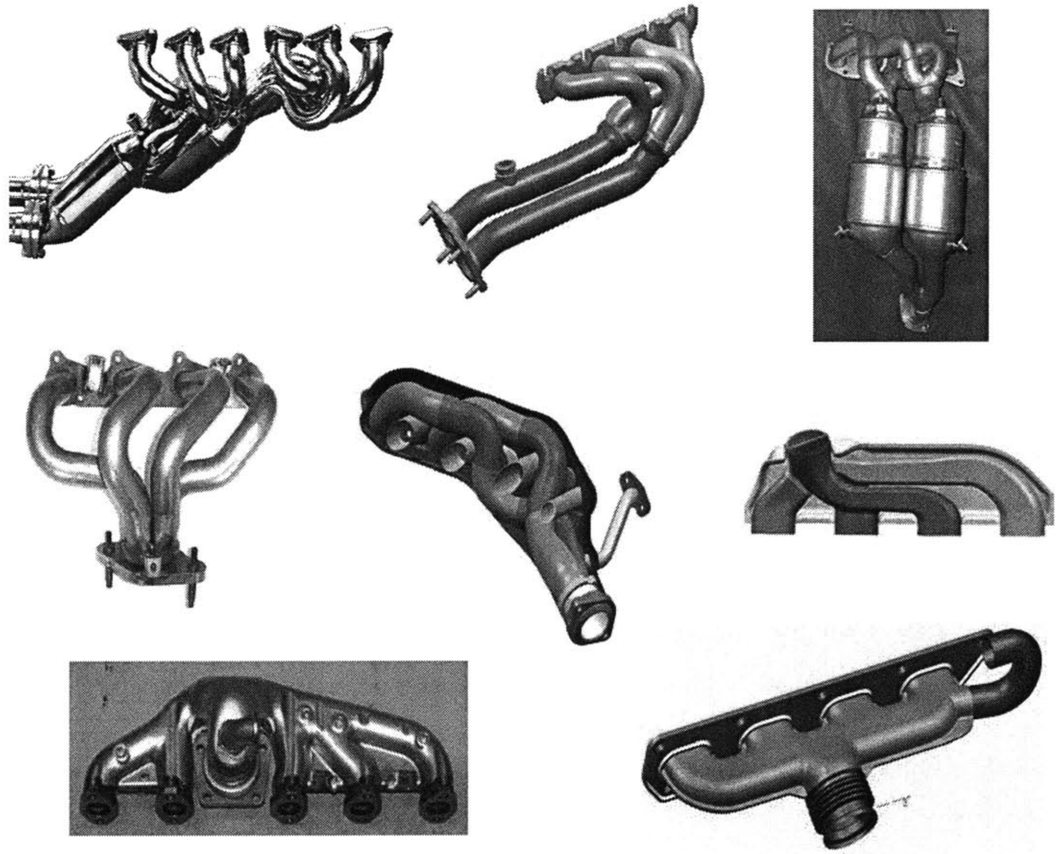


Fig. 62: A gallery of manifolds / maniverters designs

The architectural level differences among the various products can be grouped in the following categories:

- Number of cylinders of the engine
- Manufacturing technology: tubular, clamshell or hydroformed
- Single skin or airgapped
- Topology (number and type of junctions)
- Presence or absence of catalyst
- Material Type

In building a parametric model, ideally, the goal should be to have such a flexible architecture that is able to represent all these differences. However, the complication induced by tackling comprehensively the product diversity would make the project unviable or, at least, very complex to be properly debugged. Therefore, at the outset, a conscious simplifying decision has been made on what subset of products the MDO approach would have been applied. The aim has been to conceive a model that is simple enough to be managed within the given hardware / software / skill set constraints and yet that has as many features of real life manifolds as possible.

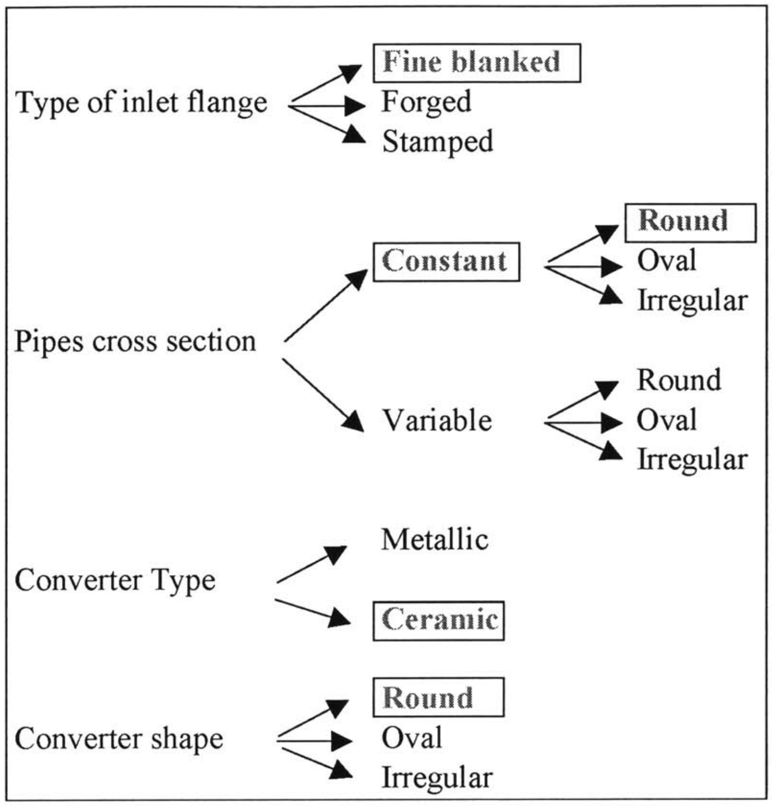


Fig. 63: Product Conceptualization Decision Tree

The choice was made to focus on a **<u>manifold product for a four cylinder engine, out of a single skin tubular technology with a 4>1 topology (i.e. four pipes joining into a single junction) and embedding a catalyst</u>**.

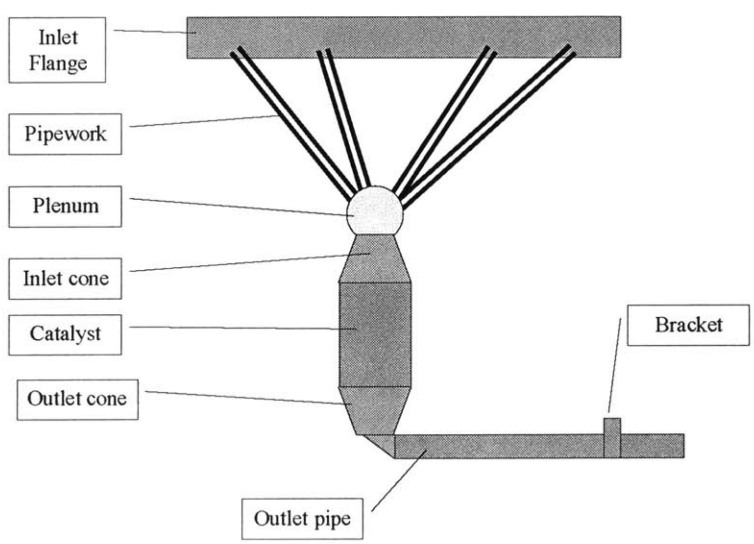


Fig. 64: Simplified Maniverter Concept

After these basic assumptions, several other decisions had to be made at the outset with the purpose to narrow down the complexity to a manageable level.

The decision tree is reported in

Fig. 63. The final choices are boxed.

In a nutshell, we've decided to represent a maniverter with constant section round pipes and with a fine blanked inlet flange. The resulting conceptual model is shown in Fig. 64. The four pipes are connected to the inlet flange (which is imagined bolted on the cylinder head) and join in a plenum where all the gas streams mix. Then an inlet cone leads the exhaust gases to the catalytic converter and, when they exit from it, they are guided to the outlet pipe through an outlet cone. A bracket connected to the engine block supports the maniverter. It is implicitly assumed that the maniverter is followed downstream by an exhaust system where the silencers are placed.

All components are modelled with solid elements. In the following Sections more details about the modelling of each of the maniverter elements is given.

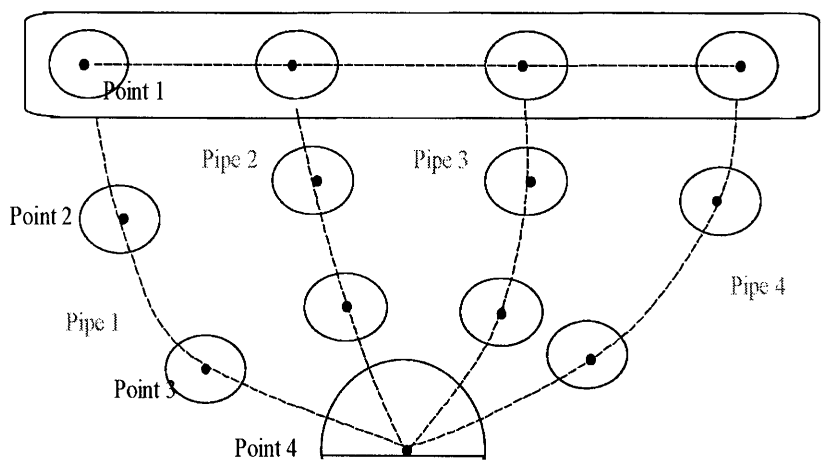


Fig. 65: Inlet Flange, Pipework and Plenum conceptual model

3.6.5. Parametric Model Details

Inlet Flange, Pipework and Plenum

The inlet flange is modelled a parallelepiped with four holes to allow the four round pipes to be inserted. Height, width and depth of the flange as well as the flange hole diameters are set as parameters associated to the coordinates of the ports and diameters of the pipes.

As mentioned, only the 4>1 topology is assumed. This implies that the four pipes join together at the same location.

The piperuns are characterized by three elements: the centreline, the diameter and the thickness.

The centreline is defined as a cubic spline with four control points. The 3 spatial coordinates *(x,y,z)* of each control point are set as parameters. The last points are imposed to be coincident.

Diameter and thickness are set as parameters separately for each of the pipes for maximum flexibility.

Collision between pipes is governed in the optimization phase by imposing the clearances between pipes have to be a value greater than a positive value (3 mm is used a value). This also ensures the manufacturing feasibility of the dome, which is thought as a stamped component.

Pipes are modelled using the "cable" feature of Unigraphics, using the "single segment" option. The cables are set to be perpendicular to the inlet flange.

The resulting pipework is illustrated in Fig. 65

The pipes join together in a collector. This is modelled as a spherical dome for simplicity reasons. Diameter of and thickness of the hemi-sphere are set as parameters. Its center is set coincident with the 4th point of the pipes centreline. Pipes are trimmed onto the sphere surface.

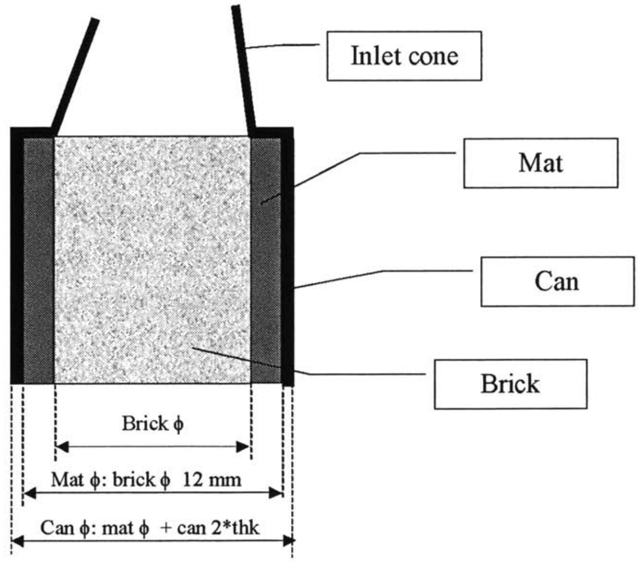


Fig. 66: Catalytic Converter Body Conceptual Model

Inlet Cone / Catalytic Converter

The catalytic converter used for this application features a ceramic monolith (also called brick or substrate), a cylinder with a honeycomb structure made by cordierite. The substrate is coated with precious metals (the catalyst itself), wrapped in a support mat and enclosed in a metallic can. A longitudinal cross section of the embodiment is shown in Fig. 66.

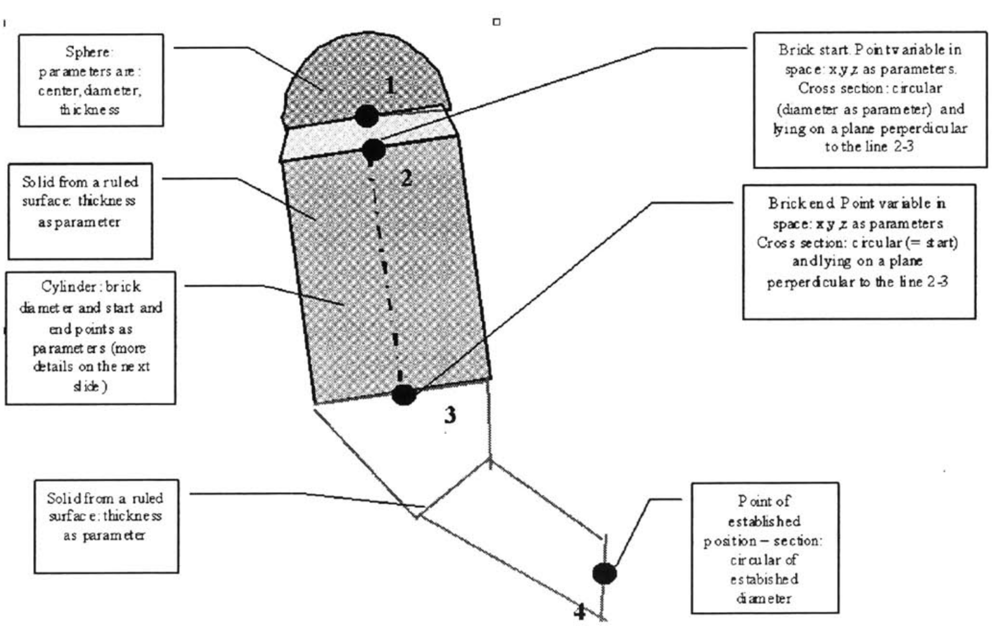


Fig. 67: Catalytic Converter & Inlet / Outlet cones conceptual model

From previous knowledge of the application, substrate dimensions suitable to convert the pollutants coming out of the selected engine are known. From the geometry standpoint, the converter can be assimilated as made by three coaxial cylinders: the substrate, surrounded by an annulus, which is the mat, and by a second annulus which is the converter can. General design rules, then, provide guidelines for the type of mat and the volume of the mat. The height and the different diameters are set as parameters but they are fixed during the optimization process.

The converter is connected to the plenum through an inlet cone (Fig. 67), which is built as a ruled surface. The point that defines the centre of the inlet circular cross section of the converter (point 2) as well as the angles of the converter axis with respect to reference planes are set as parameters so that the converter position and the orientation can be varied.

Outlet cone, outlet pipe and bracket

The maniverter ends with an outlet pipe, which is supposed to be connected to the exhaust system placed downstream. As in the majority of the front port engine applications, the pipe is assumed to run in a slot in the oil sump. Consequently its diameter and position are usually fixed. It is modelled as a cylinder with diameter, length, position and thickness set as parameters, but fixed with the application.

The outlet cone connects the converter to the outlet pipe and it is modelled as a ruled surface which runs through the circular cross section of the outlet pipe inlet section and the outlet circular cross section of the converter and it's tangent both to the converter and to the outlet pipe cylinders.

The outlet pipe is assumed to be connected to the oil sump via a bracket. In most applications a bracket is needed to take part of the maniverter load that otherwise would be imparted entirely on the connections between the pipes and the inlet flange causing stresses so high to generate a fatigue failure.

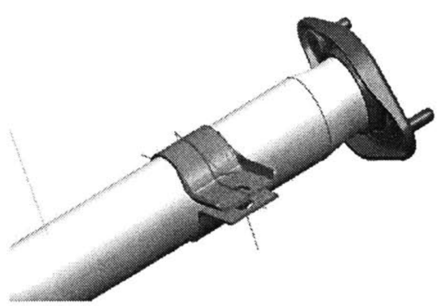


Fig. 68: Example of a Downpipe Bracket

Following some of the previous ArvinMeritor designs (see Fig. 68 for an example), the bracket is modelled as a clamp. The position of the clamp, width and connecting point are set as parameters; its diameter is associated to the diameter of the outlet pipe.

3.6.6. Modeling Outcome

The result of the parametric modelling effort is shown in Fig. 69.

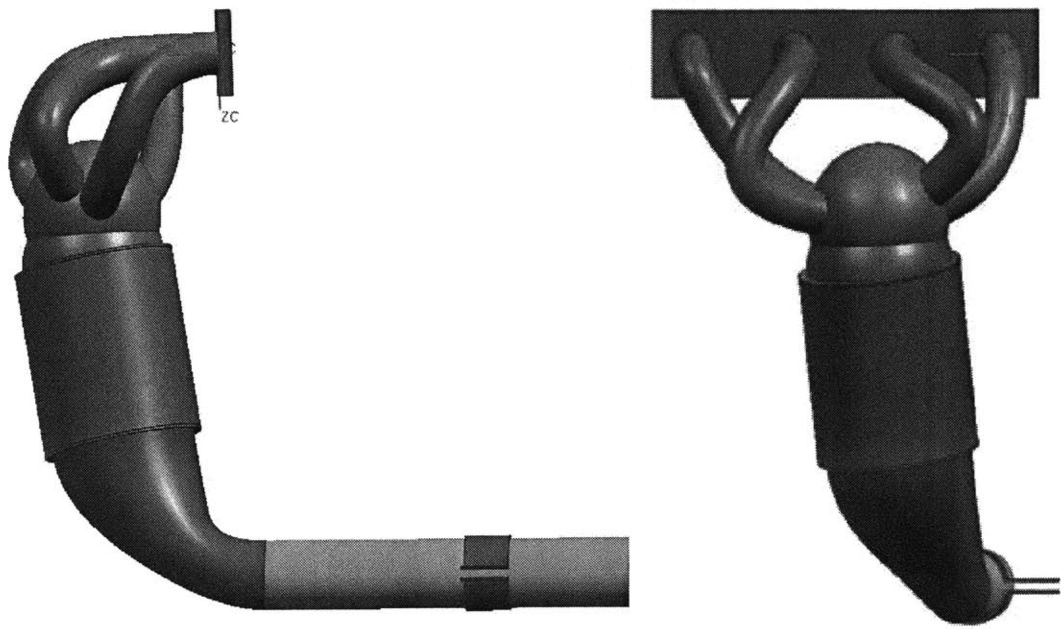


Fig. 69: The Maniverter Parametric CAD Model

Here are some overall figures.

The total number of parameters is 196, of which

- 118 are dependent
- 78 are independent. Of these:
  - 32 are fixed with the application
  - 46 can be varied with the application, i.e. during the optimization process

The complete list of the 78 independent parameters is given in Appendix 7.1.

Great attention has been given to the robustness of the model. Robustness is affected by the following issues:

- Parameters correlation. Even though the 78 parameters are considered as all independent, in reality they are loosely correlated. In fact not all the parameters value sets give rise to a feasible geometry. To exemplify, if the converter is placed at a higher vertical position than the sphere, there is a solid penetration of the converter cylinder and the sphere and the inlet cone vanishes. Similar errors are obtained when the curvature radii of the pipes are lower than the diameter of the pipe. Unigraphics translates these geometry inconsistencies into a geometry regeneration failure, returning an error or, sometimes, crashing.

- Software related issues. Even if when there is no evident geometric inconsistency, not infrequently during the geometry regeneration process, UG stops or crashes. The reasons are not completely evident; our best guess is that this might be due to internal bugs or to conflicts with the operating systems.
- Model quality for subsequent structural analysis. Tiny solid entities, small local radii of curvature, invisible gaps in the geometry are usually generated and handled within the CAD modeller. However, if and when exported in a different format and imported in a structural analysis software, they can generate major issues: the translation might fail or the finite element analysis might fail. These elements need to be non generated or, at least, eliminated from the model prior to export.

Most of the modelling time has been spent to mitigate these issues since, as we will see, the robustnessless of the geometry is a substantial limitation factor in the subsequent optimization phase. For the verification of the degree of robustness, in addition to the other methods, we used the possibility offered by UG to manipulate the parameters in an Excel spreadsheet and to update the geometry accordingly. Using the Excel random generator functions, the independent parameters have been varied within a specified range and a record of the geometry regeneration failures were made.

As far as parameters correlation is concerned, resolving completely the issue at the CAD model level would mean introducing several other non-trivial relationships and controls on the parametric solid elements, increasing model complexity a lot. Therefore we chose to accept this weakness in the CAD model, handling the consequences in the optimization process (see 4.3.1).

For the UG software related issues, a lot of trial and error loops have been performed. The same solid feature (i.e. a solid pipe) could be generated in several ways, i.e. using different commands. In our experience, the resulting feature is more or less prone to geometry regeneration issues depending on which route has been followed. Different modelling strategies were tried and then selected the "best".

The same trial and error process was followed to eliminate surfaces and solid glitches that would impair the structural analysis.

Despite the described limitations due to the assumptions and to the geometrical and software constraints, the result we arrived at is characterized by a great flexibility.

The simplified model allows mimicking very closely real applications: in Fig. 70 an actual manifold developed by ArvinMeritor is compared with a version obtained with the parametric model. As can be noted, the similarity is striking.

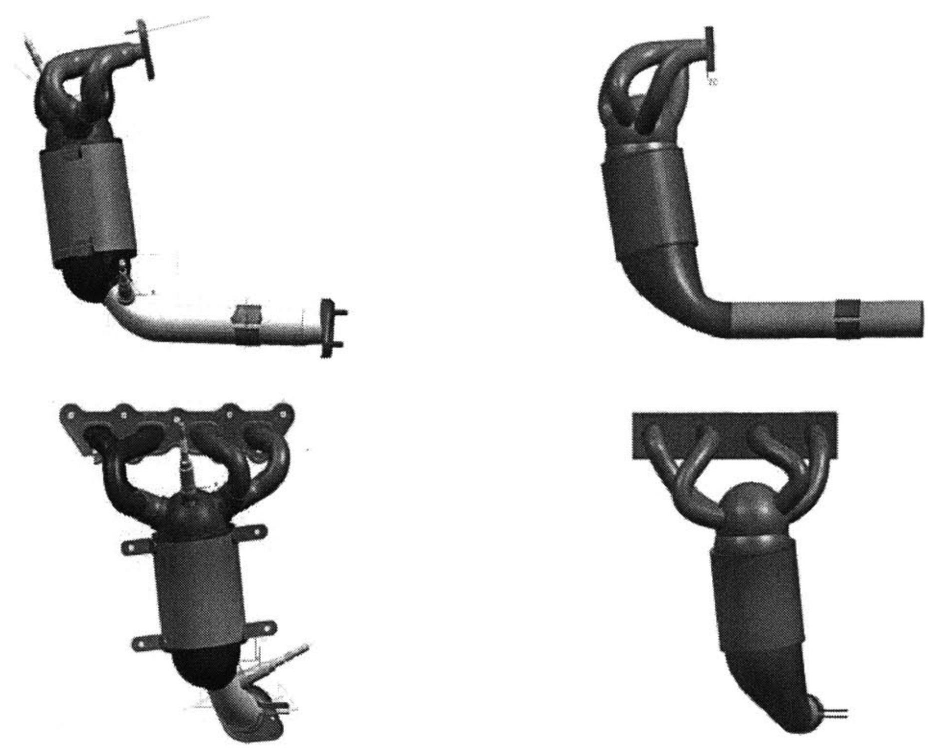


Fig. 70: Side and Front Views of a prototype actually developed for the Fiat Fire1.4 16V engine (left) and the corresponding version obtained with the parametric maniverter (right)

3.6.7. CAD Preparation for the Geometry Handler

The parametric / associative CAD model needs to be further enriched before it's suitable to be manipulated by the routine that allows the operations on the geometry to be executed in batch.

As we will see in the next Section, part of the operations is data extraction, which is realized using the advanced meta-language Knowledge Fusion (KF). KF operates on conceptual objects, which are not automatically generated when the model is created. Therefore an additional step is required, which translates the geometry features into logical objects that can be subsequently manipulated, Fig. 71. This step is performed interactively once.

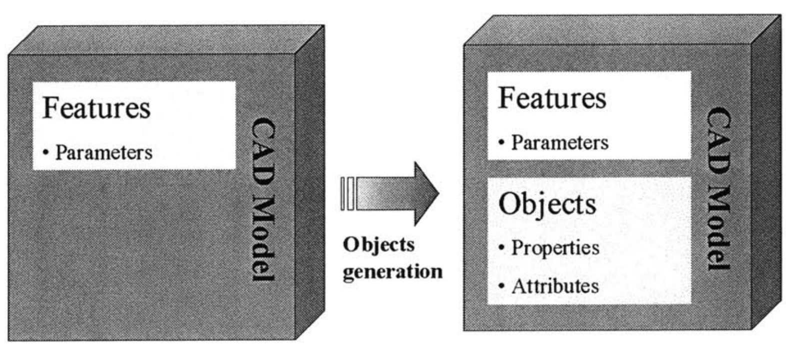


Fig. 71: CAD Model is enriched by KF Objects

During the translation process, objects are given some pre-defined properties, i.e. related information, such as the mass of a solid element or the length and the curvature of a curve. If a non standard piece of information is required (e.g. the radius of curvature of a curve), objects need to be manually edited and the relevant "attribute" added, calculated from existing properties.

3.6.8. Geometry Handler module

Once the model was created and prepared, the next step has been to create a code that handles the model in batch, i.e. without any user interaction.

The routine, named KEFAOptimizer, consists of more than 2000 lines of code and has been developed by Centro Ricerche Fiat (CRF) in C language. It relies upon the experience that CRF matured in a recent one-year long project where the automatic generation of vehicle subsystems models were studied (KEFA, in fact, stands for Knowledge Engineering for Fiat Auto). The architecture and algorithms are proprietary to CRF and therefore cannot be disclosed. In what follows, however, an overview of the functionality and inputs / outputs will be illustrated. In addition, in Appendix 7.2, the syntax of the routine command line and the system requirements are given.

The capabilities of KEFAOptimizer are the following:

- It allows the geometry model to be changed, given a set of parameter values
- It extracts from the model some relevant physical and geometrical properties, such as dimensions and masses
- It exports the native UG model (.prt) in a format (Parasolid) that can be read by FEA codes

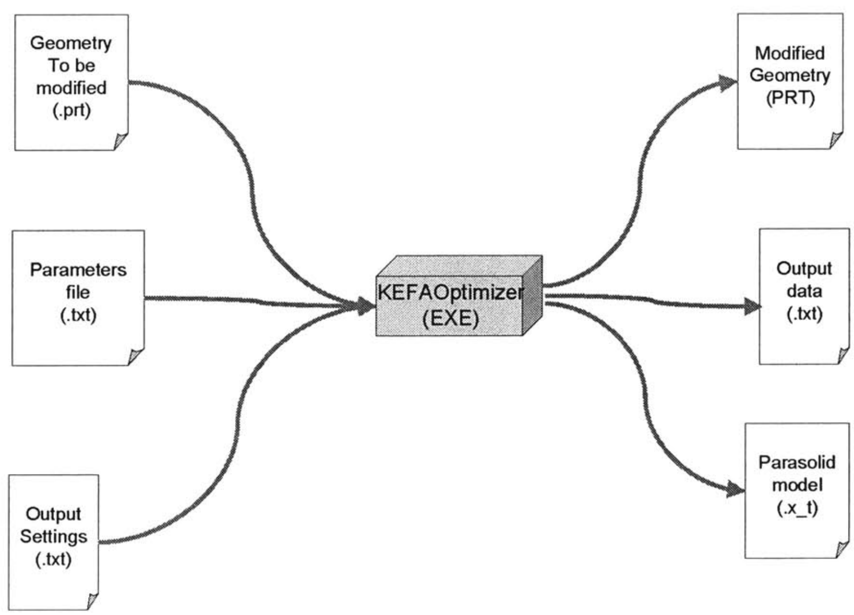


Fig. 72: KEFAOptimizer: Inputs and Outputs

The inputs to KEFAOptimizer are (Fig. 72):

1. The UG model to be changed
2. The list of parameters / expressions that KEFAOPtimizer will use to modify the geometry
3. The list of data to be extracted from the updated model

The outputs that KEFAOptimizer provides are:

1. The modified UG model
2. The data extracted from the updated CAD model
3. The Parasolid model

The process is executed in 50 seconds on the selected hardware platform.

Hereafter some details about the software architecture as well as inputs and outputs are given.

Software Architecture

KEFAOptimizer exploits both the native parametrical functionalities of UG managed by the open API and the ones available through its rule-based meta-language Knowledge Fusion (KF).

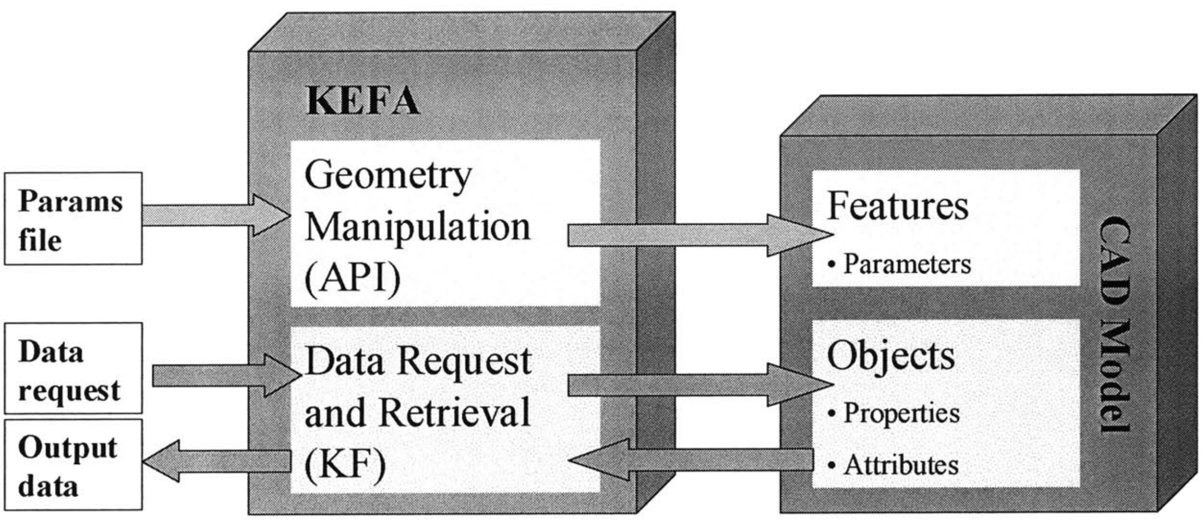


Fig. 73: KEFAOptimizer Architecture and its Interaction with the CAD Model

The KEFAOptimizer is made by two main blocks, which are executed in sequence:

- Geometry Manipulation block. This section reads the values of the parameters and, using the open API of UG, modifies the features accordingly

- Data Retrieval block. This section, which exploits the capabilities of KF, receives as input the information that are requested from the input file, identifies the objects' attributes and properties (such as the length of a pipe or its mass) and retrieves the values, storing them in the output data file.

UG Model to be modified

The CAD model to be modified must be a .prt model in UG NX2 format. As illustrated in 3.6.7, the CAD model, enriched with KF objects, must be edited and the KF "attributes" added.

Parameters / expressions used to modify the geometry

The parameters define the UG model geometry. Some of them are actually numerical values (absolute numbers of geometric features dimensions or coordinates), some others are defined by an algebraic expressions. The parameters list can be extracted from the model through the command *Tools* > *Expressions* > *List* executed within UG.

When created, parameters are given a standard name in the form of *_pxxx* where *xxx* is a 3 digit number; then they can be renamed by the user to allow unambiguous identification.

Here are some examples:

```
C4a_z=110          //z coordinate of the 4th control point of tube A
C4b_x= C4a_z       //x coordinate of the 4th control point of tube B
Dexta=Dinta+spa*2 //outer diameter tube A
```

(the // identifies the starting point of a comment).

Some of the parameters cannot be changed because the expression or the numerical values they are given are internal to UG and essential for the geometry consistency. If one attempts to modify them, UG replies with the following error message: "This expression cannot be modified because it is used by other feature". So, great care has been put in operating only on those parameters that are completely in control.

Data to be extracted

From the CAD model several data are extracted for subsequent use by the other CAE modules or by the optimizer. The obtained data fall into two main categories:

- Masses: of the different components (e.g. tubes, flange, converter, inlet/outlet cone, etc) and of the complete maniverter
- Geometrical dimensions: lengths, diameters, curvature radii and clearances (e.g.: tube and cone lengths and curvature radii, distance between tube A and tube B, etc).

Some examples of how they appear in the output settings file are given below with a short description.

`massTubeA.Mass=?->`	mass of tube A
`lengthPipeA=?->`	length of tube A
`curvPipeASeg2=?->`	first radius of curvature of the tube A
`distA_C[1]`	Distance between tube A and tube C

UG Modified model

UG .prt model modified with the values defined in the parameters list.

Output data

In the output data file, named forOptimizator.txt, all the values of the required data are written, in the same order in which they are specified in the output settings file.

As mentioned in the previous Sections, given a parameters values set, the geometry regeneration might fail for geometric or software reasons. To capture that, an execution completion code is written in the last line: if the error code is 0, no error has occurred and the generated geometry is valid; if the error code is 1, KEFA or UG, during its execution, has encountered an error and the resulting geometry is not valid.

A sample section of the forOptimizator.txt file is shown below:

```
massTubeA.Mass = 241.459734
lengthPipeA = 217.109355
lengthPipeASeg2 = 42.297333
lengthPipeASeg3 = 53.281194
lengthPipeASeg4 = 121.746355
lengthPipeATot = 217.324881
distA_C[1] = 87.657109
<<ERRORS>> = 0
```

Parasolid Model

The geometry handler routine exports the CAD geometry in a Parasolid format, useful for subsequent structural analysis. The Parasolid format is selected among those available because it can be generated quickly by UG and read easily by Patran (for more details, see 3.7.3).

Since for the structural analysis not all the features are required, only those needed are exported.

Main issues

The following issued challenged the development of the KEFAOptimizer:

- As for the CAD modelling, more than one API function exists which accomplishes a certain task. Trial and error helped in identifying the more "robust" functions.
- The API set for features manipulation is rather complete. The KF application, which should provide, at least, the same functionalities, being a more recent application does not map 100% to the API. Appropriate workaround were found where necessary.
- Both the API and the KF functions often stop the execution with a fatal or irreversible error when they fail to perform their task. To allow the automated design loops, KEFAOptimizer is particularly sophisticated in error handling so that the application terminates regularly in any case, just signalling that an error occurred.

3.7. Structural Analysis Module

3.7.1. Introduction

As anticipated in Section 3.4, to keep the complexity of the platform at an adequate level and given the computing power limitations of the selected hardware platform, the only structural analysis that was decided to be performed is the calculation of the first resonance frequency.

If excited at the resonance frequency, the system will exhibit very large displacements (for low damping levels), which are almost likely to degenerate in a fatigue failure. Therefore the resonance frequency is an important performance attribute of a maniverter.

For frequency calculation, the system is considered in hot conditions, fixed at the inlet flange (which is connected to the cylinder head) and at the bracket (which is connected to the oil sump). The physical understanding of the problem was instrumental to create an efficient and yet sufficiently refined model.

For the analysis, a Finite Element model is built using the software programs MSC.Patran and MSC.Nastran by MSC.Software Corporation.

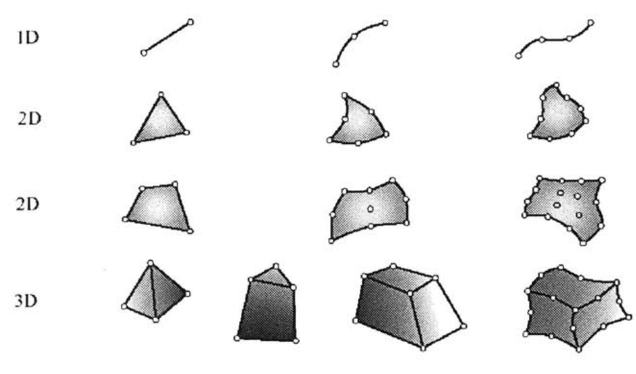


Fig. 74: Finite Element Types

3.7.2. The Finite Element Method and Patran / Nastran

The basic concept behind the finite element method (FEM) numerical technique is that a body or structure may be divided into smaller elements of finite dimensions called as "Finite Elements"

(Fig. 74). The original body or structure is then considered as an assemblage of these elements connected at a finite number of joints called as “Nodes” or “Nodal Points”. The properties of the elements are formulated and combined to obtain the properties of the entire body.

The equations of equilibrium for the entire structure or body are then obtained by combining the equilibrium equation of each element such that the continuity is ensured at each node. The necessary boundary conditions are then imposed and the equations of equilibrium are solved to obtain the required variables such as Stress, Strain, Temperature Distribution or Velocity Flow, depending on the application.

FEA was first developed in the late forties for use in structural analysis and it is used to analyze objects and systems that are of such a complexity that the problem cannot be solved in closed-form.[3] By the early 70's, FEA was limited to expensive mainframe computers generally owned by the aeronautics, automotive, defense, and nuclear industries. Given the phenomenal increase in computing power and the development of incredibly efficient algorithms, FEA packages nowadays run happily on PCs.

MSC.Nastran is a general FEA program capable of solving engineering analysis problems in the following areas:

- Linear and Nonlinear Static Stress Analysis
- Buckling Analysis
- Dynamic Transient Stress Analysis
- Steady and Unsteady Heat Transfer
- Optimisation Analysis

MSC.Patran is a pre-processor and post-processor for FEA.

Any Finite Element Analysis involves a pre-processing phase, a solution or processing phase, and a post processing phase, Fig. 75.

[3] From Wilkipedia, http://en.wikipedia.org/wiki/Finite_element_analysis

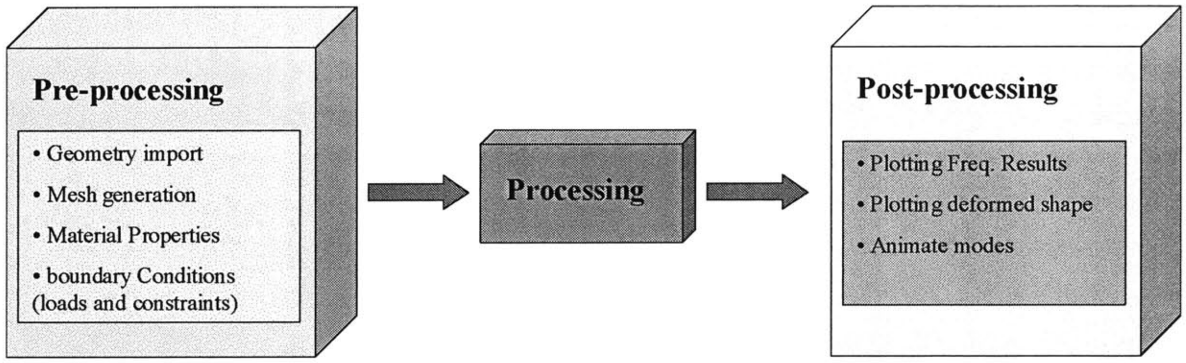


Fig. 75: The Different Phases of a FEA

In the pre-processing phase, a Finite Element model is built starting, if existing, from a CAD model (as in our case). Then a mesh is generated, material properties assigned to the solids and the constraints applied to the system. In the processing phase, the problem is solved and the results generated. In the post-processing phase, the results are gathered and analyzed. Pre and post-processing are performed with MSC.Patran; the problem solution is done with MSC.Nastran.

Each of the steps will be described in some detail in the Sections that follow. The description is made assuming that operations are performed interactively. In 3.7.3, we will see that the process can be automated with a minor effort.

3.7.3. Pre-processing: the FE model

CAD geometry import

As mentioned in previous Sections, even several decades after CAD and CAE tools were first introduced, software interoperability is still an issue to be considered with great attention when planning a multi-tool environment. Intellectual property protection pushed CAD software vendors to conceive model databases exclusively accessible; consequently CAE tools could not have directly access to CAD created model geometric databases. For years, the data transfer from one application to another has been occurring and still frequently occurs through open formats such as IGES (Initial Graphics Exchange Specifications), which, since its birth in 1979 is probably still the most popular format, or, STEP (STandard for Exchange of Product model data), officially known as ISO 10303. Data translation, however, always implies data loss or misinterpretation.

In these years we have seen a convergence between CAD and CAE tools. For example, most recently Ansys Inc. has introduced its Ansys Workbench environment where a Unigraphics .prt file can be read directly by the Ansys pre-processor and MSC.Patran 2004 has made several enhancements to improve its ability to work with leading CAD packages including CATIA, Unigraphics NX2.0.

This interoperability mode is still in its infancy and, in our experience, the import process of the CAD geometry by CAE tools is still a delicate phase full of pitfalls. It usually requires a lot of processing time, more often than not software crashes in the import phase and privileged affinity exists between CAD and CAE tools.

For our work, we chose a data translation format that both UG and MSC.Patran have demonstrated to have minor issues in working with: the Parasolid format. Recognized as one of the world's leading, production-proven core solid modeler, Parasolid is actually a geometric modeler supporting solid modeling, generalized cellular modeling and integrated freeform surface/sheet modeling. Developed by Unigraphics Solutions in Cambridge, England, Parasolid is used within Unigraphics Solutions' products and is licensed to many of the world's other leading CAD/CAM/CAE vendors. Designed for high-end CAD applications, Parasolid is now used in a wide diversity of leading mid-range systems. The global reach of Parasolid-powered applications spans multiple industries and has grown well beyond one million end users – all of whom benefit from the ability to seamlessly share geometric models through Parasolid's native *x_t* file format. Parasolid users also benefit from intrinsic, tolerant geometry processing that combines with Parasolid's translation and healing technologies to facilitate the interoperability.

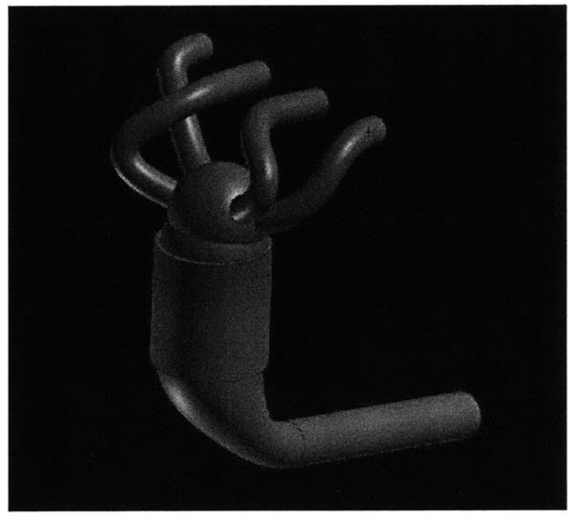


Fig. 76: Parasolid Geometric Model imported in MSC.Patran

As described in sub-section 3.6.8, at the end of the CAD geometry update, KEFAOptimizer exports in Parasolid format the eight solids that will have to be meshed: 4 pipes, the inlet cone, the converter can, the outlet cone and the outlet pipe. The inlet flange, the bracket, the support mat and the catalyst brick are not exported by UG because they are not used in the structural calculation process.

MSC.Patran, then, is set to import the Parasolid geometry. The work performed previously on model quality makes this process rather seamless. The average importing time is about 20 s on the selected hardware platform, Fig. 76. However, the choice to translate the data is not without drawbacks. The major issue we had to find a workaround for is related to the identification of geometric features.

Identification numbers (IDs) are unique integer numbers assigned by UG to geometric entities (e.g. to solid features) and are used by the kernel for its operations. Since these numbers are at the core of the CAD data structure and functioning, they cannot be changed by the user.

When the model is then exported, the Parasolid format retains the IDs and MSC.Patran, after importing the geometry, uses the same numbers to identify the different geometric entities. The IDs are generated each time the geometry is created; in general, therefore, they are changed from one instance of the geometry to another. This creates problems when applying material properties or boundary conditions. In the interactive operation mode it's the user who visually selects the solids or the surfaces of interest while the pre-processor picks up the associated ID. Any change of IDs is transparent to the user because he/she is able to visually locate the geometric entity of interest. If the same process, executed by program, uses the ID to identify the object of the action, it encounters a roadblock. For example, if fixed displacement are assigned to solid 3, where solid 3 is, in one case, the bracket, at the next geometry regeneration loop, if the ID 3 is assigned to one of the pipes, the boundary conditions are applied to a wrong element and consequently all the results are compromised.

Alternative ways of feature identification were therefore conceived. Detail is provided for each phase in the following Sections.

Mesh generation

The geometry in question in a Finite-Element analysis is represented by the collection of finite elements, known as a mesh. In the past, the meshing process was in great part a tedious manual time-consuming activity. In the last decade, Finite Element modellers have improved significantly their automatic meshing capabilities. Currently, meshing is essentially an automatic process.

In building the mesh, two main decisions have to be taken:

- The type of elements
- The size of elements (or elements density)

The choice of the type of element depends on the particular problem at hand and on the type of geometry. Since we decided to work with solid elements, two are the types of elements that could be chosen: Hexaedra, brick-like elements, or tetrahedra, or pyramid type elements. We chose to work with tetraedra because they can fit irregular boundaries and allow a change in elements size without excessive distortion. In addition, fully-automatic methods for generating triangular/tetrahedral meshes are well assessed. Tetrahedra are available with 4 or 10 nodes. 4-

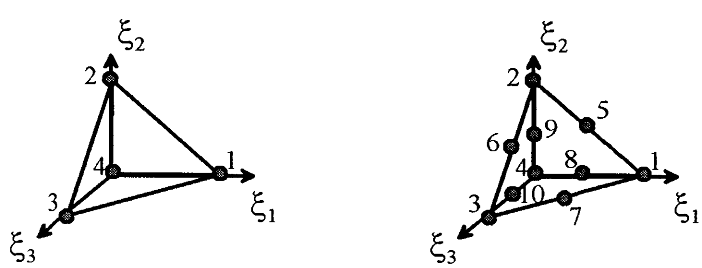
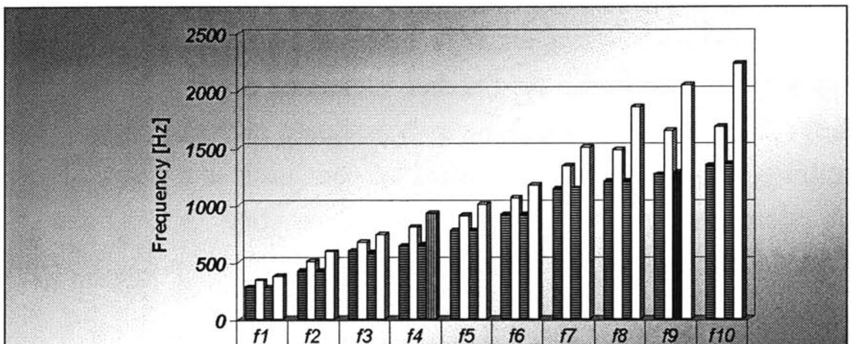


Fig. 77: 4-node (left) and 10-node (right) Tetrahedra

node elements, called in Patran Tet4, are coarser elements, with a lower accuracy relative to 10-node elements, the so called Tet10, Fig. 77.

Mesh size or element size refers to the dimensions of the tetrahedra with which the solids are discretized. Usually, regions of steep gradients in solution variables require a finer mesh. Current modelers feature adaptive meshing which automatically evaluates mesh discretization error in each element and determines if a particular mesh is fine enough. If it is not, the element is refined with finer meshes automatically. Adaptive meshing, however, is not used because, in case of resonance frequency calculation, the greater accuracy brought by adaptive meshing is not particularly significative on the first frequency. In addition, while adaptive meshing is easily accessible from the MSC.Patran GUI, it requires extensive programming if it has to be included in a batch routine.

Benchmarking is conducted to assess the accuracy loss in using Tet4 instead of Tet10 elements and in using a finer (4 mm) or a coarser (8 mm) mesh. Selected results are presented in Fig. 78 for the first ten resonances. As expected and confirmed by prior experience, Tet10 with 4 mm element size yield the most accurate result while Tet4 with 8 mm element size are the worst.



	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10
Tet-10 4 mm	283	424	593	641	772	915	1137	1201	1266	1344
Tet-4 4mm	339	510	677	809	908	1066	1341	1485	1644	1683
Tet-10 8 mm	280	419	584	651	780	923	1145	1208	1284	1358
Tet-4 8 mm	380	593	737	932	1002	1171	1507	1859	2048	2234

Fig. 78: Benchmarking of Tet-4 Vs. Tet-10 elements and 4 mm Vs. 8 mm element dimension

However, accuracy needs to be traded-off against execution time, Fig. 79. Even if Tet-4 8 mm elements give the worst results (frequencies are 30% higher), they were chosen because the solution time is incredibly shorter than with the others options. Absolute values are recognized to be affected by a significant error, but as far as trend is concerned (that allows to identify geometry configurations with higher and lower stiffness), the mesh is considered reliable enough.

	Tet-10 4 mm	Tet-4 4mm	Tet-10 8 mm	Tet-4 8 mm
Solution time [min]	253	42	74	3

Fig. 79: Solution time of problems with different mesh size and type of elements

The need to shorten the solution time drove also the decision not to mesh some parts of the maniverter but take them into account for frequency analysis purpose in different ways. These are: the inlet flange, the bracket, the support mat and the brick.

The inlet flange was not meshed at all because, being connected to the cylinder head, it doesn't influence the resonance frequency.

The bracket was not meshed too. The connection to the oil sump was modelled with rigid elements that connect the nodes in the bracket clamp area to the connection point.

The support mat and the brick are also not meshed explicitly. Resonance frequencies, in fact, are determined by the stiffness and by the mass of the system. While brick and mat contribute to the mass of the system (for 15-20%), they do not contribute significantly to its stiffness. To take this properly into account, on the inner surface of the converter can, a surface mesh is built of triangular elements (to match the tetrahedral elements of the solids). This particular mesh layer is assigned infinite flexibility but the equivalent mass of the brick and mat elements. A similar practice is in use at ArvinMeritor for this type of calculations.

The resulting mesh has approximately 25,000 elements (Fig. 80) and is built in approx 40 seconds on the selected hardware platform.

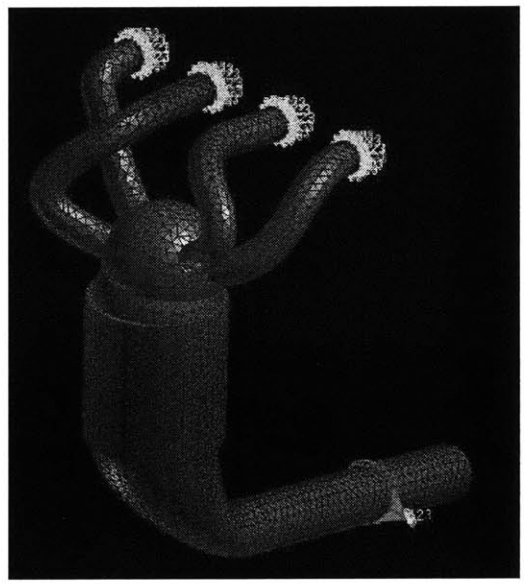


Fig. 80: Resulting Finite Element Mesh

Boundary Conditions

Boundary conditions are used to specify the loading or, as in our analysis, the constraints applied to a solid.

As mentioned, when mounted onto the engine, the maniverter is bolted though the inlet flange onto the cylinder head and it is connected through the bracket to the oil sump. These constraints were simulated in the following way.

Fixed displacements are applied to the inlet faces of the pipes, Fig. 81. As mentioned in previous Sections, the IDs of the faces could not be used to identify those geometric entities

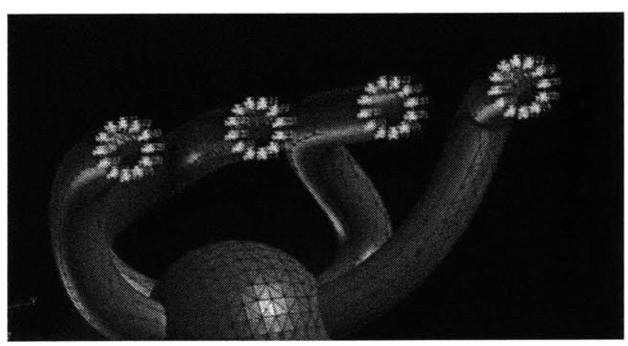


Fig. 81: Boundary Conditions applied to the Pipework

because they change with geometry generation. As an alternative method, faces were identified as lying on a plane of a fixed coordinate.

The bracket was, on the other hand, replaced by rigid connection to a fixed point. Connected nodes are identified, by having fixed coordinates, where the clamp holds the outlet pipe, Fig. 82.

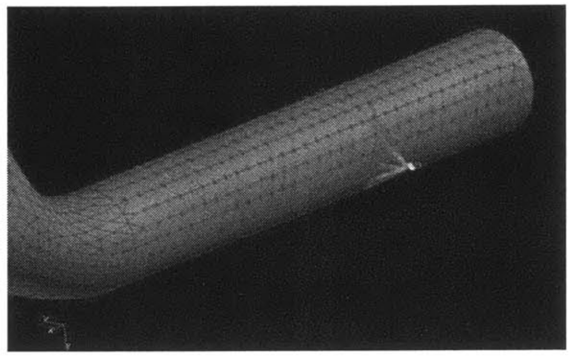


Fig. 82: Boundary Conditions applied to the Bracket

Material Properties

The maniverter is made by stainless steel. Several materials, both ferritic and austenitic, are used in common design practice, depending on the temperature working conditions, manufacturability and cost. In our application, for simplicity reasons, we will consider that all the components of the maniverter are made of the same material, which is known with the ANSI code AISI 304.

In the working temperature range (at about 800°C) we will also assume that the materials is isotropic, i.e. which has the same mechanical properties in all directions, and that it's linearly elastic, i.e. that it has a stress—strain response (valid only for small strains) as shown in Fig. 83. Such a material is then defined by the following quantities:

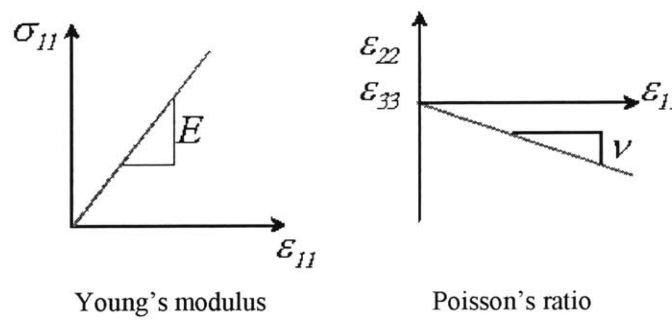


Fig. 83: Linear Elastic Isotropic Material Properties

- The Young's modulus E, which is set a 170,000 MPa
- The Poisson coefficient, ν, which is set to 0.3
- Its density, which is set to 7.8 kg/dm^3

3.7.4. Processing

For the structural analysis within the MDO framework, a modal analysis has been chosen. Modal analysis is used to find the natural frequencies of a structure. The frequencies are calculated in increasing order of frequency magnitude. Users can define number of frequencies desired for a range of frequency magnitudes. Two things are important: mode shape and frequency. The actual values of displacement are not physically meaningful, only the shape of the deformation is important.

The first natural frequency is usually the most important because it's the lowest, so only the first natural frequency is calculated. Design criteria state that this frequency should be as high as possible and definitely not below 250 Hz.

The MSC.Nastran solver is used for the calculation and the model is solved in about 2.5 min on the selected hardware platform.

3.7.5. Post-processing: gathering the results

Results are written by MSC.Nastran on a file with a default extension .f06. A sample portion is shown below.

```
E I G E N V A L U E   A N A L Y S I S   S U M M A R Y   (READ MODULE)


                         BLOCK SIZE USED ......................    2

                         NUMBER OF DECOMPOSITIONS .............    2

                         NUMBER OF ROOTS FOUND ................    1

                         NUMBER OF SOLVES REQUIRED ............    6

1    MSC.NASTRAN JOB CREATED ON 14-SEP-04 AT 11:48:39            DECEMBER  10, 2004  MSC.NASTRAN

                                     R E A L   E I G E N V A L U E S
  MODE    EXTRACTION      EIGENVALUE          RADIANS            CYCLES            GENERALIZED        GENERALIZED
   NO.      ORDER                                                                     MASS              STIFFNESS
     1        1          4.582813E+06      2.140750E+03       3.407110E+02       1.000000E+00       4.582813E+06
```

The value of the frequency (boxed) is then read by the optimizer (see 3.10.2): 340.7 Hz in this case.

If needed the modal shape can be interactively displayed superimposed on the undeformed geometry to provide a visual representation on how the structure will vibrate in resonance conditions.

MSC.Patran also provides the animation of the modal shape.

Fig. 84: Modal Shape Visualization (global lateral mode)

3.7.6. Running the Analysis in Batch

What was illustrated so far is essentially the process as it would be followed by a user in a series of manual operations. Patran records in a file, called "session file", all the instructions and user strokes. They could then be re-played automatically with a proper command.

Therefore, the entire process is executed interactively once and a session file "modal.ses" recorded. The batch program is then a .bat file with the command:

```
patran -sfp -b modal.ses -ans yes
```

where :

- -sfp (Session File Play) instructs Patran to play the session file
- -b: sets the execution in batch
- modal.ses: is the session file to be re-played
- -ans yes : causes Patran not to stop to get confirmation

The only issue that we had to cope with is, once again, geometry identification, in particular of the inner surface of the converter in order to apply the triangular mesh that carries the brick and mat mass. While interactively the user clicks on the right feature and Patran records in the session file its ID, in a fully automatic routine, this route is not viable because the ID will change. Therefore this part of the session file was replaced by a piece of code that was written in PCL (Patran Command Language). The routine identifies the surface as the one having a specified surface, quantity that does not change during the optimization process: the brick diameter, to which the converter can diameter is linked, in fact, depends only on the application, i.e. engine size.

3.8. Fluid Dynamics Module

The fluid dynamics characteristics of an exhaust system maniverter are of a paramount importance in determining the engine and the catalytic converter performances. For this reason a fluid dynamic module has been included in the ICE platform. As mentioned in previous Sections, a full 3D transient CFD (Computational Fluid Dynamic) calculation has been excluded due to hardware limitations but also because it is considered an overshoot. A 1-D transient simulation has been preferred instead. This has been used to predict the effects of the different maniverter geometries on the engine power and torque curve as well as the catalyst conversion capabilities. The commercial code used is AVL BOOST 4.0.4.

Similar to the geometry module, the fluid dynamic module is actually made by two parts: the model itself and the routine that manages the execution of the calculations in batch.

This Section, after some background on the fluid dynamic phenomena that occur in the manifold and the converter which were predicted with AVL BOOST, illustrates the main features of the simulation code, describes the model that has been set up and the results that were obtained.

3.8.1. Background

Backpressure

During the exhaust stroke, an engine may lose power through backpressure. The exhaust valve opens at the beginning of the exhaust stroke, and then the piston pushes the exhaust gases out of the cylinder. The higher the amount of resistance that the piston has to push against to force the exhaust gases out the more power is wasted. Power reduction comes also from an inefficient burn in the combustion chamber, where exhaust gasses are backed up and contaminate the next burn cycle. Backpressure is a result of the pressure losses in the manifold. Pressure losses are higher the smaller the cross sectional area, the more restrictions to exhaust flow are present and the more abrupt changes in the direction or in the cross sectional area exist.

Manifold Tuning

Any time there is a pressure change in an elastic medium (like air for instance) a series of resonances or vibrations will occur. Each time a power stroke occurs and an exhaust valve opens, a positive pressure occurs in the exhaust manifold. A negative pressure occurs in the exhaust manifold between the positive pressure pulses, especially at lower engine speeds. A positive, or high-pressure wave will propel gases in the same direction that it is travelling. A negative, or low-pressure wave will propel gases in the opposite direction that it is travelling. Pressure waves, or pulses caused by the exhaust port opening and closing will travel towards the open end of the tube the port is connected to.

These pressure waves respond in an interesting manner when they reach a sudden area change in the pipe. When a pressure wave reaches a larger cross sectional area, it will reverse its sign (positive becomes negative, and negative becomes positive) and its direction. For instance, when the exhaust port first opens, a strong positive wave will travel to the end of the pipe, change to a negative wave, and travel back to the exhaust port. This is called a reflection. Both the positive wave travelling towards the end of the pipe, and the negative wave travelling towards the exhaust port will propel exhaust gases towards the end of the exhaust system which is exactly where we want them to go. The amount of time that this cycle takes is dependant on the total distance that the wave has to travel, i.e. on the tube length. By changing the length of the manifold pipes, therefore, the cycle can be timed so that the negative return wave arrives at the exhaust port at the end of the exhaust cycle where it is most beneficial. Assuming that the negative return wave is timed correctly for a given engine at 3000 rpm, lengthening the runners will further delay the return wave so that it is timed appropriately for a lower rpm (e.g. 1000 rpm), and shortening them will time the return wave so that it is timed appropriately for a higher rpm (i.e. 6000 rpm).

The key to manifold length choice is simply timing the low-pressure return wave to give the greatest benefit for a given rpm. This process is called manifold tuning. Proper exhaust manifold/ tuning actually creates a vacuum, which helps to draw exhaust out of the cylinders and improve volumetric efficiency, resulting in an increase in horsepower.

Catalyst warm-up

When the Internal Combustion engine of a car is running, it burns fuel to produce power to drive the car. The burning of fuel also produces exhaust gases with pollutant substances. Three way Catalysts are usually installed in the exhaust system of gasoline cars to convert carbon monoxide, nitrous oxides and unburnt hydrocarbons into other gases that do not harm the environment.

However, catalysts are only effective when their temperature is above 250 to 300 degrees Celsius. This is known as the light-off temperature. When an engine is turned on, the catalyst gets heated up to its light-off temperature by the hot exhaust gases passing through it. Unfortunately, it takes some time for a catalyst to reach its light-off temperature. This time delay allows some undesirable exhaust gases to be released into the atmosphere.

Light-off time is longer if the thermal inertia of the exhaust system upstream the converter is higher.

3.8.2. The AVL BOOST Code

Overview

The thermodynamic cycle calculations in this work have been conducted using the AVL BOOST engine simulation code. BOOST calculates, in addition to the in-cylinder conditions, the unsteady 1-Dimensional gas flows in the intake and the exhaust systems. The above is achieved by solving the appropriate set of coupled non-linear differential equations using the Essentially Non Oscillatory (ENO) finite volume scheme [60] . The pipes are divided into cells and the flows of mass, momentum and energy from one cell to the other are calculated. The effects of wall friction, heat transfer and varying cross sections of the pipes are considered as source terms in the differential equations used. The gas properties at any location are determined by solving the conservation laws for unburned fuel, combustion products and air.

The flow losses at the pipe attachments are treated as quasi-steady. Catalogue values of pressure losses determined experimentally for the most common boundary elements, where appropriate, are used.

The unsteady calculations are initiated with user specified initial conditions in the system and they are continued until the solutions of subsequent cycles converge.

The calculation model is constructed by using a set of elements available in the BOOST pre-processor. The elements available comprise of:

- Pipes
- System boundaries
- Internal boundaries
- Cylinders, two or four stroke, with intake and exhaust ports controlled either by valves or piston
- Plenums, variable volume plenums (crankcases)
- Flow restrictions of various types e.g. check valve, rotary valve
- Junctions
- Air cleaners
- Fuel injectors
- Catalysts
- Air coolers
- Engine Control Unit (ECU)

In addition an arbitrary number of measuring points can be defined within any of the pipes where the flow data is monitored without influencing the flow.

The cylinder element models the working cylinder of an internal combustion engine. A conventional cranktrain motion as well as user-defined piston motion is also available in the program. For combustion modelling different models for the rate of heat release are available. In addition predictive combustion models are available for homogeneous SI, as well as DI Diesels. Heat transfers to the cylinder walls and to the port walls are taken into account.

Required Input Data

The input data that are required to build a BOOST mode data can be divided into: 1) the geometrical data (of the engine and of the intake/exhaust), 2) the flow data, 3) the data determined by the operating point and the charging device characteristics. Some detail is presented here below:

Geometrical data:

- bore, stroke and compression ratio of the cylinder

- valve sizes and valve lift curves
- length, diameter and bends of intake / exhaust pipes
- volume of the plenums
- firing order

Flow data:
- wall friction coefficient for turbulent pipe flow
- flow coefficients for the pipe attachments to the elements

Data related to the operating point:
- engine speed
- fuelling or air/fuel ratio
- combustion characteristics
- wall surface temperatures
- initial values for the manifold conditions
- pressure losses in the air cleaner, the catalyst and exhaust silencers

Output Data

The three types of calculations carried out by BOOST are single point, series e.g. full load performance, and animation calculations.

The program calculates pressure, temperature and velocity histories (i.e. function of the crankangle) of the flow in the pipes as well as pressure and temperature histories in all elements featuring a volume. In the cylinder element pressure traces, wall heat losses, ignition delay, combustion noise, gas exchange work, amount of residual gas, trapping efficiency in particular for two stroke engine are also calculated. The results of the last cycle calculated are then presented with respect to crank angle.

For the analysis of engine transients mean values for each element are available for each cycle calculated. The data obtained also include engine performance data e.g. volumetric efficiency, air/fuel ratio, residual gas content in the cylinder, mean effective pressure, power, fuel consumption, heat losses and the air flow.

For more detailed information on the AVL BOOST, the user is suggested to consult the related documentation [61] [62] [63] .

3.8.3. The BOOST model

A BOOST model is built of the entire intake, Fire 1.4 16V engine and exhaust system (featuring the maniverter), since the performance characteristics of the engine are known to be influenced by all elements that are enclosed in the boundary that goes from the intake inlet to the tailpipe. The intake and engine geometrical data as well as the fluid dynamic and combustion characteristics of the engine have been kindly supplied by Fiat-GM Powertrain in the form of a GT-Power model. The manifold section has been replaced with the maniverter designed and optimized in study.

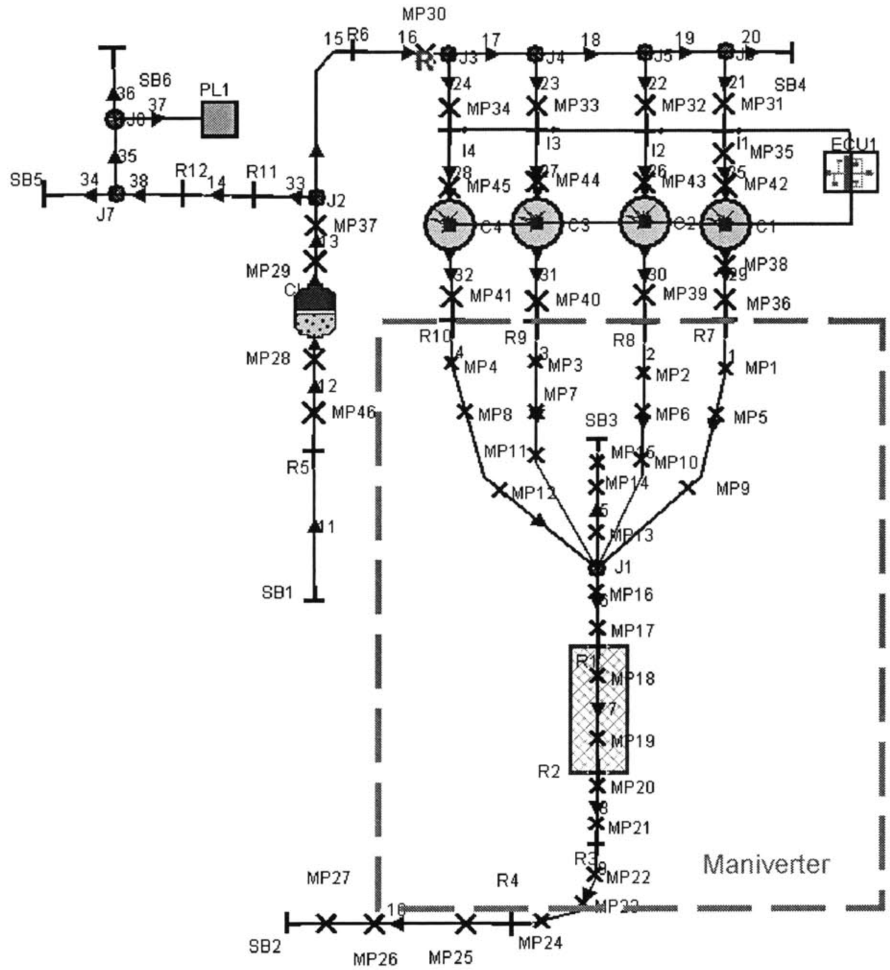
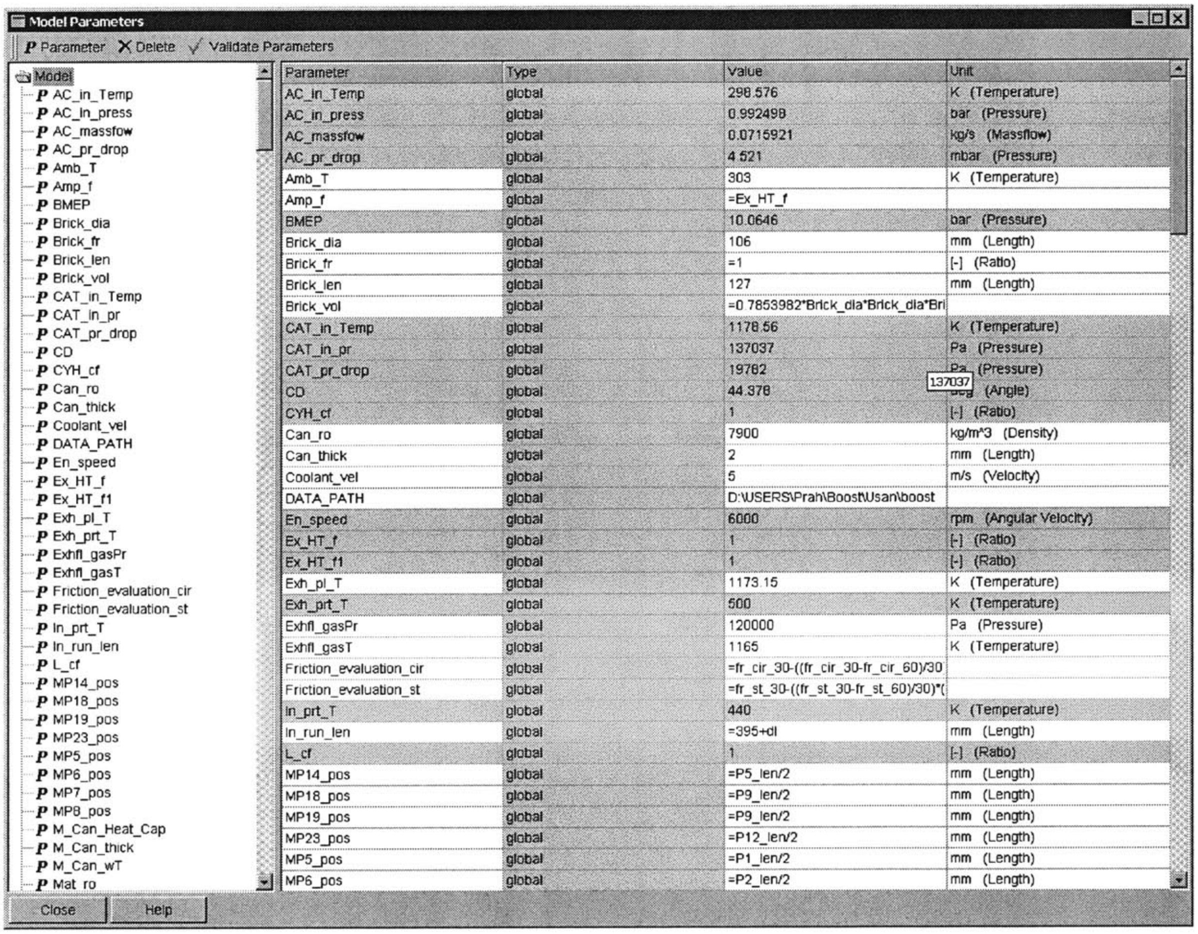


Fig. 85: The BOOST Model

The boost maniverter model, which is represented in Fig. 85, has been built parametrically. All dimensions and relevant heat transfer and fluid dynamic properties are set as parameters, for maximum flexibility. A sample of the parameters included in the model is shown in Fig. 86.



Parameter	Type	Value	Unit
AC_in_Temp	global	298.576	K (Temperature)
AC_in_press	global	0.992498	bar (Pressure)
AC_massfow	global	0.0715921	kg/s (Massflow)
AC_pr_drop	global	4.521	mbar (Pressure)
Amb_T	global	303	K (Temperature)
Amp_f	global	=Ex_HT_f	
BMEP	global	10.0646	bar (Pressure)
Brick_dia	global	106	mm (Length)
Brick_fr	global	=1	[-] (Ratio)
Brick_len	global	127	mm (Length)
Brick_vol	global	=0.7853982*Brick_dia*Brick_dia*Bri	
CAT_in_Temp	global	1178.56	K (Temperature)
CAT_in_pr	global	137037	Pa (Pressure)
CAT_pr_drop	global	19782	Pa (Pressure)
CD	global	44.378	deg (Angle)
CYH_cf	global	1	[-] (Ratio)
Can_ro	global	7900	kg/m^3 (Density)
Can_thick	global	2	mm (Length)
Coolant_vel	global	5	m/s (Velocity)
DATA_PATH	global	D:\USERS\Pran\Boost\Usan\boost	
En_speed	global	6000	rpm (Angular Velocity)
Ex_HT_f	global	1	[-] (Ratio)
Ex_HT_f1	global	1	[-] (Ratio)
Exh_pl_T	global	1173.15	K (Temperature)
Exh_prt_T	global	500	K (Temperature)
Exhfl_gasPr	global	120000	Pa (Pressure)
Exhfl_gasT	global	1165	K (Temperature)
Friction_evaluation_cir	global	=fr_cir_30-((fr_cir_30-fr_cir_60)/30	
Friction_evaluation_st	global	=fr_st_30-((fr_st_30-fr_st_60)/30)*(	
In_prt_T	global	440	K (Temperature)
In_run_len	global	=395+dl	mm (Length)
L_cf	global	1	[-] (Ratio)
MP14_pos	global	=P5_len/2	mm (Length)
MP18_pos	global	=P9_len/2	mm (Length)
MP19_pos	global	=P9_len/2	mm (Length)
MP23_pos	global	=P12_len/2	mm (Length)
MP5_pos	global	=P1_len/2	mm (Length)
MP6_pos	global	=P2_len/2	mm (Length)

Fig. 86: BOOST Model Parameters sample

The model has been validated against the current production system power and torque data, see Fig. 87 (blue line). For validation purposes, a current production maniverter has been modelled (black line). The comparison shows a non-perfect correlation. Almost likely this discrepancy is due to incorrect modelling of the junctions of different ducts, for which available data are not too precise.

However, the difference, which is less than 5%, in torque values is deemed not to impair the overall performance assessment.

The performance of the baseline maniverter used in the optimization process is shown in the same Fig. 87 (red line) and is very close the current production manifold's one.

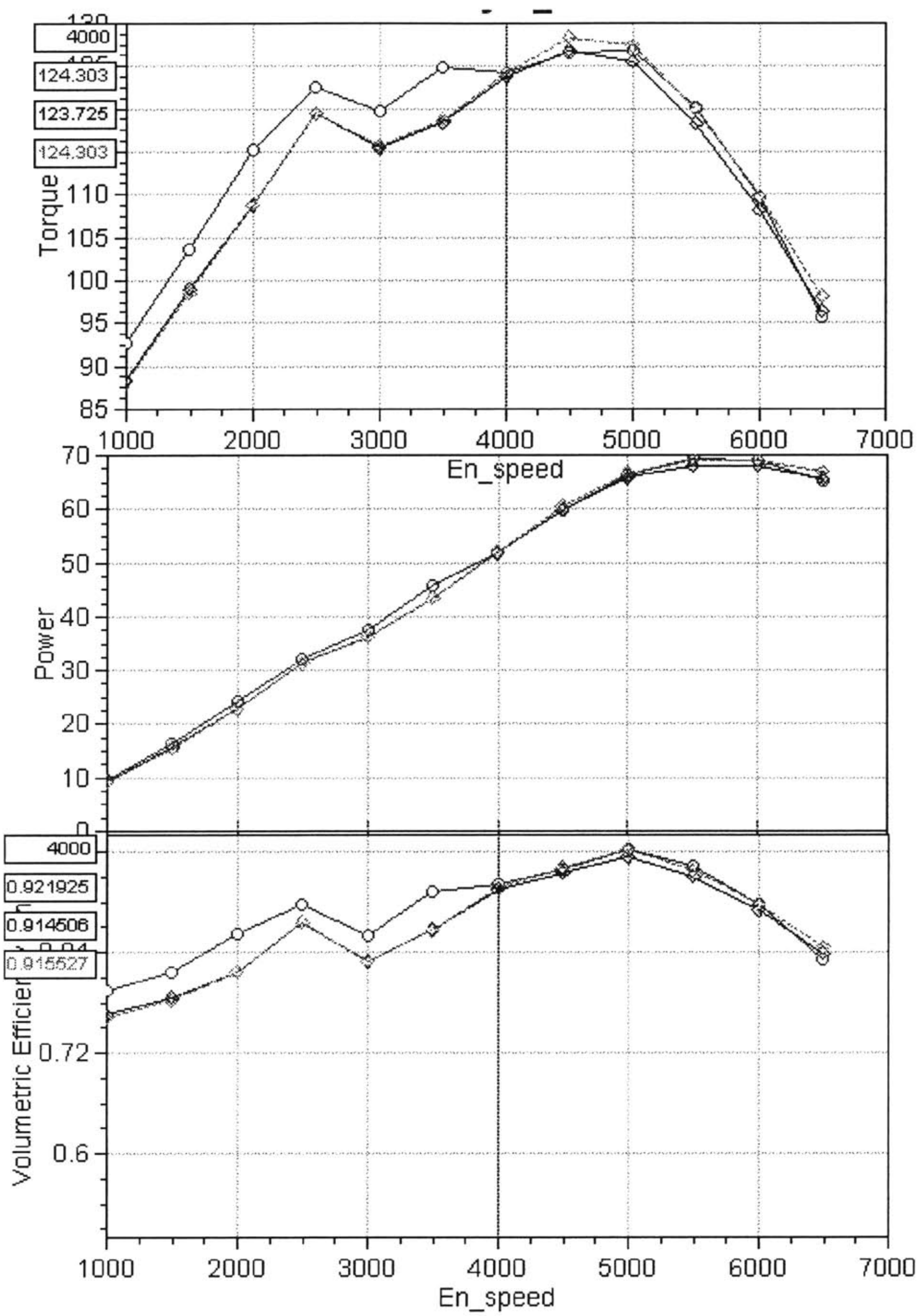


Fig. 87: Torque, Power and Volumetric Efficiency Data. Blue line: current production system GT-Power results, Black Line: current production system, BOOST results, Red Line: Baseline maniverter, BOOST results

The complete calculation time for six rpm numbers (1000 to 6000 rpm) takes about 15 min on the selected hardware platform.

3.8.4. Performance Attributes Definition

Many are the data that can be extracted from a BOOST simulation: power, torque, fuel consumption, massflow, etc. However, for optimization purposes, we synthesized a single value that could give an overall score on manifold performances.

In several discussions with Fiat-GM Powertrain engine experts, we captured that, for a conventional low to mid-size car (i.e. not a sporty one), which is the vehicle a Fire 1.4 16V engine is likely to equip, two are the important features that a maniverter has to contribute to:

- The highest torque, for best acceleration characteristics
- The most regular torque behaviour for good driveability. Particularly appreciated is the reaching of the maximum torque at the lowest possible rpm

We therefore selected the mean value of the torque across the rpm range as a metric of the first factor and the standard deviation (around the mean) as a metric for the second.

Then, we combined the two in a global performance index, which is defined as the ratio of the mean and the standard deviation of the torque:

$$Performance\ Index = \left(\frac{\mu_{torque}}{\sigma_{torque}} \right)^2$$

The ratio is raised to the second power to create an indicator, which is more sensitive to variations.

Performances are not the only driver for maniverter design. As discussed in previous Sections, another goal is to shorten the light-off time and thereby to reduce the pollutants that are emitted from the tailpipe before the catalyst starts converting. As a metric for this performance attribute a weighted average temperature at the catalyst inlet is selected, Fig. 88.

Engine Speed [rpm]	Torque [Nm]	Cat Inlet Temperature [K]	Temperature weights
1000	1.04E+02	1.12E+03	0.35
2000	1.27E+02	1.19E+03	0.25
3000	1.27E+02	1.25E+03	0.2
4000	1.25E+02	1.27E+03	0.1
5000	1.24E+02	1.30E+03	0.05
6000	1.09E+02	1.32E+03	0.05

Mean μ	119.3369083	-
Standard Deviation σ	10.14857403	-
Performance Index	**138.2736786**	**1196.5341**

Fig. 88: Performance Index and Average Catalyst Inlet Temperature

Higher weights are assigned to lower rpm, since those are the regime where the engine is more likely to revolve in the first 30 seconds after the engine start.

3.8.5. Automation of BOOST Calculations

So far the model has been described. The model has been built interactively. Then, it has been wrapped up in software layer to allow external applications to run BOOST calculations automatically. A new automation interface of BOOST has been developed, where the calling application can set input data of a BOOST model and also get results back. In what follows the outline of the interface will be given; the details cannot be disclosed since the know-how and the intellectual property reside within AVL.

Architecture

The interface is defined in a generic way. This means the external application can chose which data should be modified and also defines the required result. The automation interface is built on top of the existing python layer of BOOST[4].

All functions necessary for the automatic update of input data and results extraction are part of the new layer. This layer is also running the BOOST calculation itself, Fig. 89.

The automation interface performs three tasks:

- Supports the interface definition file
- Runs a BOOST iteration
- Gets back the result

The running of a BOOST iteration process, in its turn, is articulated in the following steps:

[4] Python is a portable, interpreted, object-oriented programming language. For a basic introduction, see http://www.python.org/doc/Introduction.html

- Loading of the BOOST model
- Reading the interface definition
- Changing the defined input parameter directly in the BOOST Model
- Running the BOOST calculation
- Generate the requested output data from BOOST results

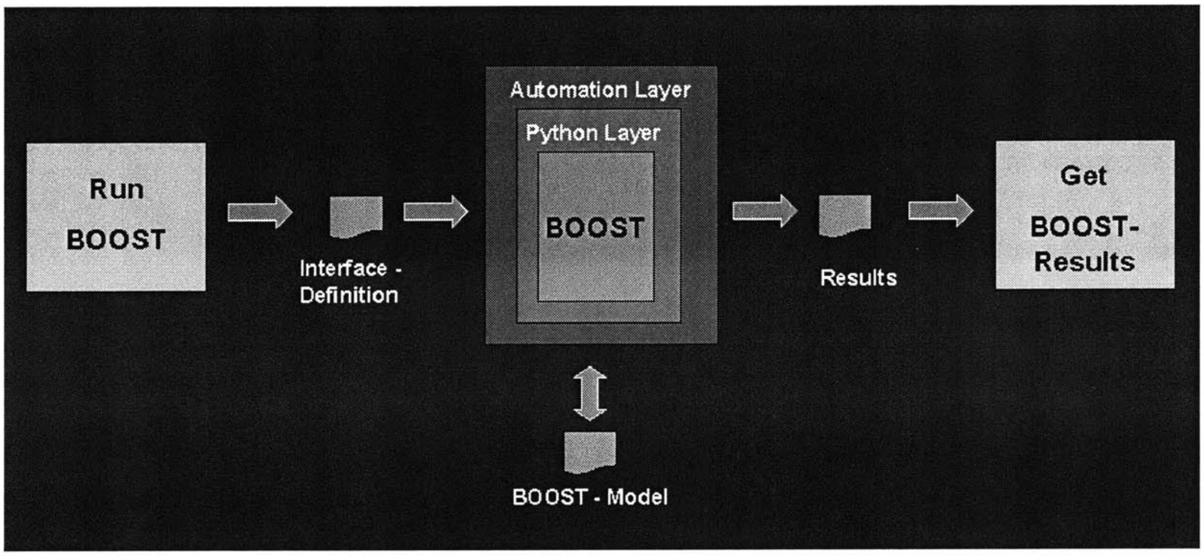


Fig. 89: Arcitecture of the BOOST Automation routine

Interface Definition File

The interface definition consists of 2 parts: the first defines the data that should be updated inside the BOOST-model, the second describes the requested results:

Example:

```
<boost_automatization_interface>
    <boost_input>
          .
          .
    </boost_input>
    <boost_output_request>
          .
          .
    </boost_output_request>
</boost_automatization_interface>
```

Details of the two sections are given in Appendix 7.3.

Output File

The automation routine writes the results in an ASCII file in three columns: the first contains the rpm values, the second the torque values and the last the catalyst inlet temperatures:

```
RPM             TORQUE            CAT_IN_TEMP
6.000000E+003   0.10868210E+003   0.12143396E+004
5.000000E+003   0.12686863E+003   0.11891622E+004
4.000000E+003   0.12496561E+003   0.11755363E+004
3.000000E+003   0.11572602E+003   0.10806941E+004
2.000000E+003   0.10889656E+003   0.10277020E+004
1.000000E+003   0.88027377E+002   0.92269092E+003
```

This file is read by the iSIGHT parser to retrieve the data (see 3.10.2).

3.9. Cost module

3.9.1. Introduction

It is rare in industry today that the cost of producing and maintaining a product is considered early in the design process. It is even more rare that the consumer's cost-of-ownership is considered. While much of the emphasis in Modelling & Simulation for design is on technology issues, integration of business issues is imperative to make a design which not only performs adequately, but also is cost-effective and guarantees adequate levels of profitability. For this reason, a cost model is included in the ICE platform.

It's common practice that the costing activity is done after the technical definition is worked out: detailed drawings are usually required and manufacturing engineers as well as key suppliers involved. Since the process can be time-consuming, quite often preliminary incomplete technical information is released. As the data are incomplete, some assumptions are made by costing engineers, which are hidden in the cost estimation. Then, prices are set by applying company's profit margins and the business case formulated. Later on in the project, when the design is technically complete and the cost updated, some of the assumptions prove wrong. What invariably happens is that a cost increase occurs, which erodes significantly the profits or, vice versa, drives a new design loop aimed at cost reduction.

We challenge that a detailed cost processing activity is needed every time and we believe that a much more efficient and yet sufficiently accurate approach can be adopted in the Concept and Preliminary Design so that it could be proficiently used to provide directions for most cost effective designs, Fig. 90 [64] .

The approach that we embrace and we applied in our MDO framework is what is called "Parametric Cost Estimating" (PCE). In the Systems Engineering Glossary, we can find the following definition for PCE: "A cost estimating methodology using statistical relationships between historical costs and other program variables such as system physical or performance characteristics, contractor output measures, or manpower loading" [65] .

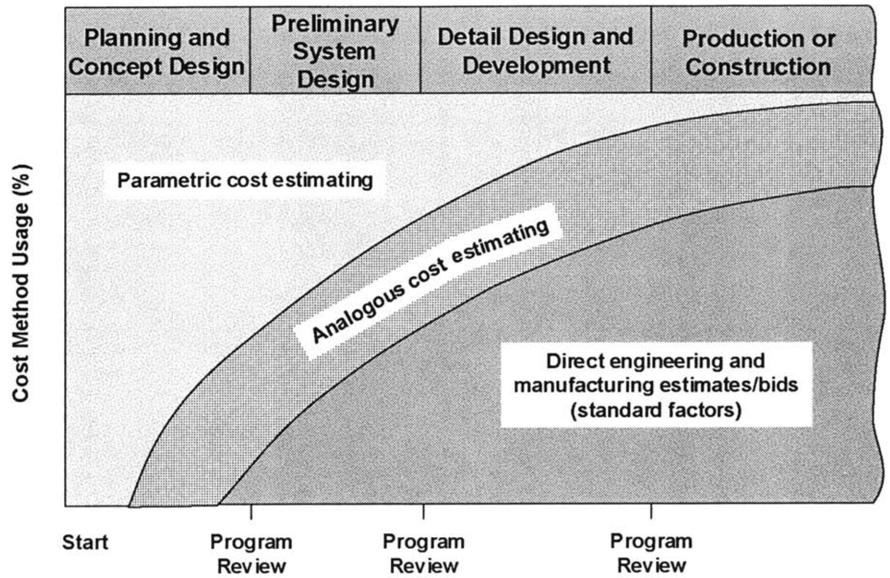


Fig. 90: Different Costing Approaches for Different Project Phases

The origins of parametric cost estimating date back to World War II. The war caused a demand for military aircraft in numbers and models that far exceeded anything the aircraft industry had manufactured before. While there had been some rudimentary work from time to time to develop parametric techniques for predicting cost, there was no widespread use of any cost estimating technique beyond a laborious buildup of labor-hours and materials. A type of statistical estimating had been suggested in 1936 by T. P. Wright in the Journal of Aeronautical Science. Wright provided equations which could be used to predict the cost of airplanes over long production runs.

In PCE costs are modelled based on past costs and "Cost Driver Parameters" are statistically/empirically fit. The assumption underlying PCE is that a clear linkage exists between cost and a product's cost drivers. PCE, therefore, search for product's / system's cost drivers and, based on past costs, tries to establish relationships between them. The accuracy on the overall cost is higher for systems made by several components. In fact, even if the individual component costs may be affected by a considerable error, when summed up, the errors partially cancel out. It can be mathematically proven that, given n components whose cost is affected by a variability σ, the variability of the total cost σ_T is considerably smaller:

$$\sigma_T = \frac{\sigma}{\sqrt{n}}$$

In case of our maniverter, parametric cost estimation is further eased by the fact that historically, 50-60% of the cost is actually material cost, which can be more accurately estimated.

In what follows, we will describe the cost model that we introduced in the ICE platform. Since costs are based on ArvinMeritor production and supply chain systems, actual numerical values are disguised and some sensitive details are not disclosed.

3.9.2. Maniverter Cost Structure

In a series of interviews with ArvinMeritor costing managers and engineers, the cost structure of a maniverter was uncovered and a simplified spreadsheet built (Fig. 91).

	Value	Units
Material Cost	12.86	[€]
Raw material	11.69	[€]
Inlet flange	2.57	[€]
Pipe A (AISI 304)	0.38	[€]
Pipe B (AISI 304)	0.42	[€]
Pipe C (AISI 304)	0.35	[€]
Pipe D (AISI 304)	0.41	[€]
Inlet cone	1.32	[€]
Converter can (AISI 304)	1.01	[€]
Mat	1.69	[€]
Outlet cone	1.57	[€]
Outlet pipe	1.38	[€]
Bracket	0.60	[€]
Freight	0.24	[€]
Dependent on Part Size		[€]
Dependent on Part Weight	0.24	[€]
Material Overhead	0.94	[€]

	Value	Units
Production Cost	11.18	[€]
Labor cost	6.20	[€]
Labor cost per unit time	28.39	[€/h]
Machine Cost	4.98	[€]
Actual machine cost	4.52	[€]
Bending Cost	2.37	[€]
Cost per bending machine hours	35.00	[€/h]
No. Of hours of bending operations	0.07	[h]
Average time per bend		[s]
Welding Cost	1.66	[€]
Welding Robot Cost	11.00	[€/h]
Total Welding Robot Time	0.15	[h]
No. Of Welds	14.00	-
Operations Time per weld	4.00	[s]
Welding length	974.16	[mm]
Welding time	487.08	[s]
Sawing, Deburring, Calibrating, Leak Testing Costs	0.50	[€]
Machine Overhead	0.45	[€]
Manufacturing Cost	24.04	[€]
Material Cost	12.86	[€]
Production Cost	11.18	[€]
Total Cost	**28.49**	[€]
Manufacturing Cost	24.04	[€]
Scrap	0.24	[€]
Overhead	2.24	[€]
Alloy cost	1.97	[€]

Fig. 91: Maniverter Cost Structure

The cost is made up by two components: material cost and production cost. Each of them and is then affected by an overhead which is usually a fixed factor of the cost component it is applied to.

Material Cost

Maniverter components, for costing purposes, are classified in the following categories:

- Tubes
- Metal sheets
- Stamped components
- Support mat

The brick is usually not considered in a cost submitted by exhaust system manufacturers, since this component is selected by the OEM and the price agreed between the OEM and the brick supplier.

For each of those categories, the main cost drivers were identified and, based on the extensive ArvinMeritor database, a statistical relationship worked out. For illustration purposes, we describe here in details, the tube cost only.

For tubes of a certain material type, the main material cost driver was identified as weight. Secondary effects were expected to come from: tube length, diameter, thickness, thickness/diameter ratio, and purchase quantity.

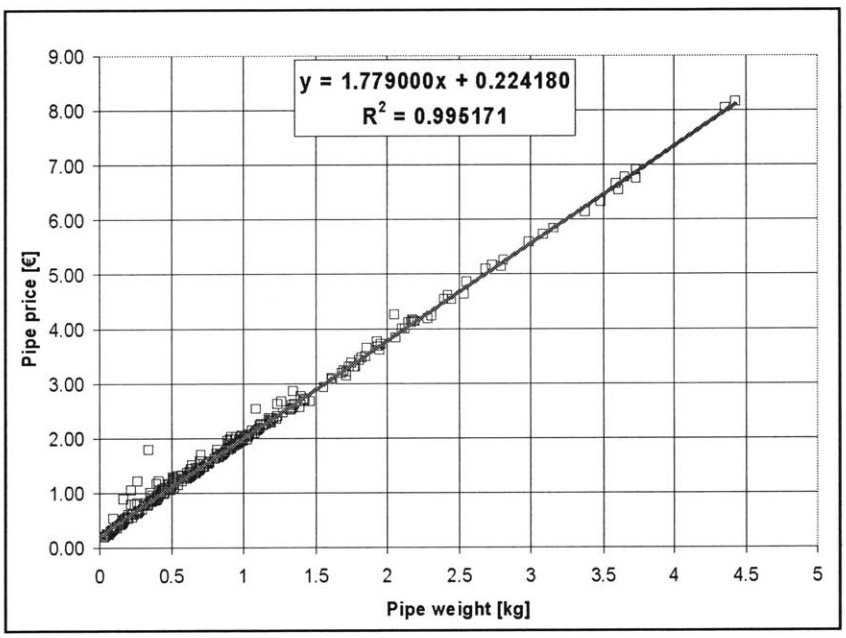


Fig. 92: Pipe Price as function of Pipe Weight (AISI 304)

About 300 orders of tubes (of the same material type) with different quantities, diameters, thickness and length were considered and the tube cost plotted against weight. The resulting graph is reported in Fig. 92. The linear correlation between material cost and weight is striking, with a correlation factor of over 0.99. The anticipated secondary effects are, therefore, minor and they are neglected.

Mass information of the different pipes in the maniverter (the four piperuns and the outlet pipe) is extracted from the CAD model and inputted in the cost model that yields back the cost.

Similar material-cost drivers relationships are worked out for the other material categories. Also for those, weight proved to be a prominent cost driver.

Production Cost

Production Cost consists of two components: the cost of machine operations and labour cost.

The main machine operations considered are: pipe bending and welding. Machine costs are essentially calculated as the product of the machine operations time by the time-unit cost of the machine.

Bending time is estimated to be mainly dependent on the diameter of the pipe to be bended, its thickness, the radius of curvature of the bend, the angle of the bend and the number of bends.

Welding time is considered fundamentally dependent on the welding length and welding speed. Additional time is then considered for welding robot positioning and disengaging.

Empirical or statistical relationships are elaborated that link these drivers to the machine time. Past quotations were used as database.

Times are considered at the end of an initial learning curve.

Machine unit operational costs (consumables, depreciation, etc) are known for any machine.

Labour cost per unit time is also known depending on the number of shifts and the number of operators needed to overview the machine operations. The number of shifts is determined by assuming an approximate production of 100,000 parts per year.

The same historical reference time period (2003) is considered for labour, machine and material cost to avoid the spurious effect of inflation.

Total Cost

On the sum of production and the material cost, a fixed percentage overhead is applied. Then, to come to the total cost, a fixed percentage of scrap is added together with the cost of the alloy. With “alloy cost”, the part of stainless steel material cost which refers to the alloy elements present in the steel (Ni, Mn, etc) is designated. The reason why this component is separately taken into account is as follows. Alloy cost fluctuates. In the past three years, its incredible rise has considerably increased the cost of the stainless steel and has forced exhaust system manufacturers to increase the costs of their products. Car manufacturers then required having explicit visibility on this component to track any unjustified cost increase claimed by their Tier 1 suppliers. Alloy costs, in fact, are published monthly and are a function of material weight and of the type of material.

In our cost model, the average of alloy cost per kilo over the year 2003 was considered.

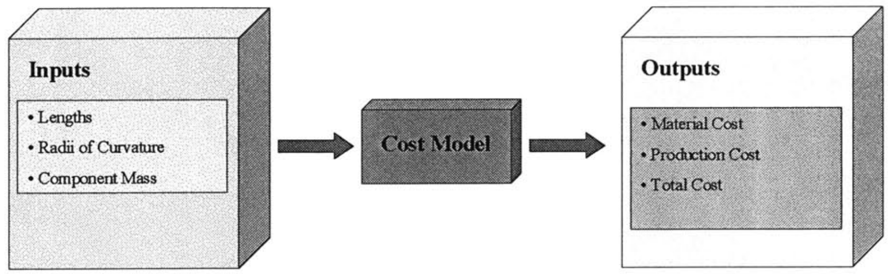


Fig. 93: Cost Model Overview

The resulting cost model is an Excel spreadsheet. As input it receives, from the CAD, maniverter components masses and dimensions data, and using the relationships between those data and the different cost components, gives back the maniverter cost, Fig. 93. The total processing time is less than 1 sec.

3.10. The Enhanced Development Framework

Up to now we've described the individual modules that compose the ICE platform. In this Section we are going to describe in detail how we assembled them together in the final platform and the MDO algorithms that can be executed. We will start the description from the "glue" that connects all the pieces together: the optimizer, which is iSIGHT, from Engineous.

3.10.1. The Optimizer

iSIGHT, from Engineous Software Inc., was developed to replace the manual trial and error processes with an automated, iterative procedure (i.e., a software robot). The software integrates all relevant design tools, such as CAD/CAM, in-house codes, and Microsoft Excel, and then automatically changes the input data, runs the analysis codes, accesses the output, and changes the input again based on pre-defined mathematical exploration schemes.

The software architecture (Fig. 94) includes a graphical user interface (GUI) to provide an easy-to-use methodology for defining, executing, and analyzing design studies. All entries made through the GUI are written to a text-based format in the Engineous proprietary Multi-Disciplinary Optimization Language (MDOL). MDOL is a user-friendly language that converts GUI definitions into iSIGHT's communication protocol. It is customizable and programmable to further streamline the execution of a frequently used design procedure. The inter-process communication layer provides the fundamental glue to seamlessly integrate a collection of simulation programs, numerical techniques, databases, monitoring and analysis tools, and both command-driven and GUI tools into a cooperative design automation system. Due to the nature of multidiscipline design studies, iSIGHT has incorporated both distributed and parallel execution facilities to leverage existing hardware capability in a network or server environment. iSIGHT also works with commercial queueing and load balancing software.

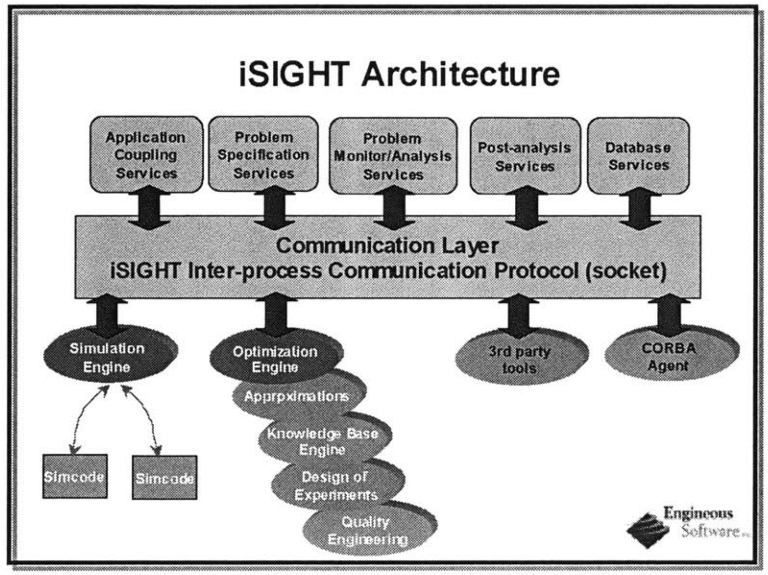


Fig. 94: iSIGHT Architecture

Bulding the ICE platform and running an analysis within iSIGHT is a 4-step process, Fig. 95 [66] :

- Process Integration: all the different analysis codes are inserted in a process flow, with their input and outputs.
- Problem definition: once the process is integrated in the iSIGHT framework, the user defines input and output bounds, initial values, and objectives for the design study. Additionally, the user defines a design study strategy. The design study strategy is dependent on the scope and type of problem being solved. It can range from a simple trade-off study, to a complex multi-disciplinary optimization formulation
- Design Automation: iSIGHT drives the different analysis codes to implement the chosen design strategy

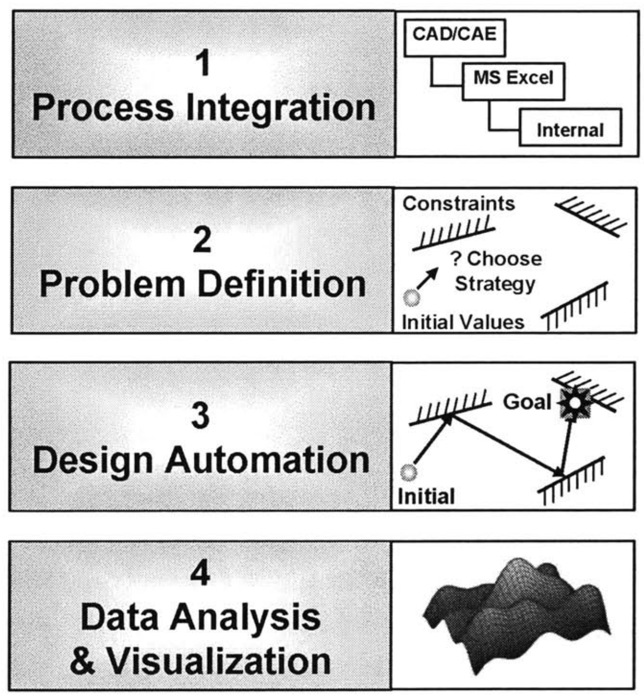


Fig. 95: The iSIGHT Four Steps

- In and post-processing data visualization: during the execution of the design studies, the user can monitor the design process as it progresses in real-time. By utilizing the graphs and tables created in iSIGHT at runtime, the user can identify trends and even make changes in the design definition or exploration plan. There is no need to wait until the end of the entire process and restart from scratch. Additionally, when the design process completes, the user has access to set of visualization, data and statistical analysis tools.

3.10.2. Process Integration: putting the pieces together

Process Integration uses a building block approach for defining model execution within iSIGHT. Users simply lay out their process on the interface with each block representing a step in the process. The fundamental building blocks of iSIGHT are the Task Block, the Calculation Block, the Simcode Block, and Custom blocks for commercial code support such as Microsoft Excel spreadsheets and MSC.Nastran models.

The building blocks are used to compose the individual modules of the ICE platform. In Fig. 96 the different modules inserted in the overall framework are shown. Given the dependency structure shown in Fig. 60, the different performance modules could be executed in any sequential order after the Geometry Module or even in a parallel order. Here the Structural-Cost-Fluid Dynamics sequence is chosen only because it reflects the order in which the different modules were built.

Fig. 97 presents a more detailed pictorial representation of the data flow and interaction of the different modules.

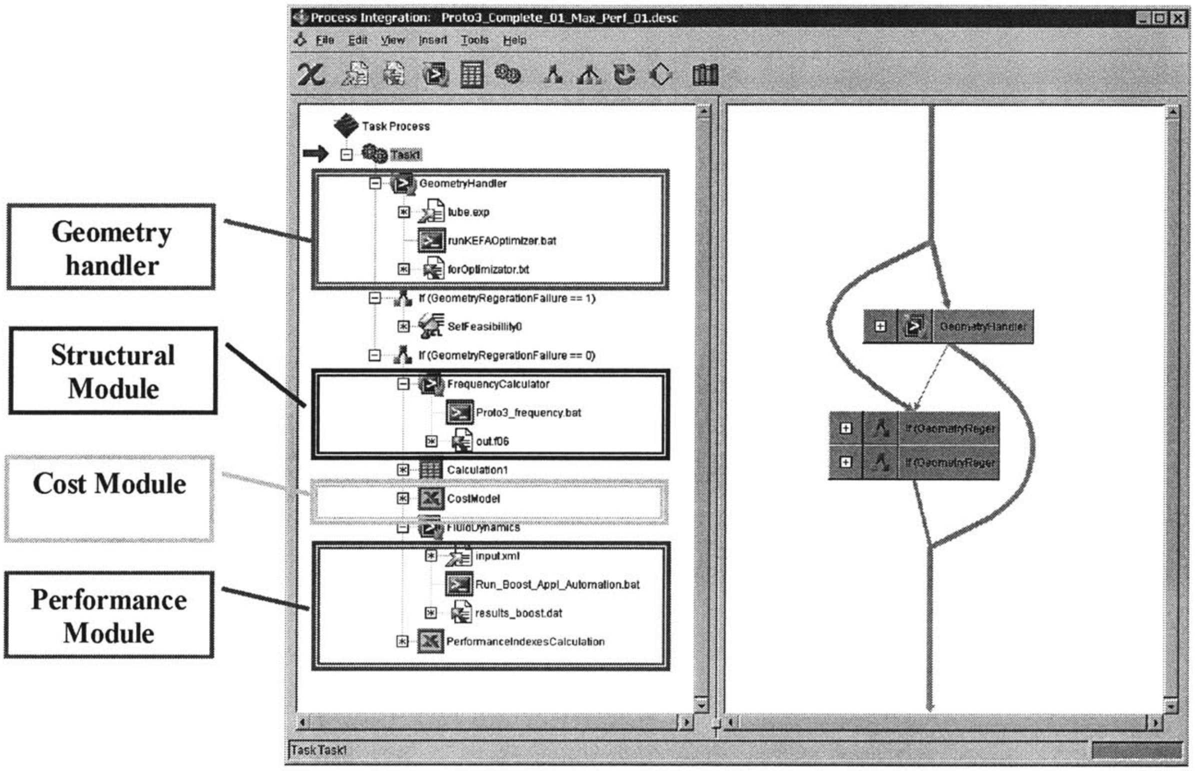


Fig. 96: The iSIGHT Implementation of the ICE Platform

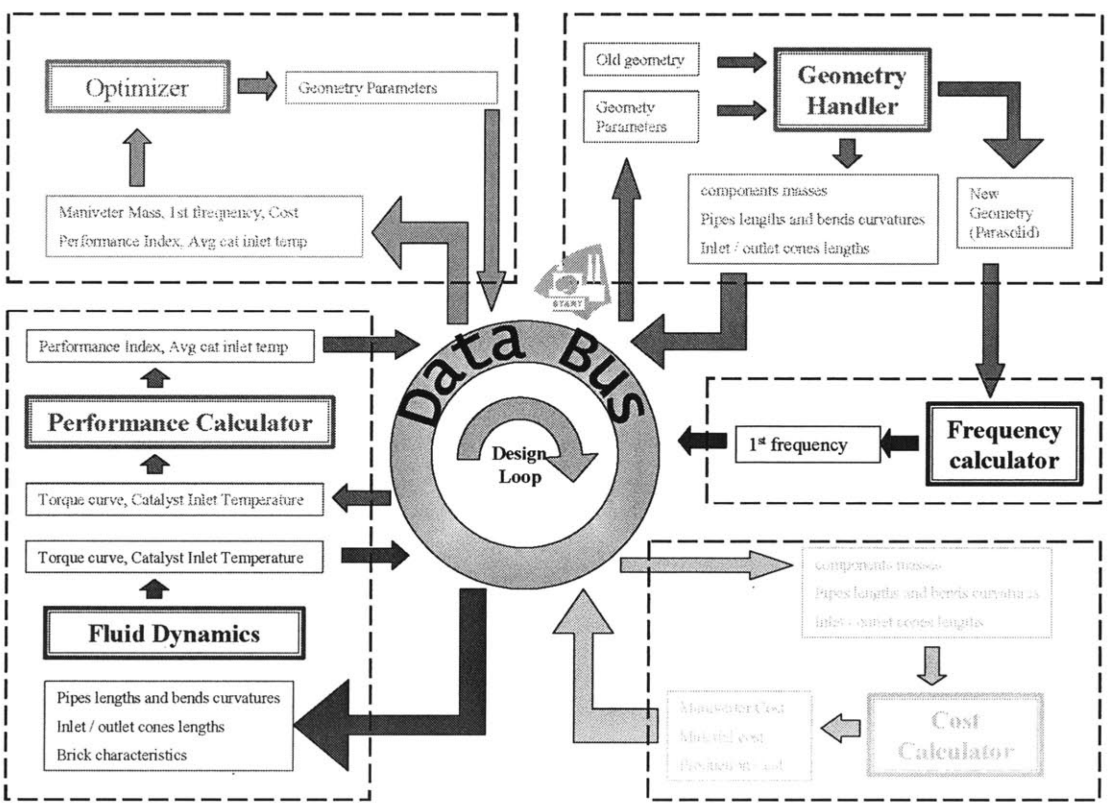


Fig. 97: ICE Platform Data Flow

The design iterations enabled by the ICE platform are articulated in the following steps.

1. The design loop starts with a baseline geometrical configuration defined by its parameters values (stored in the file tube.exp)
2. The Geometry handler module calculates the geometrical and physical dimensions that will be used in the following steps and writes the values in an ASCII File (forOptimizator.txt). In addition, it creates the Parasolid file needed by the subsequent structural analysis
3. The Frequency calculator module, reads the Parasolid file and feeds back the first natural frequency (written in the out.f06 file)
4. The Cost Calculator module receives by iSIGHT the necessary geometrical and physical data and feeds back the maniverter cost and its breakdown of material and production cost
5. The Fluid Dynamic Module receives geometrical data of the maniverter (input.xml) and feeds back the torque values and the catalyst inlet temperature over the 1000-6000 rpm range (results_boost.dat)
6. The Perfomance Calculator receives the torque values and the catalyst inlet temperature over the 1000-6000 rpm range and feeds back the Performance Index and the weighted average catalyst inlet temperature
7. The Optimizer receives from the different modules the maniverter performance attributes, i.e. mass, 1^{st} natural frequency, cost, performance index and catalyst inlet weighted average temperature and, depending on the development strategy, feeds back a new set of parameters values (stored again in the tube.exp file).
8. If the target of the performance attribute(s) is achieved or the maximum number of iteration, the design process stops, otherwise a new loop starts (from point 2.)

We note that a loose coupling exists between the different applications. Interchanges occur only in the format of ASCII files (with the exception of the Parasolid file) through a data bus provided by the iSIGHT architecture.

While this approach has some limitations, it gives two fundamental advantages: 1) interoperability is guaranteed because the communication media are ASCII files, 2) a module can be eliminated or upgraded or a new module can be added with a minor effort because the interfaces are limited and simple. The latter feature, in its turn, has the powerful consequence of fostering scalability, both horizontally and vertically:

- Horizontal scalability: new performance attributes can be evaluated when the related prediction models become available;

- Vertical scalability: a module can be generated to be rather simple in the beginning but it could be refined both in its capabilities and in its accuracy.

Each module is represented in iSIGHT with an input / simulation code / output structure. As an example, the Geometry Handler module is shown in Fig. 98.

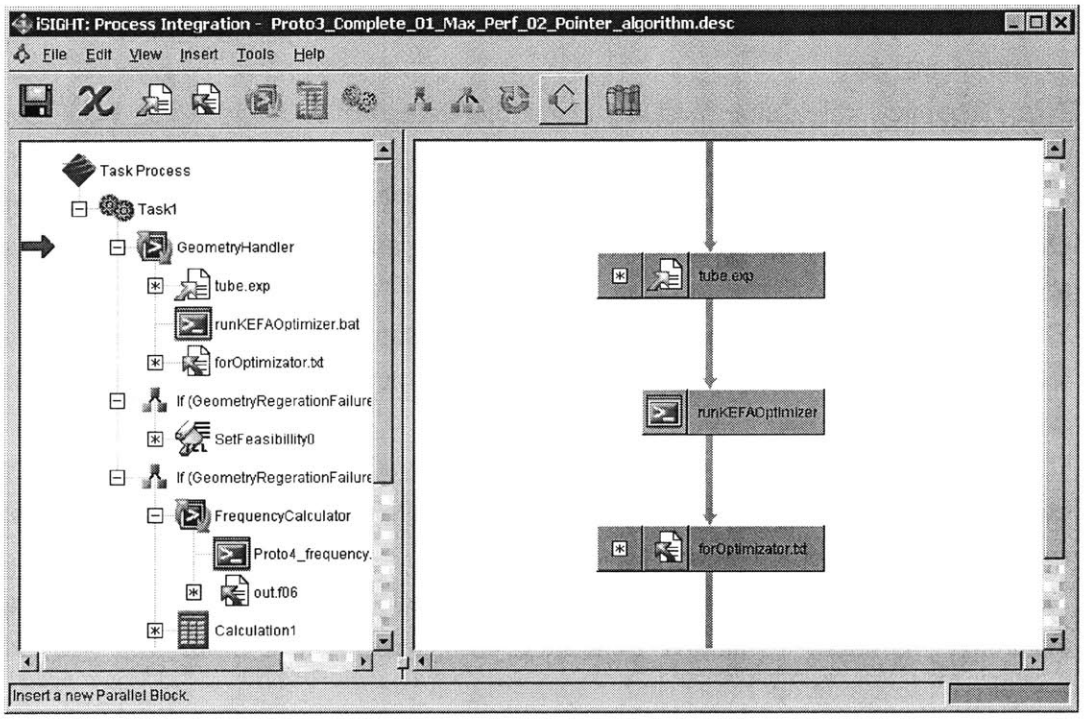


Fig. 98: The Input / Simulation Code / Output Module Structure

iSIGHT's File Parser is employed to write data in input and read data from output files required in a simulation process. The interface provides a set of buttons that allows users to graphically navigate through text-based files. Here, the users identify any values that will be changed during the design process, or any parameters that need to be monitored for acceptance criteria. Since the interface provides graphical actions and feedback, users are not required to write code to create the commands necessary to parse the data files.

The interface and an example of the parsed input data and output data files are reported in Fig. 99 and Fig. 100 respectively.

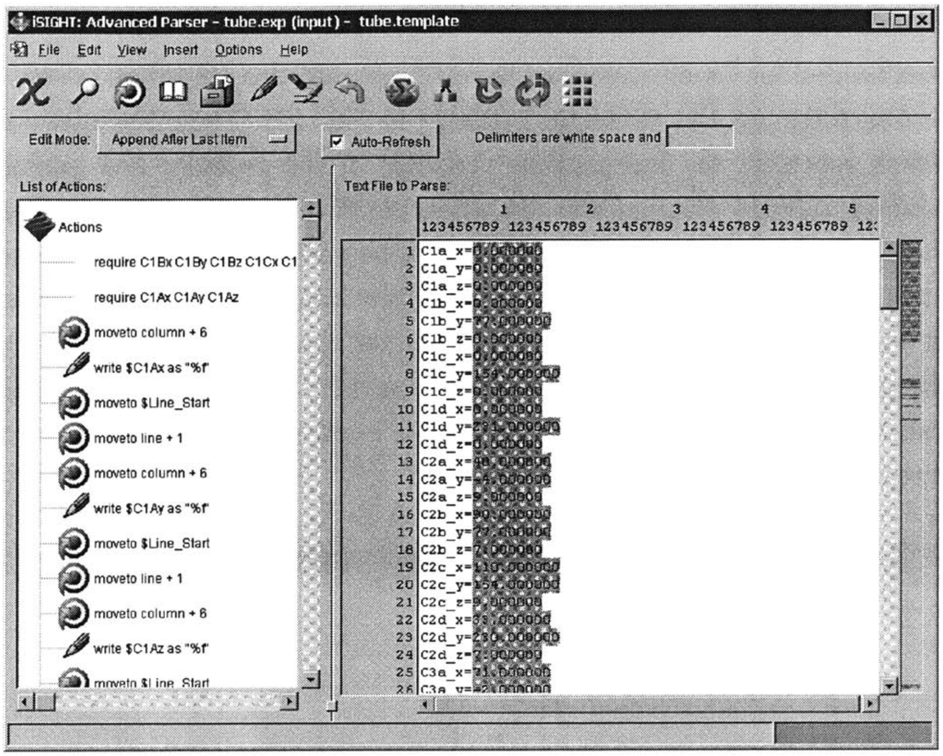


Fig. 99: Input File Parsing Sample

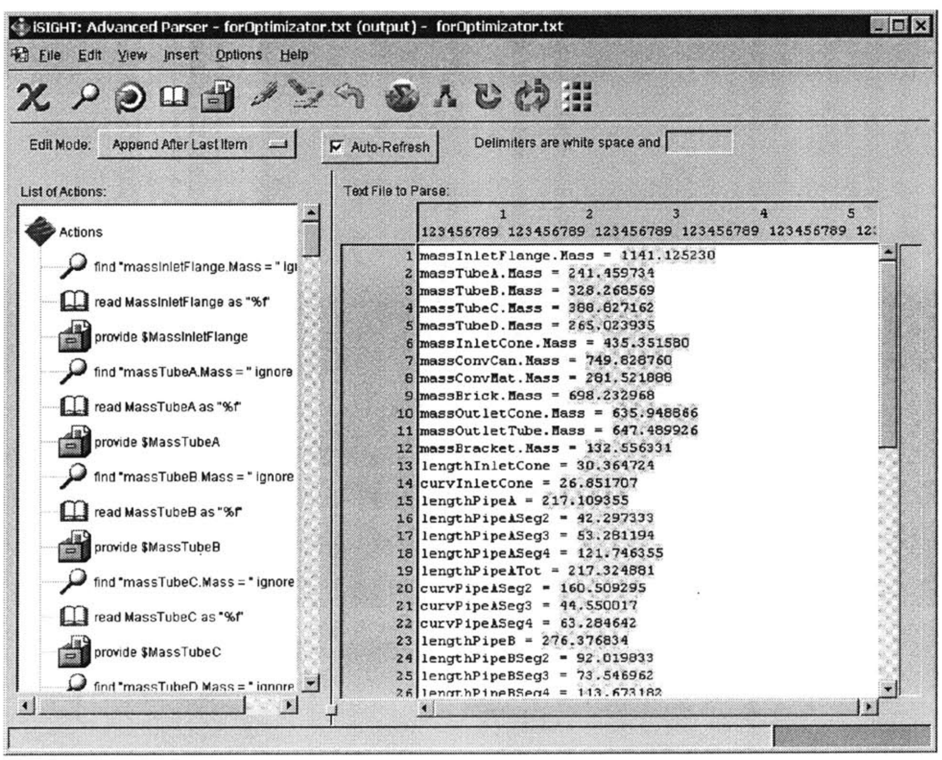


Fig. 100: Output File Parsing Sample

3.10.3. Problem definition

All quantities needed for the operations of the different modules are stored in iSIGHT. In the current application there are 144 parameters, shown in Fig. 101. They fall in two categories: input data, which represent the independent variables of the problem and output data, the dependent variables. iSIGHT distinguishes the ones from the others by color: blue for the input, black for the output. Any independent variable could be variable during the execution of the development strategy (the box in Var. column is ticked) or constant (the Var. box in unticked).

The output variables, on the other hand, can be either objective or not and the goal of the design strategy can be then to minimize or to maximize them. This is shown pictorially by the up and down arrows in the Obj. box. Fig. 102 shows an example of performances maximization and, at the same time, maniverter cost minimization.

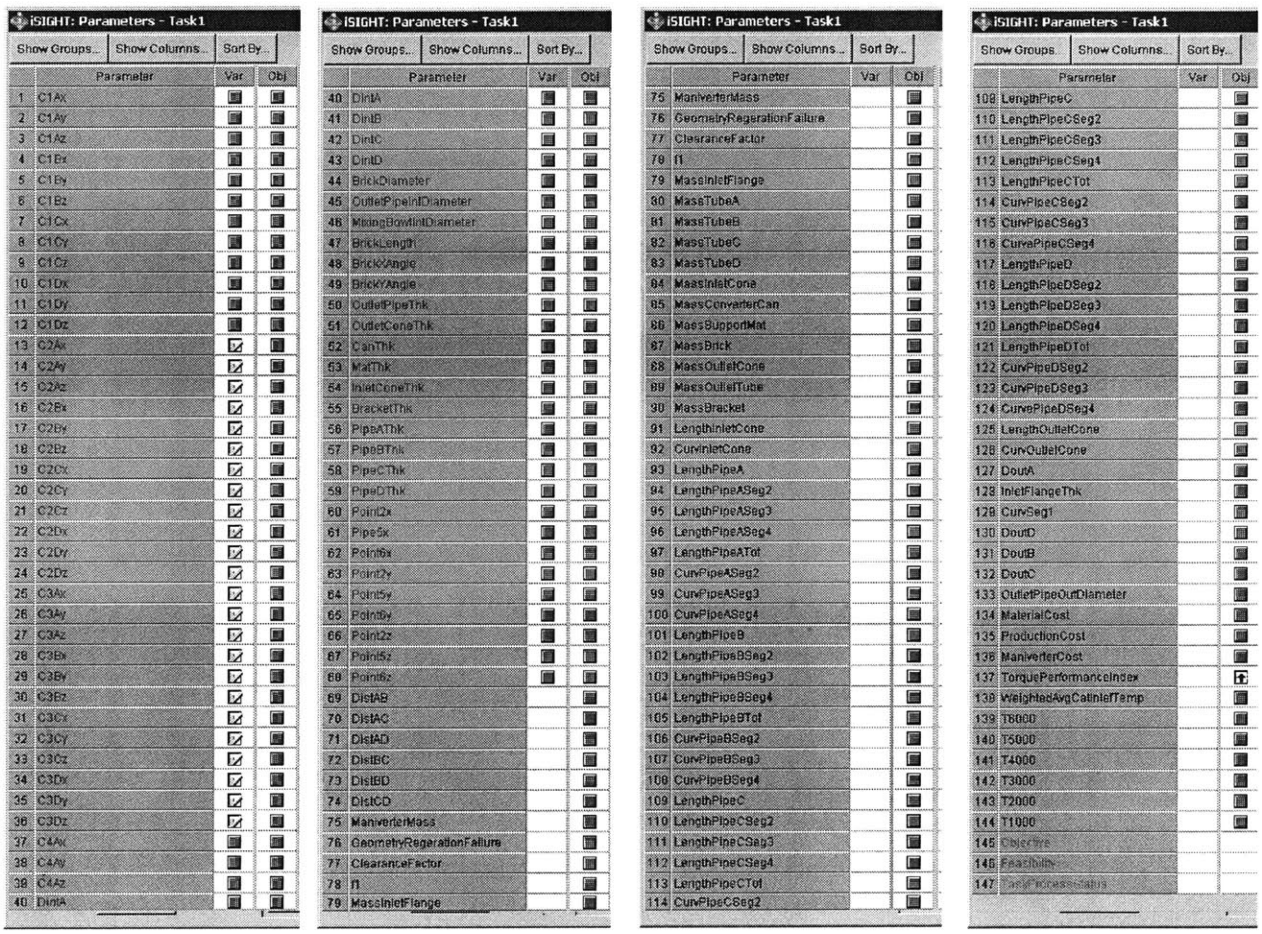
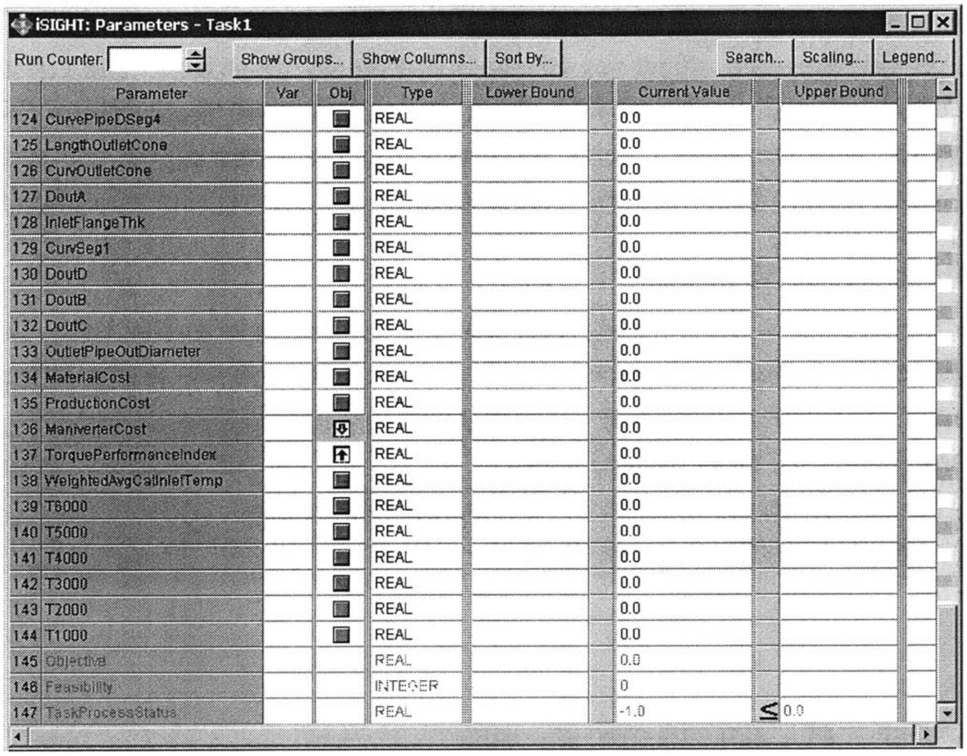


Fig. 101: ICE Platform Parameters List

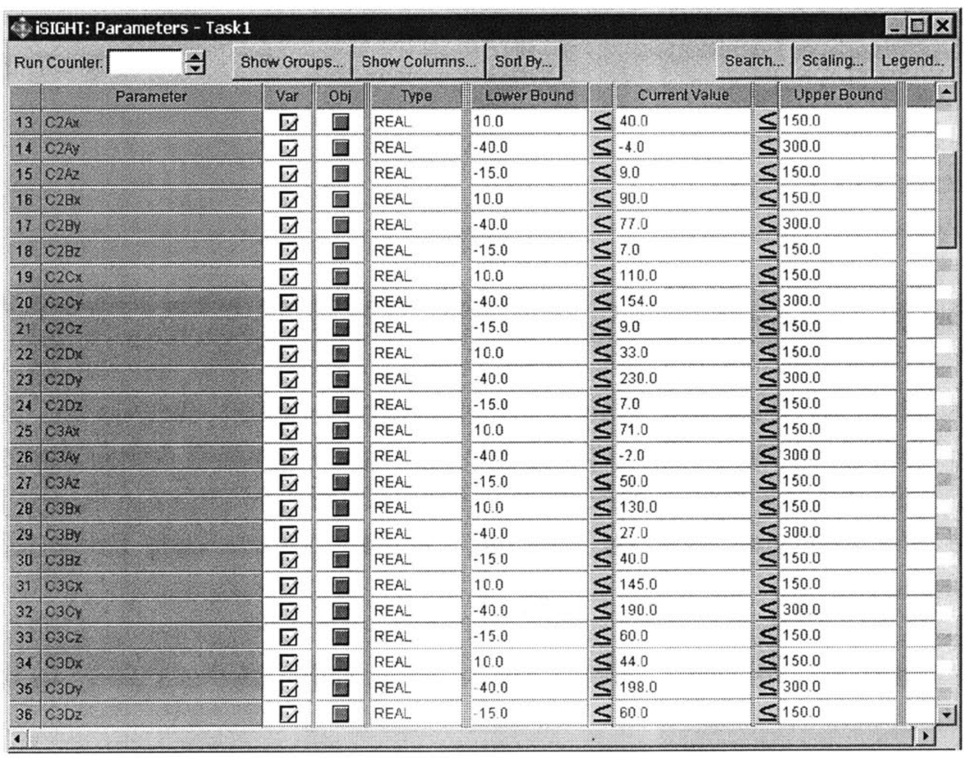


iSIGHT: Parameters - Task1

Run Counter: | Show Groups... | Show Columns... | Sort By... | Search... | Scaling... | Legend...

	Parameter	Var	Obj	Type	Lower Bound	Current Value	Upper Bound
124	CurvePipeDSeg4			REAL		0.0	
125	LengthOutletCone			REAL		0.0	
126	CurvOutletCone			REAL		0.0	
127	DoutA			REAL		0.0	
128	InletFlangeThk			REAL		0.0	
129	CurvSeg1			REAL		0.0	
130	DoutD			REAL		0.0	
131	DoutB			REAL		0.0	
132	DoutC			REAL		0.0	
133	OutletPipeOutDiameter			REAL		0.0	
134	MaterialCost			REAL		0.0	
135	ProductionCost			REAL		0.0	
136	ManiverterCost			REAL		0.0	
137	TorquePerformanceIndex			REAL		0.0	
138	WeightedAvgCatInletTemp			REAL		0.0	
139	T6000			REAL		0.0	
140	T5000			REAL		0.0	
141	T4000			REAL		0.0	
142	T3000			REAL		0.0	
143	T2000			REAL		0.0	
144	T1000			REAL		0.0	
145	Objective			REAL		0.0	
146	Feasibility			INTEGER		0	
147	TaskProcessStatus			REAL		-1.0	≤ 0.0

Fig. 102: Example of Goals Setting



iSIGHT: Parameters - Task1

Run Counter: | Show Groups... | Show Columns... | Sort By... | Search... | Scaling... | Legend...

	Parameter	Var	Obj	Type	Lower Bound	Current Value	Upper Bound
13	C2Ax			REAL	10.0	≤ 40.0	≤ 150.0
14	C2Ay			REAL	-40.0	≤ -4.0	≤ 300.0
15	C2Az			REAL	-15.0	≤ 9.0	≤ 150.0
16	C2Bx			REAL	10.0	≤ 90.0	≤ 150.0
17	C2By			REAL	-40.0	≤ 77.0	≤ 300.0
18	C2Bz			REAL	-15.0	≤ 7.0	≤ 150.0
19	C2Cx			REAL	10.0	≤ 110.0	≤ 150.0
20	C2Cy			REAL	-40.0	≤ 154.0	≤ 300.0
21	C2Cz			REAL	-15.0	≤ 9.0	≤ 150.0
22	C2Dx			REAL	10.0	≤ 33.0	≤ 150.0
23	C2Dy			REAL	-40.0	≤ 230.0	≤ 300.0
24	C2Dz			REAL	-15.0	≤ 7.0	≤ 150.0
25	C3Ax			REAL	10.0	≤ 71.0	≤ 150.0
26	C3Ay			REAL	-40.0	≤ -2.0	≤ 300.0
27	C3Az			REAL	-15.0	≤ 50.0	≤ 150.0
28	C3Bx			REAL	10.0	≤ 130.0	≤ 150.0
29	C3By			REAL	-40.0	≤ 27.0	≤ 300.0
30	C3Bz			REAL	-15.0	≤ 40.0	≤ 150.0
31	C3Cx			REAL	10.0	≤ 145.0	≤ 150.0
32	C3Cy			REAL	-40.0	≤ 190.0	≤ 300.0
33	C3Cz			REAL	-15.0	≤ 60.0	≤ 150.0
34	C3Dx			REAL	10.0	≤ 44.0	≤ 150.0
35	C3Dy			REAL	-40.0	≤ 198.0	≤ 300.0
36	C3Dz			REAL	-15.0	≤ 60.0	≤ 150.0

Fig. 103: Example of Parameters Upper / Lower Bounds setting

Both design variables and objectives can be subjected to constraints. The constraints are expressions of boundaries or sets of values that parameters must reside within. Boundaries values are chosen with two main criteria: the maniverter fits in the engine compartment and collisions between pipework and converter are mitigated. A sample is shown in Fig. 103.

3.10.4. Available Design Strategies

iSIGHT offers a suite of design study tools that can be thought of as the design intelligence engine that drives the design exploration process. The Task Plan, shown in Fig. 104, allows the user to define a sequence of steps that can utilize any number of these design study tools, in any combination. The classes of design study tools provided in iSIGHT are:

- Sampling
    - Design of Experiments (DoE)
    - Monte Carlo Simulations
- Performance
    - Optimization Techniques
    - Trade-off Analyses
- Quality Engineering Methods
    - Reliability Analysis
    - Reliability Optimization
    - Taguchi Robust Design
    - Six Sigma Robust Design

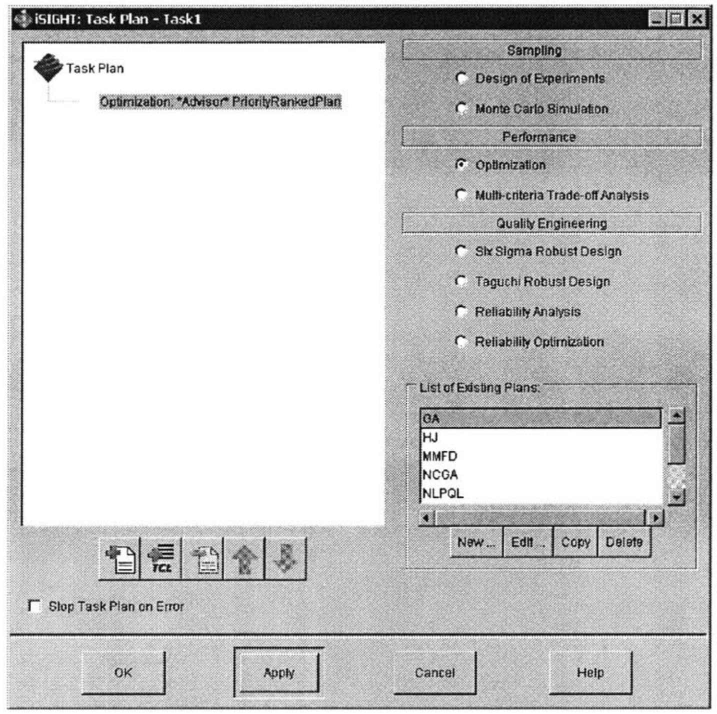


Fig. 104: Design Study Options available in iSIGHT

Individual plans for DoE, optimization, Multi-criteria Trade-off analysis and the Quality Engineering Methods can be created and added to the Task Plan, which defines the sequence of design tool application for design exploration. Approximation models can be applied at any stage in the Task Plan, and solutions from one stage are automatically fed to the next for complete automation.

In what follows we will focus only on design optimization methods since, in the present work, only those were exploited.

It's a well-established conclusion that no single optimization technique works best for all design problems. Instead, a combination of techniques can provide the best opportunity for finding an optimal solution. Seventeen different algorithms are available to be used in any planned design strategy. As outlined in Fig. 105, they are divided in numerical or gradient based methods and heuristic techniques. They can be used singularly or combined. For the problem at hand we will see that we identified an optimum strategy composed of several gradient search steps followed by a multi-objective genetic algorithm Pareto set extraction. Multi-Objective Genetic Algorithms (MOGA) are actually the only available algorithms to search for Pareto optimal solutions.

TYPE		TECHNIQUE
Numerical	1	Method of Feasible Directions – CONMIN
	2	Modified Method of Feasible Directions (MMFD) – ADS
	3	Sequential Linear Programming (SLP) - ADS
	4	Exterior Penalty - ADS
	5	Sequential Quadratic Programming (SQP) – DONLP
	6	Sequential Quadratic Programming (SQP) – NLPQL
	7	Mixed Integer Optimization – MOST, Tseng (1996)
	8	Hooke-Jeeves Direct Search Method
	9	Successive Approximation Method (LPSOLVE)
	10	Generalized Reduced Gradient (LSGRG2)
	11	Downhill Simplex
Heuristic	12	Multi-Island Genetic Algorithm
	13	Multi-objective Genetic Algorithm (NCGA)
	14	Multi-objective Genetic Algorithm (NSGA II)
	15	Adaptive Simulated Annealing
	16	Directed Heuristic Search (DHS – U.S. Patent No. 6,086,617)
Hybrid	17	Pointer Automatic Optimizer

Fig. 105: Optimization Algorithms available in iSIGHT

3.11. Post Processing: visualizing results data

Design exploration and Multi-disciplinary Design Optimization processes generate a huge amount of data, which, depending on the multi-dimensionality, can easily surpass the human cognitive capabilities. Hundreds or thousands design alternatives can be analyzed in a MDO process and each design can be characterized along several dimensions.

To enable effective engineering decisions, an appropriate method to transform data to information and to knowledge is required. Failing to recognize this will impair any previous analysis effort. That's why design data visualisation is an integral and fundamental part of the Enhanced Development Framework.

3.11.1. Introduction

Visualization exploits the powerful human visual system to effectively transport information from the outside world to the human apparatus of perception, recognition, cognition, and reasoning [67]

Visually displaying data with two or three dimensions is very common. Humans can easily recognize structures in the data (such as correlations in a scatter plot, trends in a line chart, etc.) and get a better impression of the data from images than from reading numbers. The most straightforward way to generate an image of unstructured data points is with a simple scatter plot. Traditional scatter plots capture the data in a 2-D or 3-D space. Because each variable requires its own dimension, these plots can only display 2 or 3 variables.

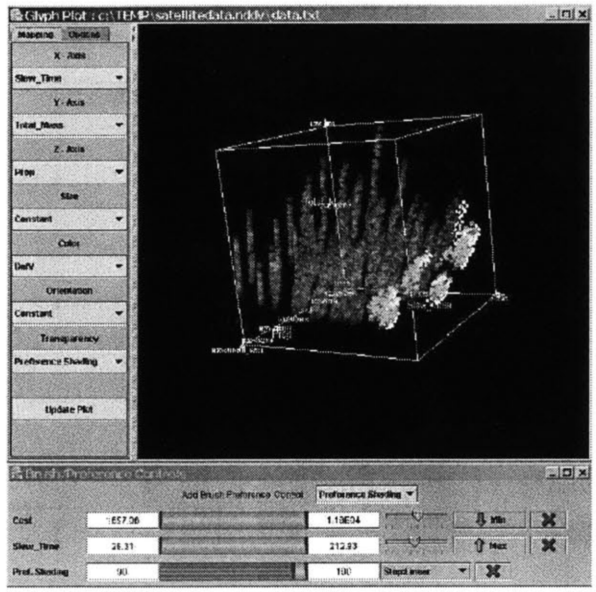


Fig. 106: An example of Glyph

The visualisation of 3+D data poses new challenges and research is active in this field. Some of the contributions could be found in [68] [69] [70] . The most common approach is in the direction of transforming raw data into glyphs that are plotted in a 3-D space. A glyph is a visual object onto which many data parameters may be mapped, each with a different visual attribute. Generally, additional dimensions (variables) beyond the standard 3 orientation axes can be mapped onto glyphs through: (1) scalar mapping, and/or (2) color/texture mapping. An example of glyph is shown in Fig. 106 [68] ; on it Pareto data are also shown as brighter points.

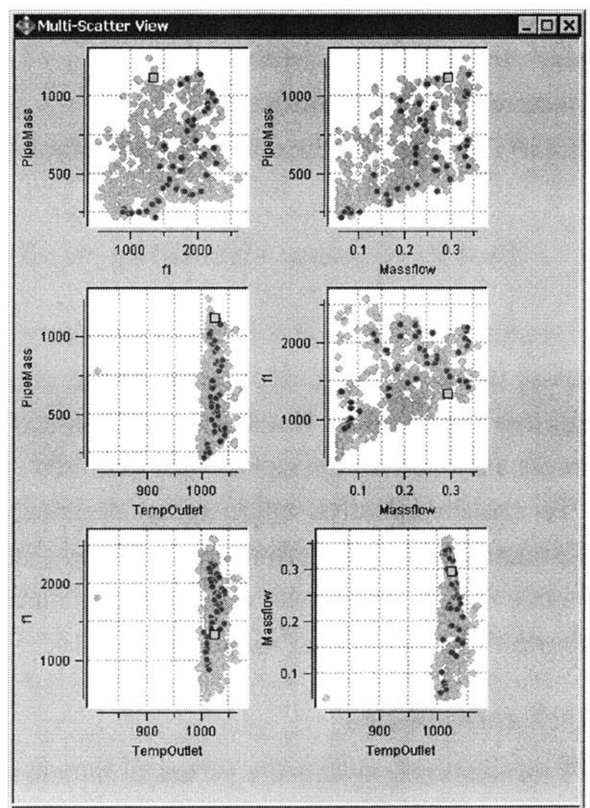


Fig. 107: The Scatter Plots Capability in iSIGHT

Even if glyphs are powerful to visualize complex data, for our work, we looked at tools that were extremely easy to use and intuitively friendly. First, we analyzed the visualization tools available in iSIGHT and then developed a simple and yet powerful and intuitive technique for Pareto data visualization and decision-making.

3.11.2. iSIGHT Visualization tools: Engineering Data Mining and Scatter plots

In iSIGHT the Engineering Data Mining application allows to produce scatter plots and to visualize Pareto points. In Fig. 107, the results for the simple pipe application are shown (for more details on the results, see Section 4.2). The four performance attributes of the more than 800 design alternatives are plotted in a light blue color. The darker points are the 50 Pareto designs (Fig. 107).

The tool is such that, passing with the mouse over a design point, the corresponding quantitative data of both the design variables and the performance attributes are highlighted, Fig. 108.

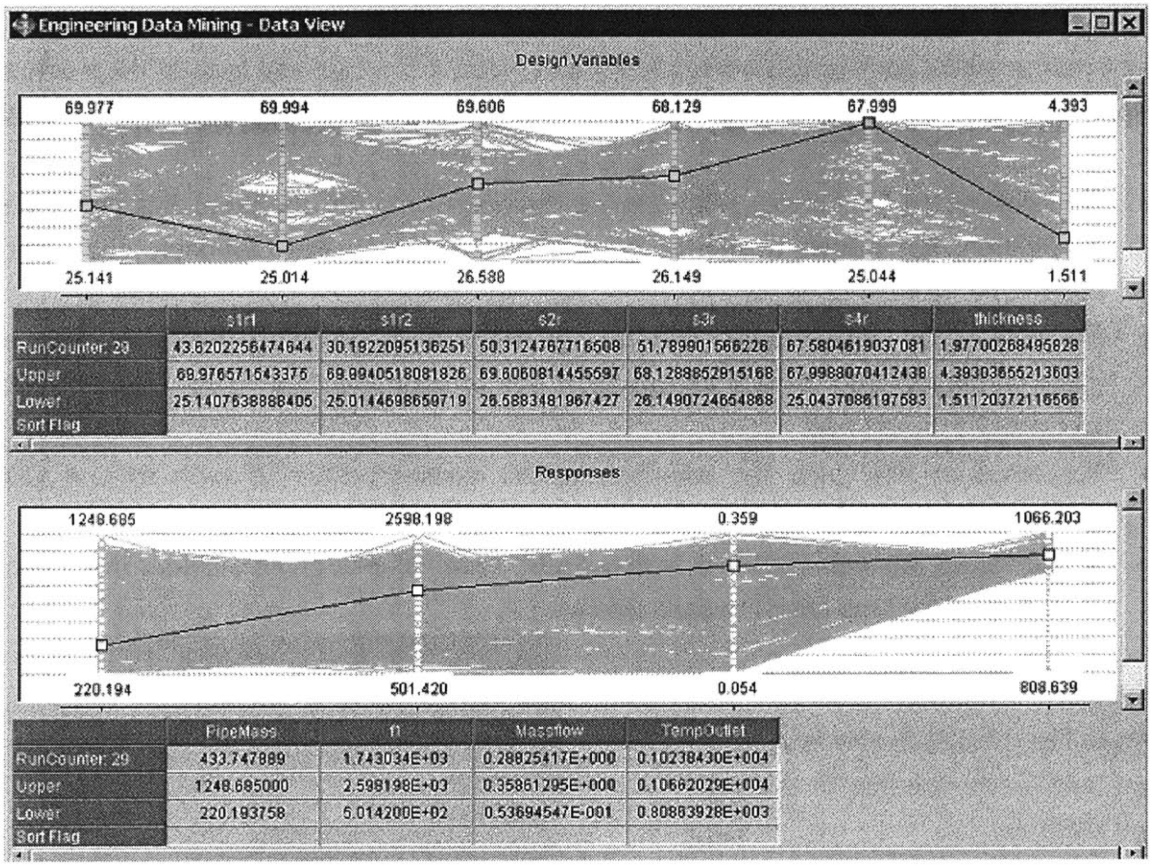


Fig. 108: The Engineering Data Mining Tool in iSIGHT

While these are powerful tools, it's our opinion that they are not able, alone, to reduce the complexity to such a level that is "humanly solvable" by the "average engineer". Since the ambition of the automatic development approach is to be used in mainstream application by project engineers with no special skill, we searched for alternatives.

3.11.3. An intuitive tools for engineers: the Rainbow Plot

Solutions to an engineering problem can be divided into two sets: dominated solutions and not dominated, or Pareto, solutions. We are particularly interested in the set of non-dominated solutions; in fact a non-dominated solution is at least as good as all other solutions on all criteria and better on at least one criterion. Non-dominated solutions define the efficient frontier of the solution space. All solutions lying on the efficient frontier are potentially preferred by the decision makers and in order to ascertain which is actually preferred it is necessary to take into account the decision makers preferences. Since engineering decisions involve the resolution of design trade-offs, our process is to identify the Pareto data first and to present them to the user to apply final preference weights and select the "best" solution.

More specifically, raw data coming out of any MDO analysis can essentially be though o as filling a matrix where each of the *m* lines refers to a particular design and each of the *n* columns is a specific response. Data are displayed to the user after the following complexity-reducing steps:

- Non-dominated (or Pareto) solutions are extracted from the complete data pool. In our application, this allows to reduce the size of the data by approximately one order of magnitude, from *m* to *m/10*.
- For each of the performance attributes, the best and the worst are identified; then each value is ranked from 0 to 1 according to the proximity to the "best" point: 0 means worst and 1 means best
- Processed in this way, the solutions matrix contains, now in each row, *n* values comprised between 0 and 1
- These values are displayed in a rainbow plot: each value is given a colour in a rainbow scale from dark blue (zero) to dark red (one).

A typical resulting plot is shown in Fig. 109: each row is a design and each column is a design attribute.

The psychological process that the engineer is supposed to follow when analyzing the data is the following. Knowing that red corresponds to "good" and blue to "bad", using the powerful pattern recognition of the human eye and brain, the

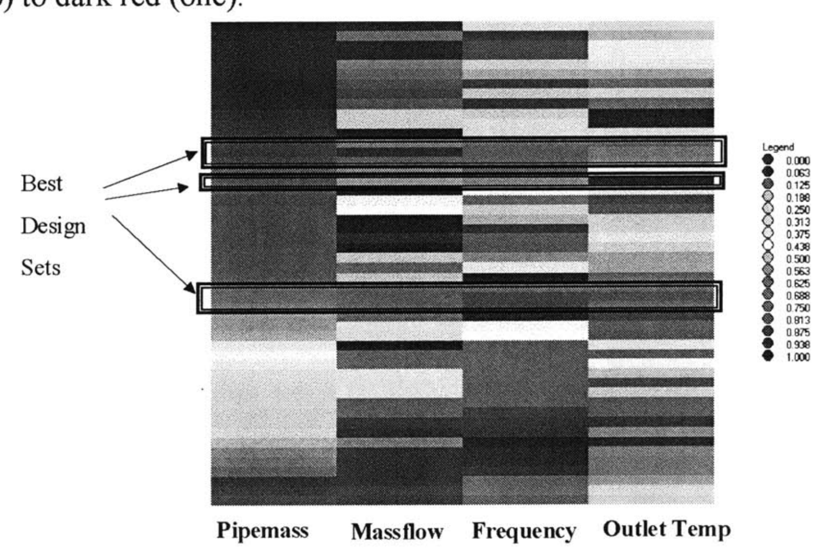


Fig. 109: Qualitative Rainbow Plot of Simple Pipe Pareto Designs

engineer can scan the rows looking for the darkest red combination (this also can be automated).

In this phase, further narrowing down of solutions could be obtained by applying preferences. For example, the first two pools of solutions (from the top) are to be preferred if pipemass and massflow are the dominant criteria for solution selection, the second pool of solutions is probably preferable if frequency or outlet temperature are the main drivers.

In the specific case presented, the original more than 800 solutions have been now reduced to less than 10 which can then further analyzed by using quantitative approaches.

A different variant of the presented rainbow plot consists in retaining numerical values in combination with colors, Fig. 110. The highest values gets the strongest colour. Zero values receive no colour. The color allows quick selection of the few design alternatives, while data analysis provides the fine-tuning.

Rainbow plots can be easily generated with Poptools, a popular Excel add-in for statistical analysis and data visualization available as a freeware. However, we deem more research is needed to explore visualization methods that, like rainbow plots, exploit two powerful human cognitive mechanisms: pattern recognition and color mapping (i.e. association of color to information).

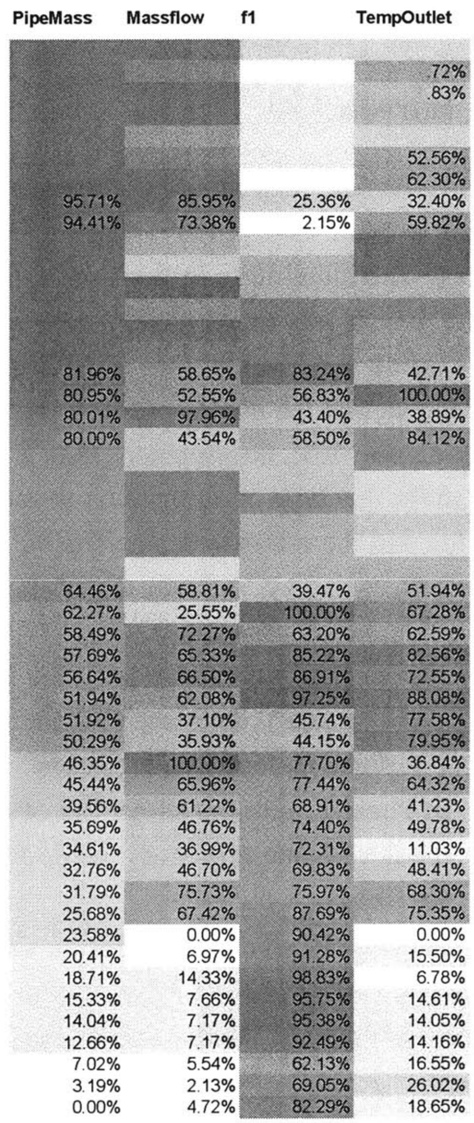


Fig. 110 Quantitative Rainbow Plot of Simple Pipe Pareto Designs

4. USING THE TOOL: DEVELOPING PRODUCTS EFFICIENTLY

> *Real business entails adding value to things by adding knowledge to them...*
>
> Akio Morita, co-founder of Sony

4.1. Introduction

In this Section, making use of the developed Integrated Concurrent Engineering platform we will use a Multi-disciplinary Design Optimization approach to identify maniverter design solutions. The identified solutions are compared, in terms of performance and piece cost with a baseline, which was tuned to reproduce an actual design that ArvinMeritor developed in 2002 for the same application (see Fig. 70), with the purpose to get a sense for the design improvement that the novel approach is capable of delivering.

At the same time, development time is recorded and compared with a standard ArvinMeritor leadtime for this type of application to estimate the reduction in development costs and in time to market and the increase design flexibility enabled by the use of this Enhanced Development Framework.

The design goal was to identify the Pareto optimal solutions. Given the high number of variables (48 in total), the search for the most efficient frontier (which, in this case, is actually an hyper-surface) is done in two phases:

- In Phase I, only the centerline of the maniverter piperuns are allowed to change during the development process, while their diameters, thickness and the rest of the converter are fixed
- In Phase II: all available design variables are considered variable

Before jumping to the maniverter application, however, we would like report some selected results from a much simpler application that was used in a Phase 0 of this project as a trainer both for the ICE platform building and in the MDO approach: a simple pipe. This application was instrumental in developing much of the knowledge that was used subsequently in building the definitive ICE platform and in the testing Phase I and Phase II. Despite its simplicity, we believe that it brings some interesting general insights.

4.2. The trainer: a simple pipe

4.2.1. System's Description

The targeted system for Phase 0 is a simple pipe, see Fig. 111 (which is just a copy of Fig. 18, but it's reported here, enlarged, for ease of reading).

The pipe has a centerline defined by a cubic spline with four control points and four control cross sections: the first is a racetrack and the other three are circular.

For this simplified problem, the design variables are:

- The two dimensions of the racetrack section and the diameters of the other three
- The pipe thickness
- The 4 control points, each of which has 3 coordinates

The pipe is imagined connected rigidly to a wall in correspondence to the racetrack section; hot gases enter the pipe at a fixed temperature from it and flow to the outlet round section, under the action of a constant pressure differential.

A simplified ICE platform is built around this application, enabling the calculation of the following four performance attributes:

- Pipe mass: the lower the better
- First natural frequency: the higher the better
- Massflow rate: the lower the better
- Outlet gas temperature: the higher the better

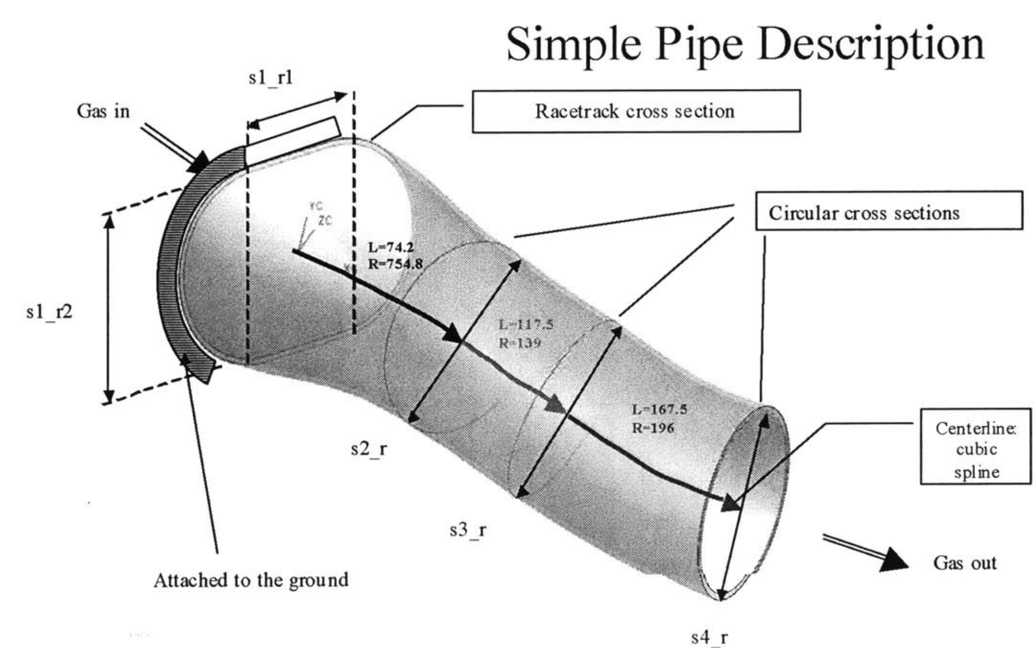


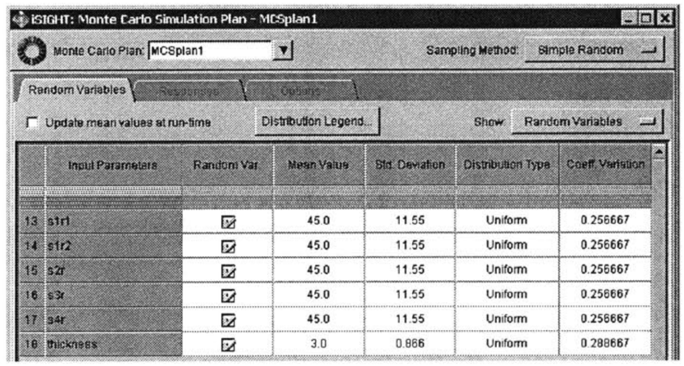
Fig. 111: The trainer: a Simple Pipe

In the analysis, the pipe centreline coordinates are considered fixed; consequently the design variables are reduced to six.

While this is a simplified application, the software programs that were used are the same of the mainstream application, i.e. UG for the CAD model, MSC.Patran and MSC.Nastran for the structural analysis (the pipe is a cantilever beam) and AVL BOOST for the fluid dynamic analysis.

4.2.2. Results

Several types of analysis were performed to test the capability of the tool to manage the trade-off intrinsic in the system. Here we report only some samples useful to illustrate some insights that were gained.



mean = (a + b)/2

$$\text{standard deviation} = \frac{b - a}{2\sqrt{3}}$$

In the two equations above, a is the lower limit of the uniform distribution, and b is the upper limit.

Fig. 112: Monte Carlo Analysis setup: variables mean values and boundaries

Mass-Frequency Trade-off

A Monte Carlo analysis was performed, Fig. 112. Design variables were varied in a given range and the mass-frequency performance attributes space mapped, see Fig. 113.

Even if strictly valid for this simple system and within the specific analysis, we can draw some conclusions that we believe are general:

- Designs can be made that, for a given level of a performance attribute, exhibit a great variability in other dimensions. In the specific example, for a given frequency 1500 Hz resonance frequency, pipes with a mass of 220g or 1000g are possible, or, vice versa, for a given pipe mass of 500 g, pipes with a resonance frequency of 500 Hz or

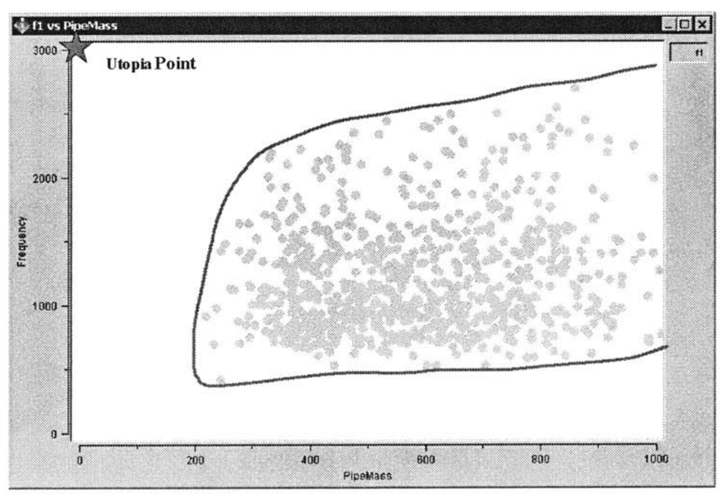


Fig. 113: Monte Carlo analysis: Frequency-Mass Trade-Space

2500 Hz are possible

- Experience drives the design towards specific areas of the design space.
- Experience is not sufficient to make designs with exceptional performances but proper tools are required. In the considered example, we can use the designs density as a proxy for the likelihood of the outcome of the design effort with very little experience. We see that the points' cloud rarefies as we move towards the Pareto front, suggesting that, even with a good level of expertise, as we push the design targets to the limit, it becomes harder and harder to identify a design solution.
- The shape of the Pareto front suggests where to stop the development: further increasing a performance attribute cannot be worth the effort. With reference to Fig. 113, it could be reasonable increasing the mass of a the pipe from a minimum of 200 g to 300 g because, by doing that, a Pareto solution exists which has a frequency of 2200 Hz, much higher than the 500 Hz of the 200 g pipe. If we wanted to augment further the natural frequency, for a minor increase from 2200 Hz to 2500 Hz, we should be prepared to accept a huge mass penalty, from 300 g to 600 g.

Massflow - Gas outlet temperature

Depending on the morphology of the performance attributes space a "sweet spot" might exist.

Fig. 114 depicts the massflow rate vs. gas outlet temperature trade-space.

The maximum outlet gas temperature is achieved with a particular value of the massflow rate; higher or lower massflow rate values reduce the gas outlet temperature.

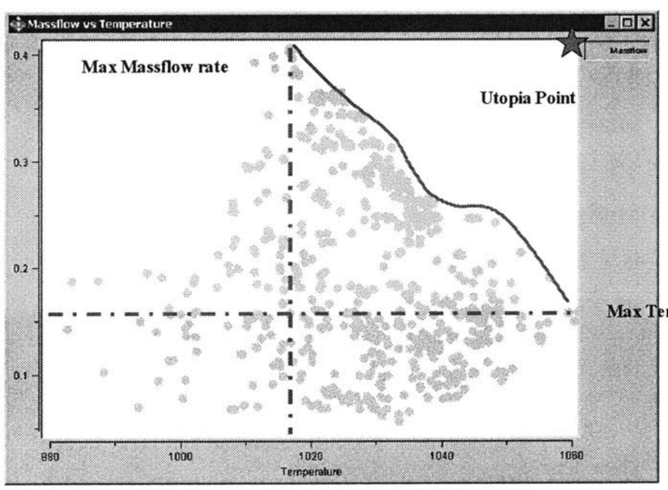


Fig. 114: Monte Carlo Analysis: Massflow-Outlet Temperature Trade-Space

In this particular case, the observed behaviour is explained as an alternation of dominance of two physical phenomena. If we imagine decreasing the pipe diameter, up to a certain point the gas outlet temperature increases due to reduced heat loss caused by the reduced pipe surface. Reducing further the diameter constrains so much the massflow that the amount of heat injected in the gas stream is not sufficient to maintain a high temperature.

4.3. Phase I: Testing the EDF on the maniverter

The Enhanced Development Framework has been subjected to a set of tests to verify possible issues and to identify the best combination of optimization algorithms for computing the Pareto hyper-surface. To avoid the complexity of the problem to hinder the understanding of the behaviour of the system and of the tool, in this preliminary testing phase we allowed only the control points of the centerline pipes splines to vary, while pipe diameters and the rest of the system was considered fixed. This means a total of 24 design variables (2 control points per each pipe, with 3 coordinates for each point; the other 2 control points of each spline, the first and the last, were fixed).

Intermediate control point design variables are given upper and lower boundaries allowing them to vary inside a block, see Fig. 115.

The execution time of a complete simulation loop is about 20 min, split in:

- 0.8 min for geometry regeneration
- 3.5 min for structural analysis
- 15 min for fluid dynamic analysis
- 0.7 min for Excel execution and optimization algorithms

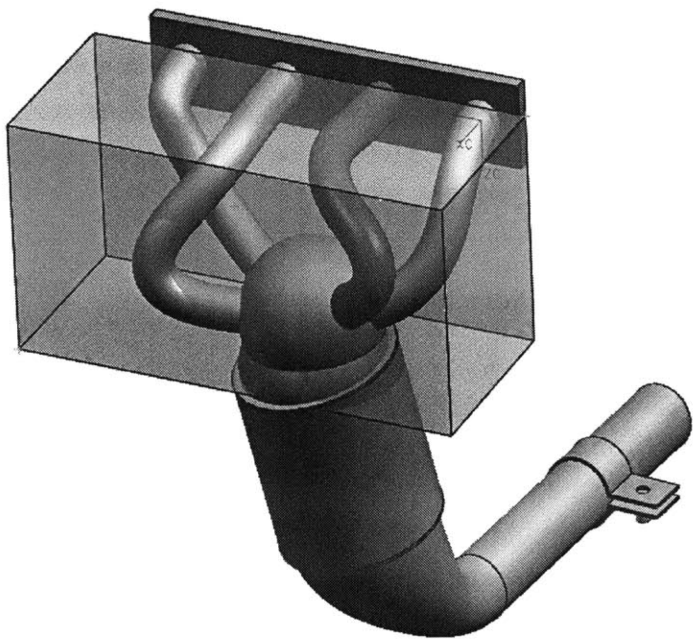


Fig. 115: Pictorial representation of boundaries range for pipework control points

4.3.1. Design Space Exploration

Whenever a new problem is tackled, for which very little is known about the design space and the behaviour of the system, good practice suggests exploring it in a systematic manner. Typical techniques used for this design exploration activity are DoE or Monte Carlo analysis. In DoE, sample points are selected in the design space (several methods exist for point selection) and performance attributes computed. In Monte Carlo analysis, design variables are varied around baseline points according to a probability distribution.

Design Space exploration can be an effective technique to locate the zones where optima are; in addition the sensitivity of performance attributes to the different design variables can often be easily established.

When we tried to execute both DoE and Monte Carlo analysis in our EDF, however, we suddenly hit against a major roadblock: geometry regeneration failures. As mentioned in 3.6.2, the geometry is governed by parameters which are not completely independent but related by a loose correlation. Let's illustrate the issue with a simple 2-D example.

If we have a pipe whose centerline is defined by 4 control points which are allowed complete freedom in the design space, we can select a combination which gives the path shown in Fig. 116. If the pipe has the diameter D shown, we intuitively understand that the pipe cannot be geometrically generated: the curve between point 3 and point 4 is, in fact, too tight. We, therefore, intuitively grasp that a loose coupling exists between the selected variables, even though the relationship cannot be straightforwardly determined.

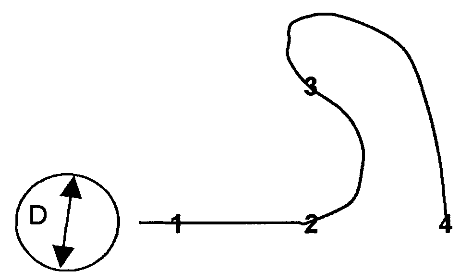


Fig. 116: an example of unfeasible geometry

Generalizing, we can image the design space as defined by "feasibility channels" where certain combinations of design variables give feasible solutions, delimited by ridges beyond which no solutions exist.

To partly uncover the nature of this channel-like design space without being stopped in the unfeasibility traps, a pseudo-Monte Carlo analysis is run.

The strategy is the following. Starting from a baseline feasible design configuration, a "local" Monte Carlo analysis is performed where design variables are allowed to change by +/- 20% with respect to the baseline. This guarantees that a sufficient percentage of the runs are feasible. Then, the farthest point in the design space is selected, i.e. the one which has got the highest design variables distance $\sqrt{\sum_n (x_i - x_0)^2}$, from the baseline and a new Monte Carlo analysis is run. A fixed number of advancement steps (100) and "local Monte Carlo analysis" (10) are set, for a total of 1000 runs.

Fig. 117 collects a sample of 9 of the 276 pairwise scatter plots of the 24 design variables.

The points where the geometry generation failed are indicated by circles, the feasible points by full dots. The direction of the stochastic path is given by the arrows.

Feasibility channels clearly emerge. May different patterns appear: some of the variables are poorly or non-correlated at all, some of them are loosely correlated and some others are highly correlated.

This finding somewhat confirms the intuition we had. The situation is further complicated by the fact that channels shape and dimensions change with the other design variables that were fixed in this testing phase, i.e. the pipe diameters and thickness.

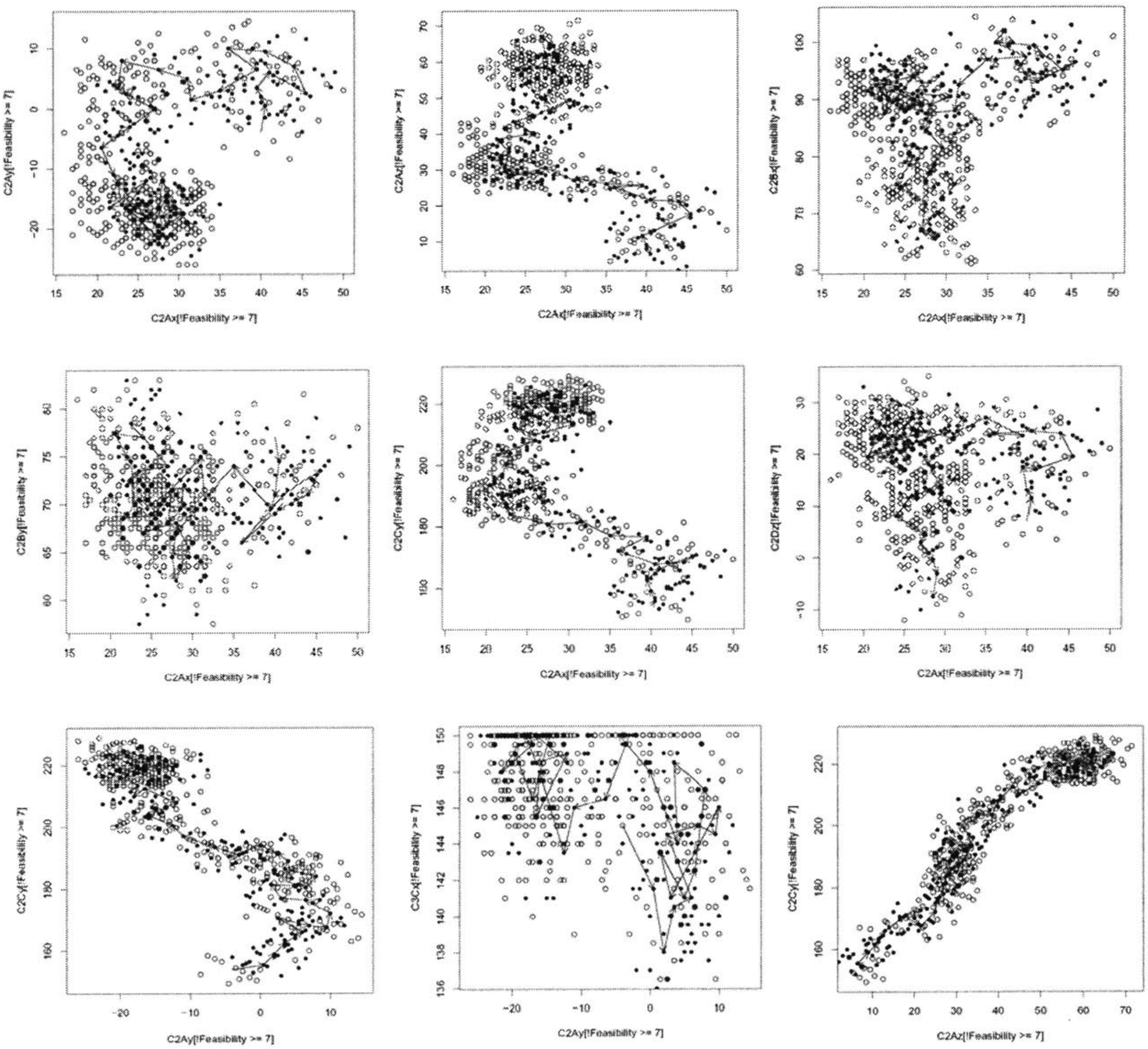


Fig. 117: Feasibility Channels

When running a DoE or a Monte Carlo analysis, values of the design variables are selected randomly in the whole design space. The morphology of the design space made by unpredictable narrow feasibility channels causes the majority of random combinations to generate unfeasible geometries. Out of a Monte Carlo trial we did with uniform probability distribution, we estimate that the feasible runs are about 1-3% of the total number of runs. Even though geometry regeneration time is remarkably low (<1 min), to have any given number of feasible geometry sets, a two order of magnitude runs would have to be performed.

Consequently, for this application, traditional design space exploration techniques are not affordable. Given the peculiar shape of the design space, on the other hand, gradient-based techniques proved to be effective.

A preliminary set of single objective optimization runs using gradient-based methods was performed for different performance attributes. The results of this activity were several:

- The design space was partially explored
- Some improved design were found
- The most appropriate gradient-based algorithm for the application was identified.

The following Section describes the details of the various runs.

4.3.2. Single Objective Optimizations

A particular effective method in "riding" the channels proved to be Hooke-Jeeves one. The Hooke-Jeeves search, in fact, is made especially for ridge-following. Its strength is that it is able to find the ridges itself and can recover if a ridge comes to an end. Some details of the algorithm are given in the Section that follows.

The Hooke-Jeeves algorithm

The search method developed by Hooke and Jeeves [71] , known also as pattern search, takes advantage of the fact that most response surfaces have one or more ridges which lead to the optimum. Thus the purpose is to find a ridge and follow it to the optimum. In pattern search the search begins by exploring the response surface in the vicinity of a selected base point. With repeated success the explorations become longer taking advantage of an established pattern. Failure to improve the criterion, however, indicates that one must abandon the old pattern and try to find a new one, which will be followed until the pattern is broken again and the process has to be repeated. The so determined pattern will coincide with the ridge. In the neighbourhood of the optimum, the steps become very small to avoid overlooking any promising directions. The optimum is reached and the search terminates when the predetermined final step size fails to improve the criterion.

The objective function is not required to be continuous. Because the algorithm does not use derivatives of the objective function, the function does not even need to be differentiable.

This technique has a convergence parameter, ρ, which lets you determine the number of function evaluations needed for the greatest probability of convergence. This parameter sets the step size reduction factor and has a value between 0.0 and 1.0. Larger values of ρ give greater probability of convergence on highly nonlinear functions, at a cost of more function evaluations. Smaller values reduce the number of evaluations (and the program running time), but increase the risk of non-convergence. The default value in iSIGHT is 0.5.

Being a direct numerical search algorithm, H-J is prone to be trapped in a local minimum. However, repeated searches from different starting points or searches for different values of ρ reduce the likelihood of the optimum being a local extreme point.

Min mass

In this run, starting from the baseline configuration, the H-J algorithm is given the task to find the minimum mass configuration. ρ is set to the value of 0.3 to increase speed.

Structural and fluid dynamic analyses are switched off to save time, as they are unused in this run.

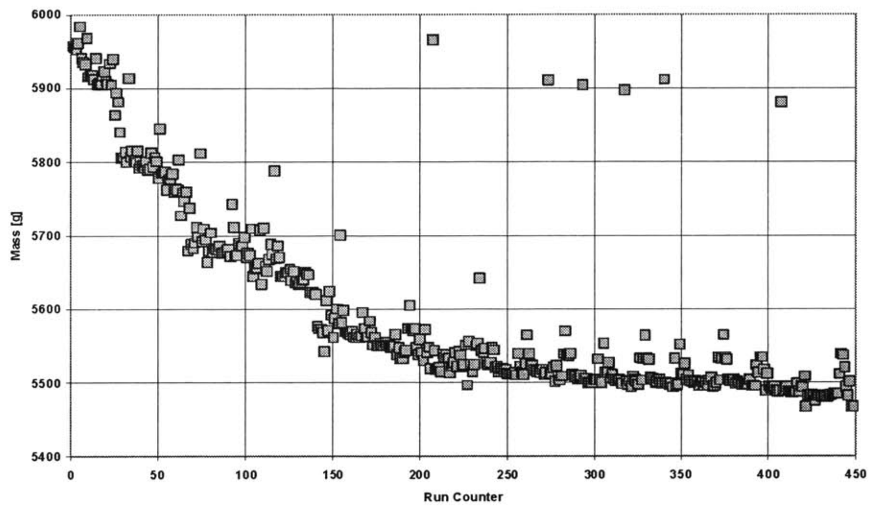


Fig. 118: Testing Phase - Maniverter Mass Minimization – Optimization History

With H-J feasible runs are about 70% of the total. The optimization history is shown in Fig. 118. Total run time is 7h on the selected hardware platform.

Total mass was reduced by 8% from 5946g to 5467g. However, if we take into account that all maniverter's design variables are fixed, but the piperuns, whose mass is 1224g, we get a more sensible figure of 39% mass reduction.

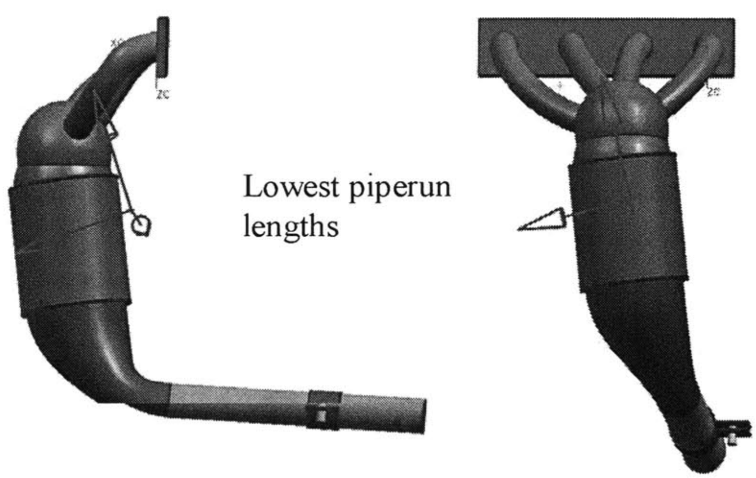


Fig. 119: Testing Phase - Maniverter Mass Minimization – Resulting Geometry

The resulting minimum mass geometry coincides with what the intuition suggests: piperuns are made as short as possible.

Max performance

In this run, starting from the baseline configuration, the H-J algorithm is given the task to find the configuration that ensure the engine the best performances, i.e. a design solution that has the max value of the performance index, the max ratio of the mean torque and of its standard deviation (see 3.8.4). ρ is set to the value of 0.3.

The optimization was stopped after 80h of runtime.

In Fig. 120, the performance index is plotted against the run counter. To understand how the optimization morphed the maniverter geometry, pipe lengths are also plotted (best fit polynomial approximations are superimposed to the analyses results for easier reading of the trend).

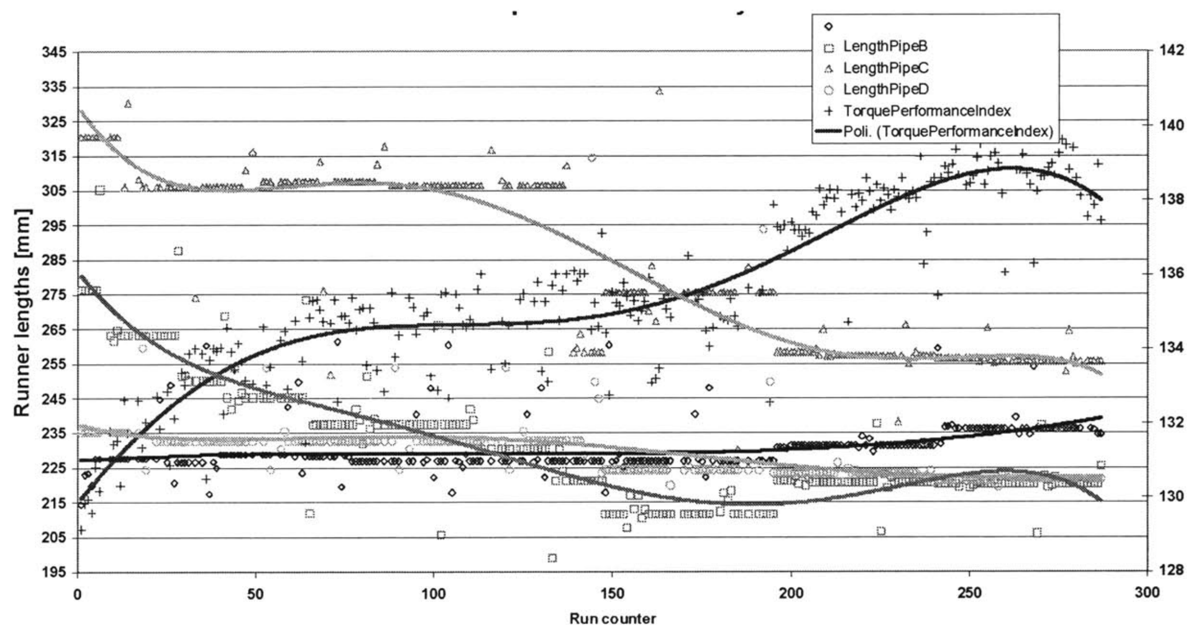


Fig. 120: Testing Phase - Performance Maximization – Optimization History

We observe that the optimization process led to an evening of pipe lengths. A closer look at the components of the performance index, i.e. the mean value of the torque and its standard deviation, reveals that the improvement arises from the reduction in the standard deviation, more than from the increase in the mean value. This result coincides with manifold design best practice, which recognizes the benefit of having even pipe lengths on the regularity of the torque.

In addition, the optimization kept pipe lengths above 200 mm. Again, this is confirmed by the design best practice, which gives exactly 200-220 mm as a threshold level: above this value power and torque level off, below it, they fall dramatically.

Last but not least, the analysis seems to head towards a differentiation of the piperuns, with two (namely, pipe A and C) about 30 mm longer than the others (pipe B and D). This may be the result of some kind of tuning.

Fig. 121: Testing Phase – Performance Maximization

The geometric configuration result of the optimization process is represented in Fig. 121. We note the interesting feature that the optimization shaped the piperuns to be, within the existing constraints, as straight as possible, with the minimum number of bends. This is, in fact, beneficial to reduce pressure losses. In addition, as we will see in the next Section, this strategy also reduces production costs.

Min cost

Starting from the baseline configuration, the H-J algorithm is given the task to find the design solution that has the lowest overall cost. ρ is set to the value of 0.3.

The optimization was stopped after 136 iterations and 50h of runtime.

The optimization history is shown in Fig. 122 while the resulting geometry is shown in Fig. 123.

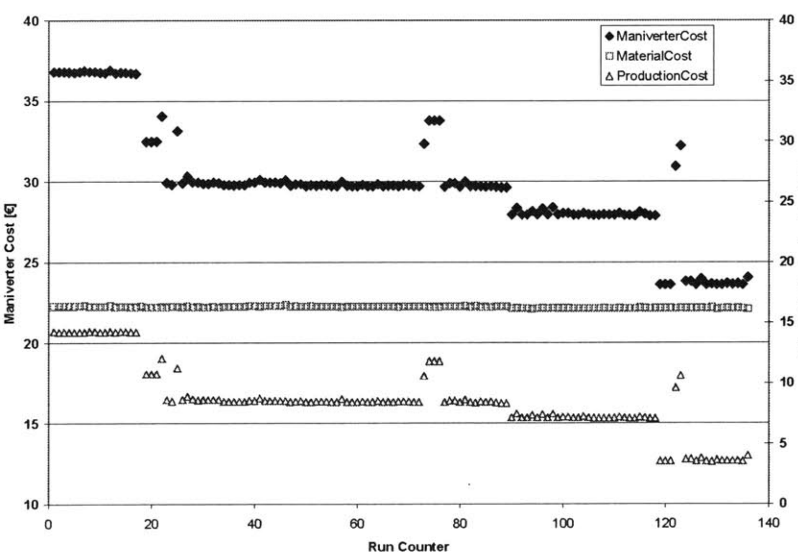


Fig. 122: Testing Phase – Cost Minimization – Optimization History

We note that maniverter cost was reduced essentially by leveraging production cost, while material cost remained essentially the same.

Production costs are reduced by reducing the number of bends and the bend angle.

In our opinion, the algorithm tackled the production cost because it found this was the easiest to reduce. However, if we let the run proceed further, we believe that the algorithm would have reduced material cost by shortening the pipes.

Even if with disguised cost figures, we note that the cost reduction has been remarkably high, from €36.8 to €24.6, i.e. nearly 35%. Given the thin profitability margins that currently exist in the automotive component industry, a difference of this order of magnitude, for this type of system, might mean a shift from a painful loss to an over the top profitability and great strategic positioning.

Since this result depends heavily on the cost model assumptions, a scrupulous analysis of the cost model is mandatory. However, it signals that significant business improvements can be gained through the optimization.

Fig. 123: Testing Phase – Cost Minimization – Resulting Geometry

4.3.3. Multi-Objective Analysis and Trade-space exploration

Single-objective runs allowed to get first feedbacks from the framework, to understand that the results it provides are sensible since they match common design experience and also offer intriguing insights from a business perspective.

Now, we can head straight to the original goal, which is Pareto front calculation through multi-objective optimization and trade-space exploration. Design variables are, by now, still only the 24 described in 4.3

In multi-objective optimization problems, several objectives are considered at the same time. When more than one objective is considered, the concept of optimum solution is replaced by the notion of the Pareto optimum set.

In this testing phase, we limited ourselves to pairwise analysis: out of the five different performance objectives, two pairs are selected for Pareto front calculation.

For multiobjective optimization, Genetic Algorithms (GAs) are very effective. These algorithms of multi-objective GA can be divided into two categories: the algorithms that treat Pareto-optimum solution implicitly or explicitly.

The majority of the latest methods treat Pareto-optimum solution explicitly. One of the most recent (2002), the Neighborhood Cultivation Genetic Algorithm (NCGA), was used for the maniverter multi-objective analysis.

The NCGA algorithm [72]

In the past few years, several new algorithms that can find good Pareto-optimum solutions with small calculation cost are have been developed. Typical algorithms are NSGA-II, SPEA2, NPGA-II and MOGA. These new algorithms have the same search mechanisms, preservation scheme of excellent solutions that are found in the search, allocation scheme of appropriate fitness values and sharing scheme without parameters.

In NCGA, each objective parameter is treated separately. Standard genetic operation of mutation and crossover are performed on the designs. The crossover process, in particular, is based on the "neighborhood cultivation" mechanism, where the crossover is performed mostly between individuals with values close to one of the objectives.

In the crossover operation of NCGA, a pair of individuals for crossover is not chosen randomly, but individuals who are close to each other are chosen. Because of this operation, child individuals which are generated after the crossover may be close to the parent individuals.

NCGA is a robust algorithm to find Pareto-optimum solutions. By the end of the optimization run, a Pareto set is constructed where each design has the "best" combination of objective values, and improving one objective is impossible without sacrificing on one or more of the other objectives.

NCGA is selected for maniverter development because, in addition of being a powerful GA for Pareto set extraction, its iSIGHT's implementation gives the possibility to give a start population that the algorithm evolves. This is the mechanism that we exploited to overcome the feasibility issue that was discussed in Subsection 4.3.1. If we tried to run any GA without a special initialization, since the starting population is generated through a random selection of design variables combination, we would fall into the same feasibility trap with many generated solutions characterized by unfeasibility.

What we interestingly found in our particular application is that "feasibility" seems to be a characteristic of the "DNA" of the any feasible solution. This has, as a consequence, that if the GA starts with a population of feasible solutions, it proceeds without encountering any major obstacle because, crossing over two "feasible" members results, in general, in feasible offspring.

From here the idea to generate feasible solutions with a pattern search algorithm such as H-J and then compose the population to be given as initial set to the NCGA. For a more comprehensive design space exploration, the initial population set should then made by design solutions at the edges of the performance attributes space.

Mass-Frequency Trade-off

Ten members of the entire population of the solutions generated during the mass minimization run are manually selected and given to the NCGA algorithm with the dual objective of minimizing the mass and maximizing the first frequency. The process is carried forward for ten generations for a total of 100 runs. Feasible solutions were more than 97% of the total number of generated designs.

The resulting Pareto plot is shown in Fig. 124. As we can see solutions immediately cloud around the utopia point.

The trade-off between frequency and mass does not appear to be as strong. In fact the shorter piperuns that the low mass design features also make the solution stiffer (the first mode is in fact a global lateral left-right swing).

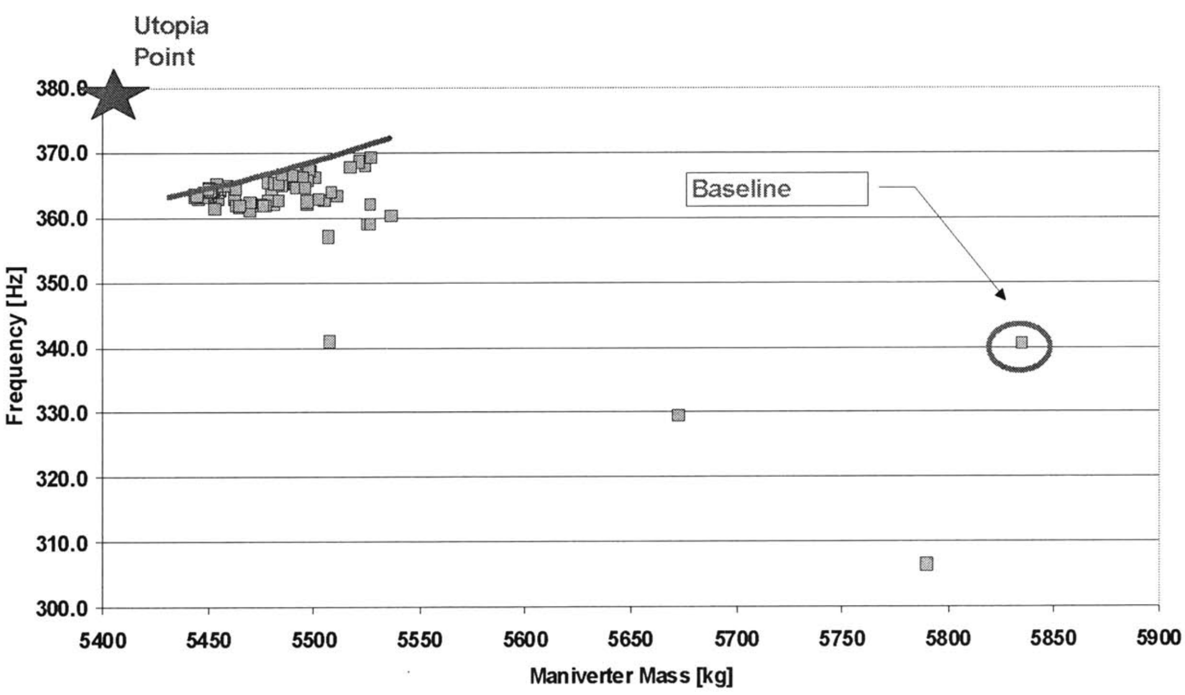


Fig. 124: Testing Phase – Frequency-Mass Trade-Space

Performance-Cost Trade-off

Here the two populations generated in the previous cost minimization and performance maximization runs are considered. Ten members are selected from both populations to form the starting set, which is then carried forward for 10 generations, for a total of 100 runs. About 90% of the runs were feasible.

In Fig. 125 results are reported in the Performance-Cost trade-space. Green squares identify the results of the previous max performance runs, purple squares are the results of the previous min cost run, orange squares are the solutions generated by the NCGA process.

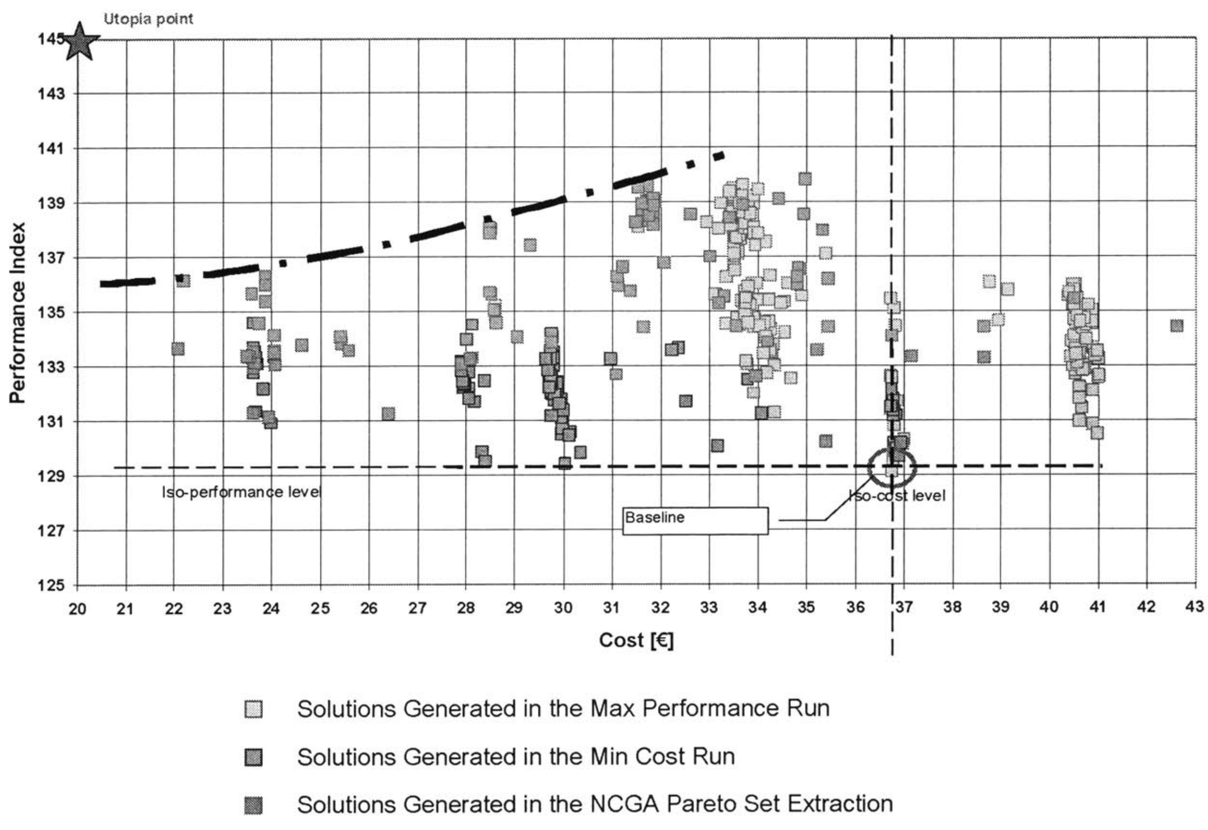


Fig. 125: Testing Phase – Performance-Cost Trade-Space

We note that a good coverage of the performance attributes space is achieved thanks to the diversity of the initial set and that some very interesting solutions were found which have higher performance than the baseline at a much lower cost.

4.4. Phase II: Running the EDF with full capabilities

With the experience gained in the testing phase, as a final step in our investigation, we simulated the development of a maniverter for the Fire 1.4L 16V with the novel Enhanced Development Framework.

For this purpose we released all the variables that are allowed by the parametric model, setting adequate upper and lower boundaries for each. This allows the optimization algorithm to consider flexible the following elements:

- Piperun centerlines
- Pipe diameters
- Position and inclination of the converter
- Thicknesses of pipes and inlet / outlet cones

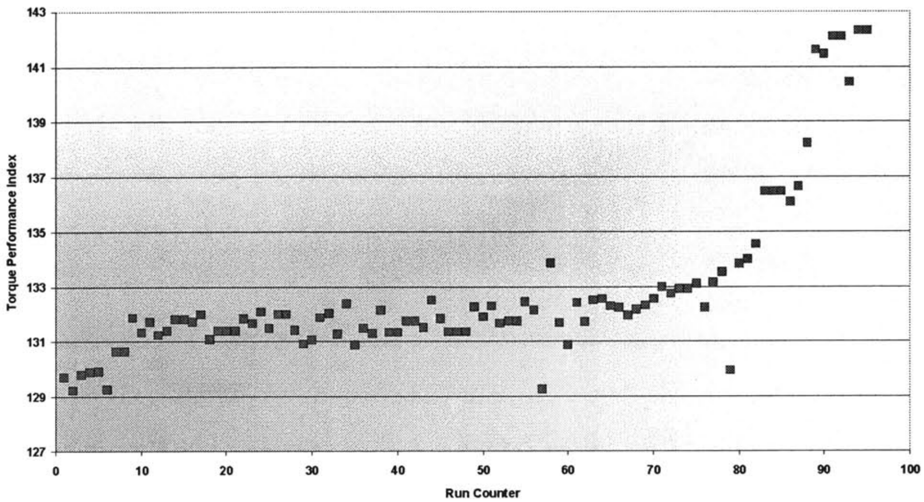


Fig. 126: Final Run – Performance Maximization – Optimization History

Following the methodology identified in the testing phase, the development run, has been articulated in two consecutive phases:

- "Anchor points" identification
- Pareto hyper-surface extraction.

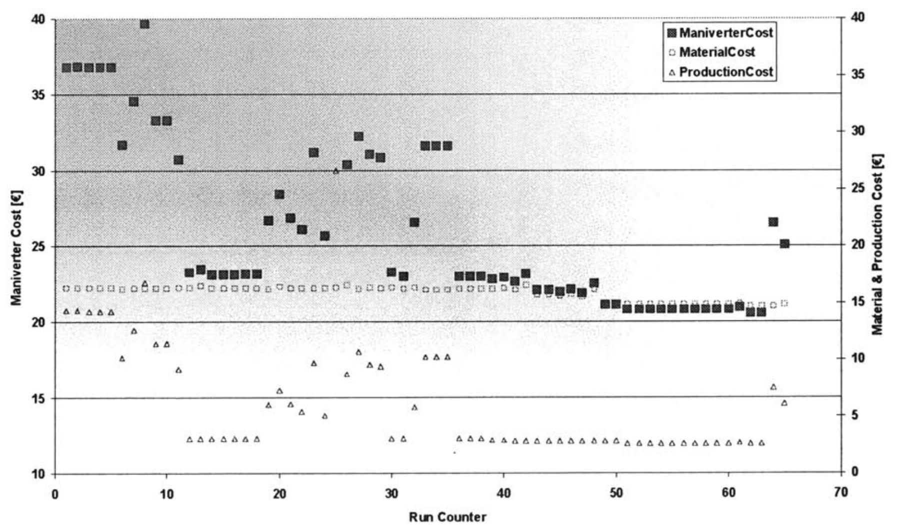


Fig. 127: Final Run – Cost Minimization – Optimization History

4.4.1. Preparation Runs

Preliminarily, feasible solutions sufficiently far apart among them and each one reasonably close to the "anchor point"[5] for the different performance attributes have been identified.

Approximately 100 runs of a single objective optimization for each of the five selected performance attributes have been done. The algorithm used has been Hooke-Jeves with a ρ=0.2 to achieve good speed.

The five single objective runs totalled 7 days of computing time.

Fig. 126 to Fig. 130 report the optimization histories. Plotted solutions are all feasible.

Compared with the runs done in the previous Section, we note that "better" solutions have been achieved in a lower number of iterations. This is due to reduction of parameter ρ in the H-J algorithm and to the higher number of degrees of freedom of the model.

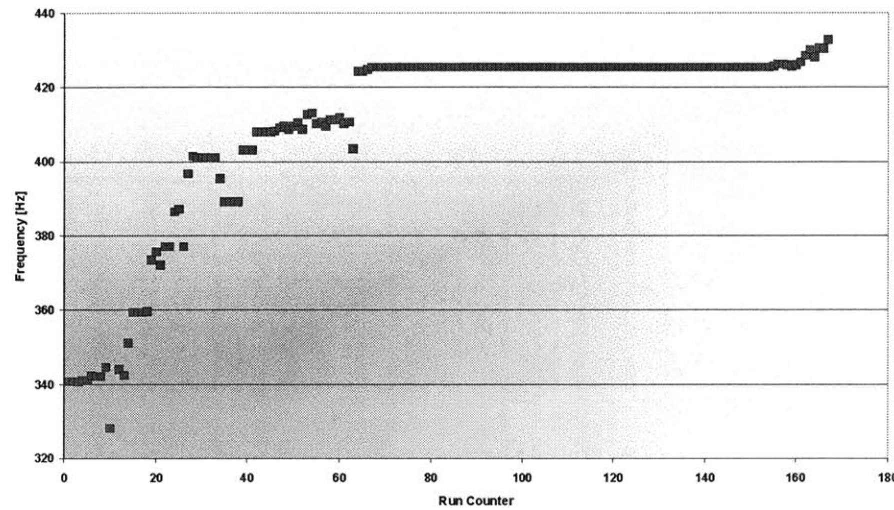


Fig. 128: Final Run – Frequency Maximization – Optimization History

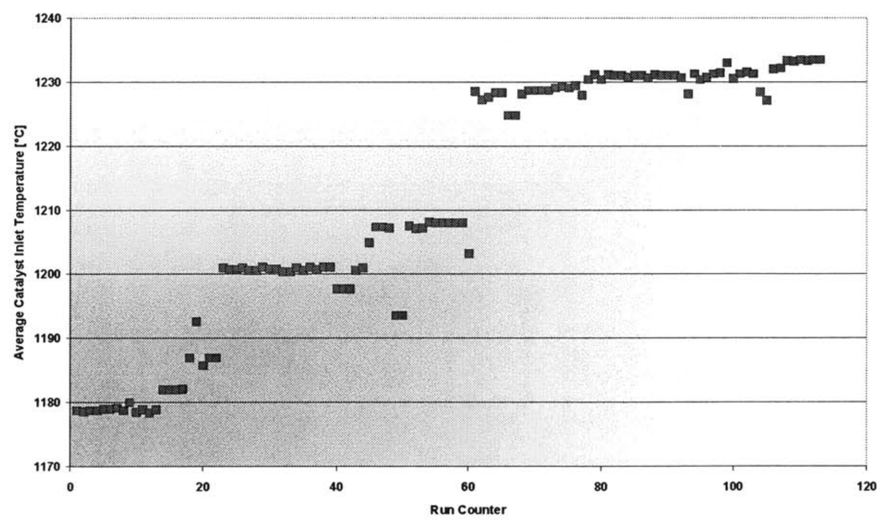


Fig. 129: Final Run – Catalyst Inlet Temperature Maximization – Optimization History

---

[5] In Multi-disciplinary Design Optimization, an anchor point is defined as a design solution characterized to have the best absolute performance attribute in one dimension.

4.4.2. Pareto Set Extraction

Four design configurations per each of the five performance attributes among the best of those identified in the exploratory runs have been selected for a total of twenty designs. These were used to compose the initial population used in the Pareto set extraction run. The NCGA algorithm was used for this purpose. The population was advanced for 25 generations for a total of 500 runs. As for the 500 preliminary runs, 7 days of computing time were required.

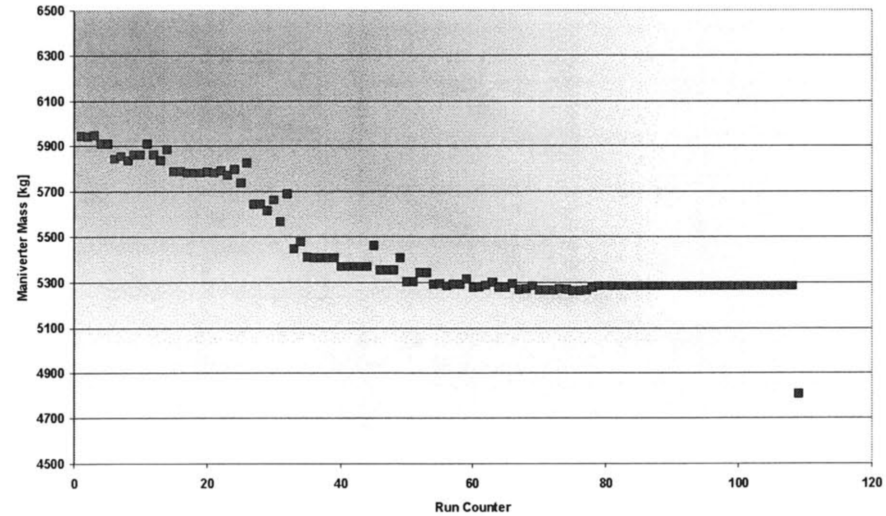


Fig. 130: Final Run – Mass Minimization – Optimization History

The initial population of feasible designs was instrumental to keep feasibility as high as 80%.

The Pareto set was generated automatically by iSIGHT's NCGA's algorithm and includes 12 designs solutions.

Fig. 131 and Fig. 132 report the different pairwise scatter plots. The red star symbolizes the utopia points in the represented dimensions.

We get the confirmation that huge variation of performance attributes in one dimension may correspond to similar performances in other dimensions. For example, for the same cost of €35, a maniverter of 4900g mass or 6200g can be designed or the same performance index of 140 can be achieved with designs which are characterized by an average catalyst inlet temperature of 1188 °C or 1234°C, i.e. nearly 50°C difference, which translates in enormous pollutant emissions levels.

However, when data have more than three dimensions, scatter plots lose their effectiveness and they must be replaced by different and more efficient forms of visualisation that aid the decision-making process.

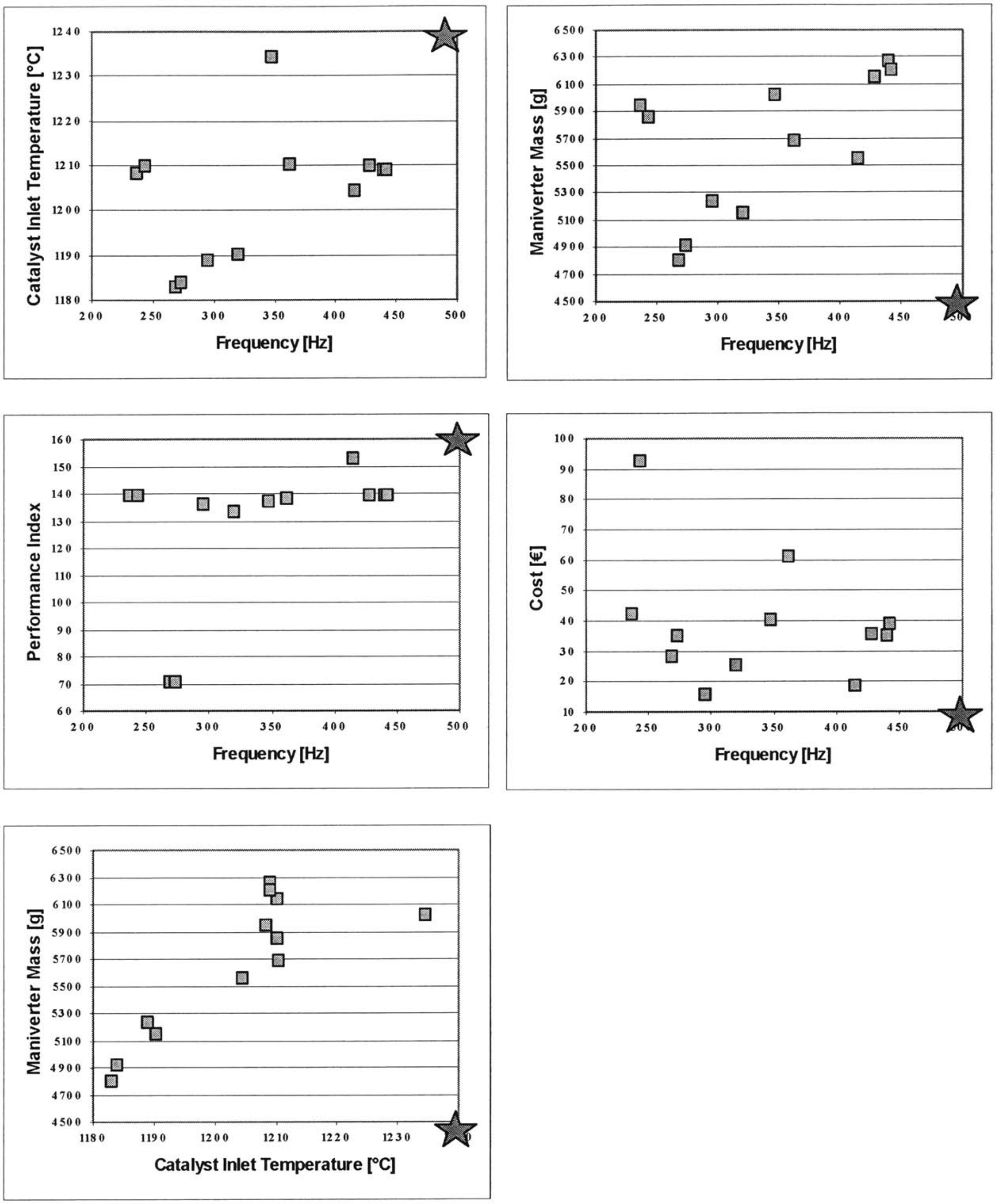


Fig. 131: Final Run – Performance Attributes Scatter Plots

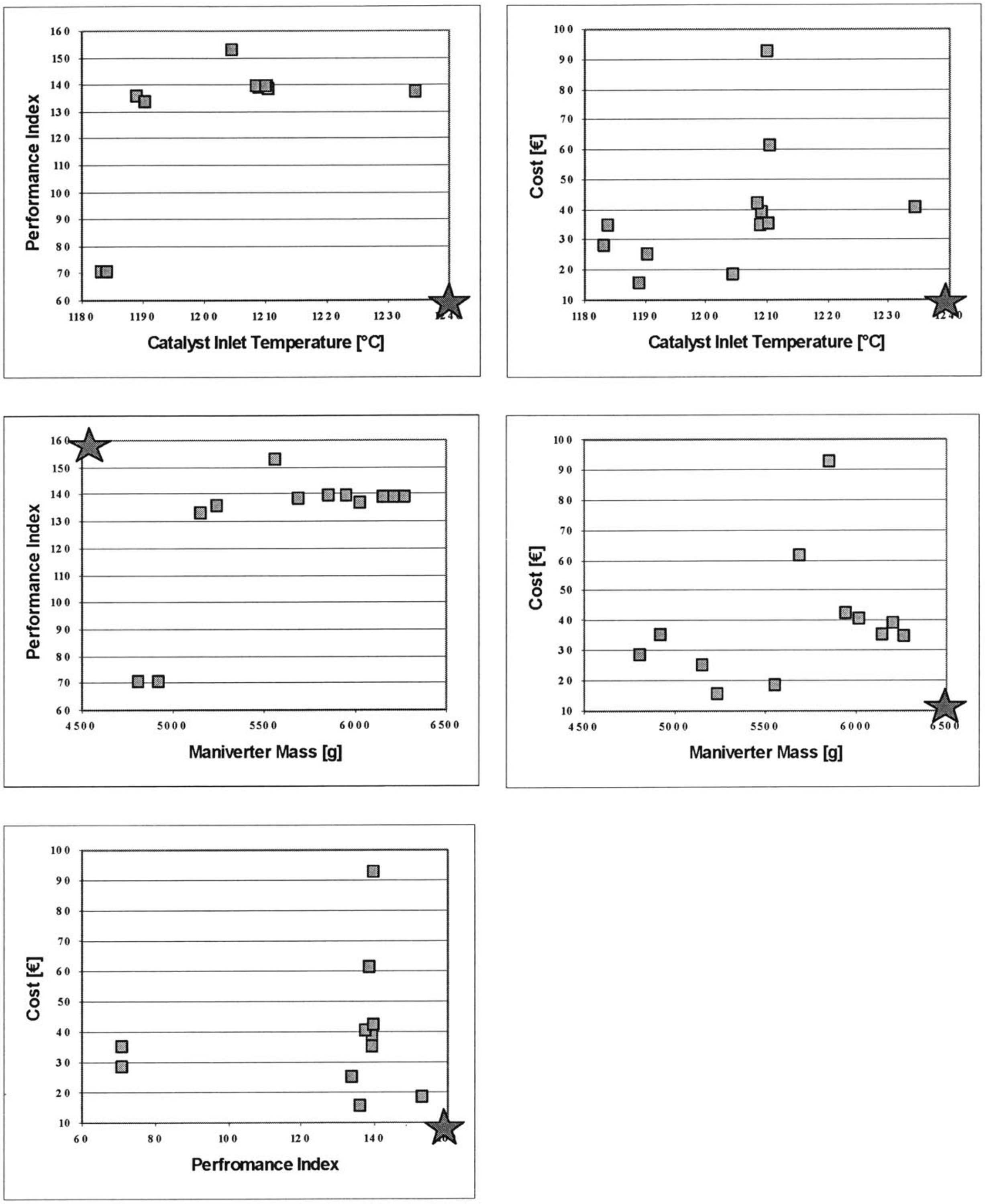


Fig. 132: Final Run – Performance Attributes Scatter Plots (cont'd)

4.4.3. The Engineering Decision

It is important to draw a critical distinction between the phase of generating Pareto solutions, which is objective, and the phase of choosing a solution from the Pareto set, which is subjective. The latter depends entirely on designer and decision-maker preference, while the former objectively seeks to generate Pareto points in the design space – regardless of their relative desirability. In the previous phase, using optimization algorithms we've extracted the Pareto optimal solution. Exploiting advanced visualization techniques, we simulated the ultimate engineering decision on the best configuration to choose.

Pareto data were processed as described in Subsection 3.11.3 and the rainbow plot generated, Fig. 133. We recall here that red color is associated with "good" performance and the blue with "bad" and that each column is a performance attribute, while each row is a particular design in the Pareto set. Solutions are ranked in maniverter cost order, since cost is deemed the most important factor in decision-making.

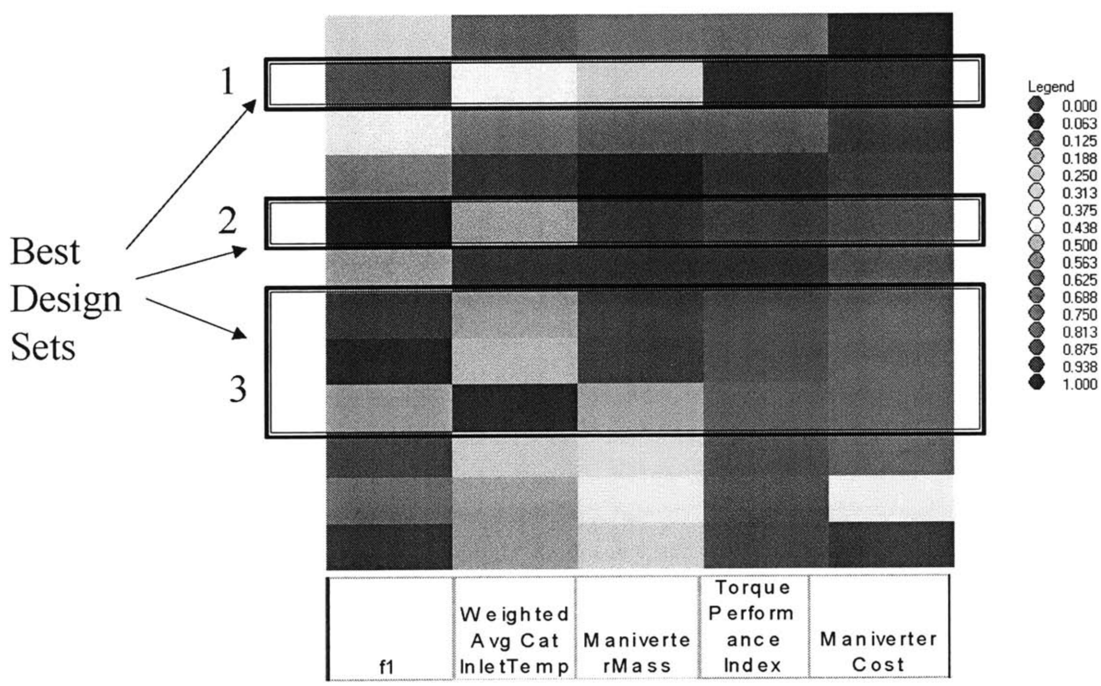


Fig. 133: Final Run – Pareto Data Qualitative Rainbow Plot

The rainbow plot conveys pictorially and intuitively several qualitative important information:

- Maniverter mass and Catalyst Inlet temperature are negatively correlated, i.e. good values (i.e. low) of the former correspond to bad (i.e. low) values of the latter. This is evident by observing that red colors in mass column are always associated to blue colors in temperature column

- Mass is positively correlated with Cost, i.e. lower mass corresponds to lower cost. This is testified by the fact that designs are associated similar colors in the cost column and in the mass column
- The inverse happens with Mass and Performance: high levels of performance attributes are generally associated with poor (i.e. high) values of mass
- Torque performance does not exhibit huge variability and it is particularly insensitive to the variation of the other performance attributes. This is the highlighted by the fact that its related column features a red-side color for most of the designs
- High levels of all performance attributes at the same time are difficult to achieve. This is witnessed by the fact that no rows with red color marked in all columns exists

We visually divided the attributes levels in "good", associated to a dark red, red, orange and yellow colors, and "bad", associated to green, light blue, blue and dark blue. In the role of decision makers, we scanned the rainbow plot to look for balanced solutions with all good performance attributes. Since no single solution exists which has all "good performances", we sub-select those which have at least four distinctively good attributes, see boxed solutions in Fig. 133. We then fine-tuned our choices by looking at the quantitative version of the rainbow plot, Fig. 134 (for a full description on how it is generated, see Section 3.11). Here the left-hand side contain the numeric values and the right-hand side the performance percentage ranking. In bold are the designs that correspond to the previously boxed solutions.

f1	Weighted Avg Cat InletTemp	Maniverter Mass	Torque Performance Index	Maniverter Cost	f1	WeightedAvg Cat InletTemp	Maniverter Mass	Torque Performance Index	Maniverter Cost
294.93	1188.86	5238.26	136.01	15.83	28.48%	11.28%	70.42%	79.41%	100.00%
415.26	**1204.38**	**5556.17**	**152.96**	**18.56**	**86.92%**	**41.48%**	**48.65%**	**100.00%**	**96.44%**
319.92	1190.33	5150.39	133.48	25.28	40.61%	14.14%	76.44%	76.34%	87.69%
268.17	1183.06	4806.43	70.79	28.49	15.47%	0.00%	100.00%	0.20%	83.52%
440.06	**1208.96**	**6266.49**	**139.27**	**35.09**	**98.97%**	**50.39%**	**0.00%**	**83.38%**	**74.92%**
273.48	1183.89	4918.21	70.62	35.27	18.06%	1.61%	92.34%	0.00%	74.69%
427.91	**1210.14**	**6146.74**	**139.31**	**35.47**	**93.07%**	**52.68%**	**8.20%**	**83.42%**	**74.43%**
442.18	**1209.00**	**6208.10**	**139.20**	**39.16**	**100.00%**	**50.47%**	**4.00%**	**83.28%**	**69.63%**
346.86	**1234.46**	**6019.93**	**137.27**	**40.64**	**53.70%**	**100.00%**	**16.89%**	**80.94%**	**67.70%**
236.31	1208.38	5947.09	139.67	42.43	0.00%	49.26%	21.88%	83.85%	65.36%
361.97	1210.38	5689.49	138.36	61.50	61.04%	53.15%	39.52%	82.26%	40.53%
243.15	1210.02	5853.49	139.45	92.64	3.32%	52.45%	28.29%	83.59%	0.00%

Fig. 134: Final Run – Pareto Data Quantitative Rainbow Plot

Our preferences set estimates as too high the mass of the second and third solution groups (>6200g). Consequently, we chose the solution from the set #1.

In Tab. 9, the identified solution is compared with the baseline design that was the starting point in our optimization work and which, as mentioned, constitutes a solution that took ArvinMeritor about 9 weeks to develop and to optimize for the particular application back in 2002.

The new solution is far better in all dimensions: it has better torque performances (+20%), better vibration characteristics (+75 Hz), better emission characteristics by ensuring faster warm-up of the catalytic converter (+25 °C) and lower mass (-389g). Last but not least, it has a remarkable 50% less cost.

Performance Attribute	Optimization Target	Baseline	Selected Solution	Difference
Torque Performance Index	⇧	129.70	152.96	17.94%
Cost [€]	⇩	36.83	€ 18.56	-€ 18.27
1st Natural Frequency [Hz]	⇧	340.71	415.26	74.55
Catalyst Inlet Temperature [°C]	⇧	1178.68	1204.38	25.70
Mass [g]	⇩	5945.63	5556.17	-389.46

Tab. 9: Performance Attributes Comparison: Optimized vs. Baseline Solution

The two different geometric configurations can be visually compared in Fig. 135. In addition, in Tab. 10, we provide a comparison of the different thicknesses:

	Pipe A [mm]	Pipe B [mm]	Pipe C [mm]	Pipe D [mm]	Inlet cone [mm]	Outlet cone [mm]
Best Solution	2.48	1.20	2.40	1.20	2.40	1.20
Baseline	2.00	2.00	2.00	2.00	2.00	2.00

Tab. 10: Thickness of different maniverter components – Baseline vs. Optimized Solution

The new design looks odd to the eye of an "experienced" designer because of the different pipe diameters and different thicknesses and in no way this would be the result of a manual development effort.

While the results of Tab. 9 would require an attentive check, particularly the cost figures which are the most striking, we note, however, a similar scenario is likely when performing automatic optimization: a high performance solution that does not correspond to well-established design pattern.

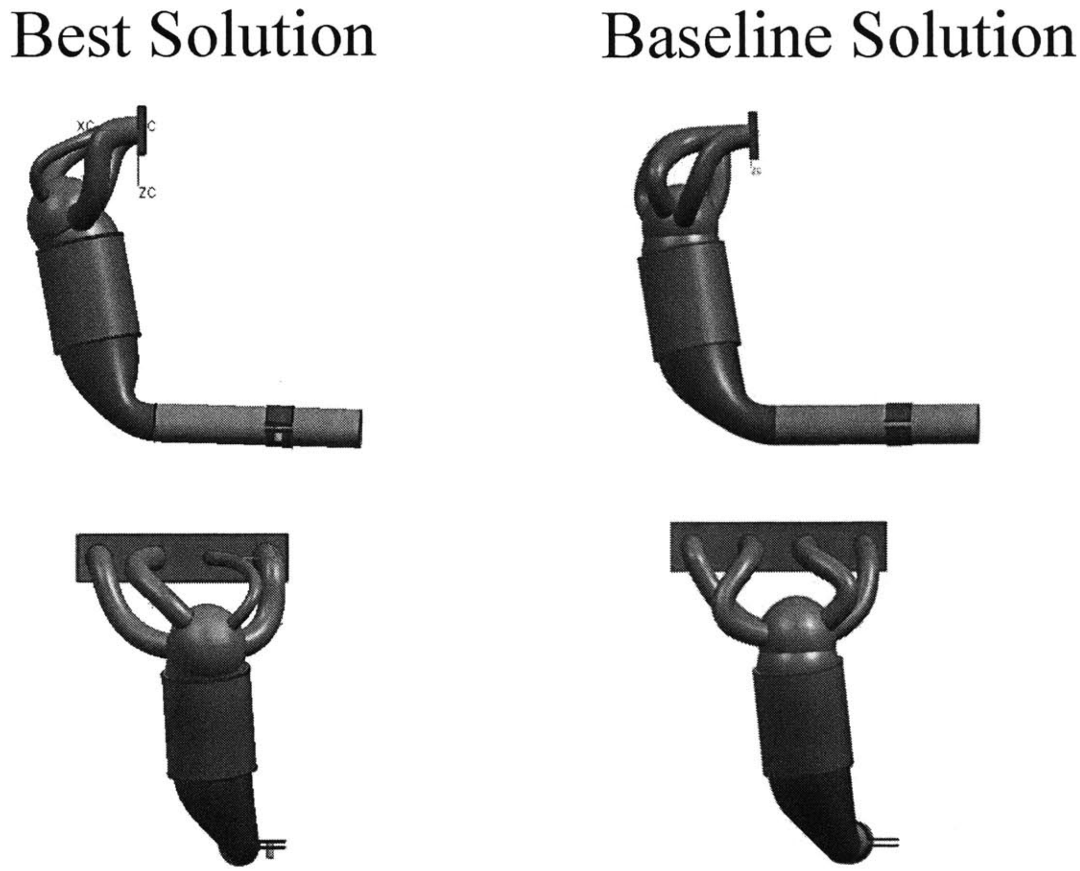


Fig. 135: Baseline (right) and Optimized (left) Maniverter Geometry

Optimization algorithms are not forced to ride the old paths of experience but are only governed solely by the goals they are given. Fast execution of design iterations enables many designs to be checked and the "sweet spots" identified. The results are as good as the underlying models.

The designer must be therefore willing to replace the natural scepticism with an authentic open mindset and be ready to accept the solution proposed by the optimization process. Sanity checks are anyway required to avoid making a mistake due to modelling errors, but when they will give confirmation on the performances of the solution, the design engineer should take the time to reflect on the reasons why the performances of the identified solution are so good. That phase is a fruitful moment of knowledge creation.

As an example of the physical explanation of some of the good performances we mention:

- The two external pipes, with their higher diameters, contribute to raise the first natural frequency, which corresponds to a lateral movement.
- The higher overall pipes cross sectional area, in addition, contribute to lower the backpressure
- The maniverter mass is lower, thus raising the frequency and lowering the cost
- The number of bends is lower and bends are with smaller values of diameter/bending radius ratios, thus lowering bending cost
- The pipework mass is lower, thus raising the average catalyst inlet temperature
- Pipe diameters are different to compensate for different lengths and bends: 1) smoothness of piperun is a factor for backpressure reduction and can be balanced against a smaller radius; 2) the same tuning frequency can be achieved with a longer pipe with small radius or with a shorter pipe of a big radius [73] .

We recognize therefore the automatic multi-disciplinary design optimization, compared with the traditional design process, has several benefits:

- Lower development time (14 days against 60+)
- Lower development cost (related to development time and resource allocation)
- Better product performances
- Innovation
- Knowledge widening

4.5. The value of the tool: summary of insights

Throughout the present project, in the build-up, testing and utilization phases, we came across several findings that, even if strictly limited to the specific application, are believed to have the potential to constitute general insights. Even though we have dispersed them where data provided evidential proof for each of them, we deemed useful to group all the insights together in a sort of body of knowledge gained throughout the project. Hereafter they are therefore listed from general design related items to more specific MDO implementation related issues.

Design:

- *Design solutions exist with similar performance along one dimensions but much different along at least one other.*
- *Design solutions with extremely good performance attribute levels in many dimensions represent a tiny subset of the design space.*
- *The majority of the generated designs are characterized by mediocre performance if compared with what those systems have the potential to deliver.* Average design practice, limited by time and budget constraints, results in poor design space exploration. Even if product experience may guide to explore good design areas, in general only 50-60% of the value that could be obtained by a system is extracted. Therefore a huge opportunity exists for both product cost reduction or product performance enhancement.
- *Relationships between performance attributes are not intuitively obvious for complex systems and not even for simple ones.* Intuition, engineering knowledge and experience usually drive the design: we use them to correlate performance attributes to design characteristics. However, for complex systems, the interrelations between physical phenomena is so intertwined that our limited cognitive capabilities may fail to find the right relationships, even in the case of relatively simple systems. A false intuition pushes the designer in wrong directions, not differently from what a mirage in a desert does to the voyager.
- *Effective designs can be found by exploiting the characteristics of the Pareto fronts.* Inter-dependence of performance attributes is, in general, not linear. Regions of the design space can exist where, by worsening slightly the performance attribute A, a huge benefit can be obtained in attributes B and/or C, etc. Moving from one area of the design space to the other, the relationship may invert, i.e. attributes B and/or C, etc. may

be insensitive even to a huge variation of the attribute A. The inflexion point can represent a zone where to search for the "best design".

- *Automatic Optimization widens design knowledge paths but requires an open mindset.* We've seen that optimization algorithms in some cases confirmed current maniverter design practices (e.g. even runners for constant torque, shortest runners for lowest mass). However, they are not constrained by "common sense" and "past experience", but they chase only numerical minima or maxima. In doing that, they are not restrained from riding new design avenues and, by doing that, they become a means for innovation. Design engineers must be open-minded, take the solution proposed by the MDO tool and find the necessary confirmation. If performances are confirmed, the innovation is real and the examination of the root causes leads to extend current product knowledge
- *Multi-disciplinary analysis shifts the engineering focus from design to performance evaluation and decision-making.* Quite often, in a design review, the question is asked, by management or customers, "what if I wanted more of this attribute?" or "what if I needed less of that attribute, can I get something in exchange?" The request invariably starts a design iteration, which consequently results in lengthening the development time. With MDA, all the design solutions are evaluated in advance and trade-offs explored at the outset. Data are presented to the design engineer / manager for him/her to take the ultimate decision.

MDO implementation:

- *Geometry generation importance cannot be overemphasized.* If a CAD tool is used, its flexibility in representing with completeness the design family and its robustness with respect to parameters variation are key to the successful execution of the design search process and to the significance of the obtained outcomes.
- *Knowledge-based design can be used proficiently to generate an adequate geometry for optimization.* Embedding design rules in the geometry generation is an efficient way of establishing the dependence relationships between the parameters. This greatly helps in making the design space more continuous and consequently in having a simpler and faster design exploration.
- *If the design space is discontinuous and characterized by channel–like feasibility zones, the Hooke-Jeeves algorithm shows good performances in single-objective optimization:* it locates ridges of a channel and follows them efficiently up to the optimum. However, it shows its weaknesses when a channel is forking: the algorithm, in fact, follows the branch of the channel which looks more promising, completely neglecting the other(s). Future research might resolve this issue.

- *Design of Experiment, Monte Carlo Design Space Exploration, and Genetic Algorithms have troubles design spaces made by feasibility channels.* Only few percent of the randomly or pre-determined generated solutions fall into the feasibility channels, the rest sink into the unfeasibility ocean and cause those algorithms to be so highly inefficient to be useless. However, Genetic algorithms with explicit Pareto optimality management, suitable for Pareto hyper-surface extraction, if properly fed with an initial population of feasible solutions, generate feasible offspring and are able to locate the Pareto front efficiently
- *Software interoperability and interfaces management is key in the success of any MDO approach.* Clear and comprehensive analysis of requirements of each analysis module must be done at the outset to ensure efficient execution.
- *In designing any engineering tool for analysis of complex systems, information processing capabilities of human users must be taken into account.* Failure to recognize the essential role of the tool/human interface may lead to develop tools that, despite their power, are perceived as too complicated and ultimately rejected by the engineering users community. That's why it's particularly important to develop adequate methods of presenting the huge amount of data coming out of the design space exploration in a way that captures the attention of decision-makers and allows using the powerful capabilities of intuition and synthesis of the human brain.
- *The automated MDO process has the potential to identify solutions with performance attributes levels much higher than with traditional manual processes at vastly lower cost and time.* Key in time saving are: 1) the resolution of the interfaces issues once for all the design iterations; 2) efficient jobs scheduling allowed by computerized queuing; 3) 24/7 activity possible only with machine operations (downtime excluded). In the maniverter example, one design iteration was accomplished in 20 min against the several days that would have required if performed manually.
- *Modular architecture for the ICE platform is to be preferred.* MDO requires different analyses to be performed. For an effective implementation, it's important that incremental building is possible: whenever a new analysis module becomes available, it must be inserted seamlessly in the platform; similarly, whenever an existing module needs to be removed or upgraded, the operation needs to be transparent for the platform. Only a modular architecture and particularly a bus architecture gives the required flexibility.

5. COST BENEFIT ANALYSIS AND IMPLEMENTATION

> *"When you can measure what you are speaking about, and express it in numbers, you know something about it; but when you cannot measure it, when you cannot express it in numbers, your knowledge is of a meager and unsatisfactory kind."*
>
> Lord Kelvin

5.1. Introduction

In Chapters 3 and 4, we have described a prototype of an enhanced development tool and tried to apply it to maniverter development. Hopefully we've succeeded in showing some of the benefits that the application of this tool can offer to product development.

The framework developed so far, however, is rather simple and limited in the product variants that it can handle and in the accuracy of performance attributes prediction. To transform the tool from the current status to a sufficiently sophisticated level so that it could be used to transform the current paradigm of maniverter design, a lot of further development is needed.

The current Section is attempting to outline a "research" project that could fulfil such a goal, i.e. a process through which, building upon the knowledge gained in this Thesis project, a tool for maniverter development that could be used in a real setting of a for-profit-company like ArvinMeritor is created. The intent is also to show that such a project is characterized by a strong business case.

The planning is presented first that sets the roadmap and its timeframe; in addition, the resource requirement (including the additional hardware and software resources) is elaborated to quantify the related expenditure.

A vision of the new product development process enabled by the EDF is then projected and the benefits highlighted qualitatively and quantitatively.

The total costs are finally balanced against the expected benefits showing that the reward is undoubtedly worth the effort[6]. Needless to say that the result is strongly dependent on the many assumptions that were made. Proper business analysis would require an attentive assessment of each of them.

Lastly, the strategy for the deployment of such an advanced development tool is envisioned together with the consequent organizational changes that are expected to be required for the company to reap its full benefits.

[6] In all economic calculations a standard conversion rate of 1.35 USD for 1€ is assumed.

5.2. Development Plan

As mentioned in Section 3.4, in order to build a manageable platform within the available hardware and time constraints, several simplifying assumptions were made. While they helped to reduce significantly the complexity, they also limited the power of the resulting tool.

In looking ahead to a framework to be used in a real business setting, all the main limitations need to be removed. The following enhancements are deemed essential:

- CAD module.
  - As mentioned in Section 3.6, this part of the platform is of paramount importance; consequently a significant portion of the project effort should be spent on this module. The possibility to cope with engines with different number of cylinders and to handle different manifold technologies (clamshell and airgapped in addition to the tubular one) as well as different manifold topologies needs to be introduced together with adequate clearances management and variable geometries piperun. In addition, the new EDF should allow the re-use of legacy components and include already some manufacturing constraints. Knowledge-Based Engineering (KBE) is expected to be instrumental in creating a geometry module which is able to include the design and manufacturing rules in the CAD model. Arguably, KBE will overcome the geometry regeneration issues and therefore enable the effective use of Genetic Algorithms. However, skill and expertise availability are expected to be a major hurdle. In addition, the complexity of the CAD model is projected to increase and, with that, the number and type (continuous, logical, discrete) of design variables.
- Structural module:
  - Natural frequency analysis. A more refined model is required to improve accuracy in prediction, for example a Tet10 element mesh (or equivalent) or finer. Critical analysis of which type of elements is best suited is essential. In addition, methods that allow straightforward automatic meshing in the presence of a great degree of geometry variability need to be developed. The selection of the proper CAD/CAE data translation method is key.
  - Thermal-induced stresses analysis. This section needs to be completely developed. Strategic synergy with the natural frequency meshing needs to be investigated. Adaptive meshing is to be included in order to have an accurate stress prediction while keeping the model relatively small in size and complexity. Methods of creating and applying correct temperature pattern boundary conditions are essential. Performance attributes goals needs to be

worked out, linked to durability / reliability targets required by car manufacturers.

  - Vibration-induced stresses. Issues similar to the thermal-induced stresses hold. Proper specification of engine vibration characteristics and adequate methods of applying them as boundary conditions are key. Acceptance criteria need also to be set.

- Fluid Dynamic module:
  - CFD analysis for flow distribution prediction on the catalyst. This section needs to be completely developed. As for the structural module, proper geometry transfer methods need to be worked out as well as adequate meshing algorithms.
  - Engine performance prediction. Differently from the other models, since the model for engine performance calculation is 1-D, the challenge is to find a convenient translation mechanism (from the 3D CAD to 1-D data) that is able to cope with the extreme variability of the geometry. At the same time, the accuracy of the model with different geometry configurations needs to be assessed.
- Cost Module. The cost model needs to be improved, the tooling costs added and its error margins assessed. As for the engine performance model, a reliable geometrical data extraction method from the 3D CAD model needs to be developed. Getting the support of the Company costing and manufacturing engineers is expected to be challenging. Since the activity of building a cost model has the purpose to give a reasonable prediction of the part cost, it is expected to interfere with current cost calculations efforts. Cost engineers are also expected to be reluctant to share their tacit knowledge.
- Optimization strategy. Careful analysis of the design space and of the optimum combination of optimization algorithms needs to be investigated.
- User Interface. An interface that allows a direct and friendly exchange of inputs and outputs with the user needs to be further developed.

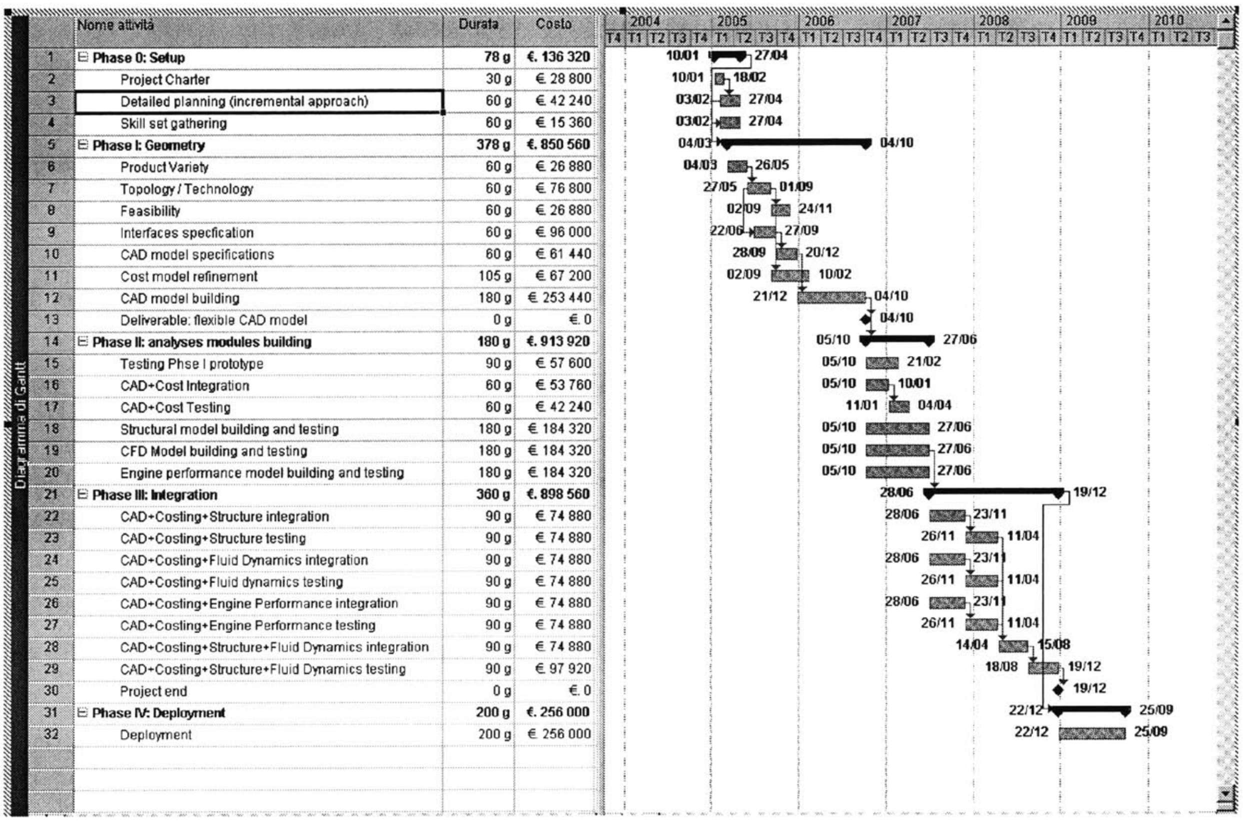
These tasks are collected in a provisional timing plan that is presented in Fig. 136. The development effort is planned to start with an initial setup phase where the definition of the partner companies and the necessary agreements are made, a skill set requirement analysis is done and the composition of the working team is established (including future users). After this Setup, the project is executed articulated in four main phases:

- Phase I: Geometry. In this Phase, the foundations of the platform are laid.

- The fundamental architecture is set up. Similarly to what was done in the thesis project, a comprehensive analysis of product variants that the architecture should be able to reproduce and the product conceptualization are done.
- Interfaces of the geometry with the main modules and with the optimizer are carefully analyzed and communication protocols outlined.
- Maintenance strategy needs also to be worked out in this early stage: clear distinction needs to be made between the parts that will be updated periodically and frequently and what is expected to need major redevelopment. Plug and play capabilities should be embedded in the architecture to cope with new visions that will emerge and a constantly changing CAD/CAE/CAM environment.
- The cost model, which is probably the simplest and at the same time, the most delicate module can be already reviewed and improved even if no CAD model exists yet. The challenge is expected to be creating a cost module able to cope with different technologies and having a reasonably low error margin.

- <u>Phase II: Analyses modules building</u>. This is the most intense and expensive phase. The CAD model and the cost model created in the previous phase are evaluated, all the building blocks are built and the platform prepared for the integration. More specifically:
  - Simple tests, which can be carried on with the CAD model only, are performed: design space exploration for feasibility, mass minimization, clearances management, etc.
  - The costing module is integrated in the platform and two objective analyses can be done (i.e. cost and mass). Production costs and material costs trade-offs can be evaluated.
  - Structural, CFD and engine performance modules are built.
- <u>Phase III: integration</u>
  - Following an incremental approach, modules are integrated one by one and the coupling tested during this phase.
  - While doing the testing, different optimization and design space exploration algorithms are also screened.
  - Approximation methods are surveyed and their capabilities assessed.
  - Automatic Pareto data extraction and post-processing are also enhanced for the most efficient analysis of the results.
- <u>Phase IV: deployment</u>

- The tool is now complete and it can be applied to existing projects in a difficult situation from the technical, timing or cost standpoint to verify its effectiveness. It can also be applied to coming RFQs.
- Design engineers, members of the development team, will start using the tool. They will also train a selected core team of the design engineers who are candidate to become Product Managers (see also Subsection 5.7.1)
- The results of "field-testing" are used to fine-tune the tool and to list its major weaknesses for the next release.



	Nome attività	Durata	Costo
1	⊟ **Phase 0: Setup**	**78 g**	**€. 136 320**
2	Project Charter	30 g	€ 28 800
3	Detailed planning (incremental approach)	60 g	€ 42 240
4	Skill set gathering	60 g	€ 15 360
5	⊟ **Phase I: Geometry**	**378 g**	**€. 850 560**
6	Product Variety	60 g	€ 26 880
7	Topology / Technology	60 g	€ 76 800
8	Feasibility	60 g	€ 26 880
9	Interfaces specfication	60 g	€ 96 000
10	CAD model specifications	60 g	€ 61 440
11	Cost model refinement	105 g	€ 67 200
12	CAD model building	180 g	€ 253 440
13	Deliverable: flexible CAD model	0 g	€ 0
14	⊟ **Phase II: analyses modules building**	**180 g**	**€. 913 920**
15	Testing Phse I prototype	90 g	€ 57 600
16	CAD+Cost Integration	60 g	€ 53 760
17	CAD+Cost Testing	60 g	€ 42 240
18	Structural model building and testing	180 g	€ 184 320
19	CFD Model building and testing	180 g	€ 184 320
20	Engine performance model building and testing	180 g	€ 184 320
21	⊟ **Phase III: Integration**	**360 g**	**€. 898 560**
22	CAD+Costing+Structure integration	90 g	€ 74 880
23	CAD+Costing+Structure testing	90 g	€ 74 880
24	CAD+Costing+Fluid Dynamics integration	90 g	€ 74 880
25	CAD+Costing+Fluid dynamics testing	90 g	€ 74 880
26	CAD+Costing+Engine Performance integration	90 g	€ 74 880
27	CAD+Costing+Engine Performance testing	90 g	€ 74 880
28	CAD+Costing+Structure+Fluid Dynamics integration	90 g	€ 74 880
29	CAD+Costing+Structure+Fluid Dynamics testing	90 g	€ 97 920
30	Project end	0 g	€ 0
31	⊟ **Phase IV: Deployment**	**200 g**	**€. 256 000**
32	Deployment	200 g	€ 256 000

Fig. 136: Timing Plan for the Maniverter EDF development

The effort is planned to have a 4 years development + 1 year deployment. Estimation of the amount of resources needed is outlined in Fig. 136.

The development team is planned to be conceptually composed by two groups:

- System Integrators: they will lead the project, set the directions, take critical decisions and have always in control the whole activity. ArvinMeritor main role is within this team. If adequate methods to preserve confidentiality could be put in place, it would also beneficial to include members of advanced engineering departments of the major car manufacturers.

- Platform Contributors: they are mainly those subjects responsible of the individual modules (CAD, CAE, Optimizer). This team is made both by members of partner companies and by members of the ArvinMeritor CAD/CAE/Costing/Manufacturing teams. The former will bring the expertise on the particular software packages, the latter will bring the specific product knowledge. The two components will be fused and complemented by System Integrators.

5.3. Hardware Requirements

Increasing the complexity of the software models has the risk of transforming the platform in such a "heavy" environment that no computer is powerful enough to be able to handle it. In what follows, a projection is made about the final computational requirement with the purpose of assessing whether this will be compatible with an affordable hardware currently or shortly available.

To make this projection, an estimation of the required computing power by the different analyses is made (see Tab. 11). The order of magnitude of the Floating Point Operations per second of the different analyses is estimated based on the experience gained during this project and some guidelines provided by the literature [74] . The calculation is done first for the current prototype to calibrate the estimation and then scaled to the projected final definition level.

As we can see, with the current demo tool, performing the 1000 iterations necessary for a complete Pareto front extraction takes about 14 days of runtime on the selected Pentium 4 laptop (whose results are shown in 4.4).

Type of calculation		Current model on Pentium 4 - 1.8GHz (3 Gflops)					Projected model on Pentium 4 - 1.8GHz (3 Gflops)					Projected model on a 10 Teraflops machine		
		Estimated FLOPS per iteration	Note	Time per iteration [min]	No. Of Iterations	Total time [days]	Estimated FLOPS per iteration	Note	Time per iteration [min]	No. Of Iterations	Total time (days)	Time per iteration [min]	No. Of Iterations	Total time [days]
Structural analysis	Natural frequencies	5.58E+11	8000 grid points per 6 degrees of freedom	3.10		2.15	1.36E+14	50,000 grid points per 6 degrees of freedom	757.50		5260.42	2.27E-01		1.58
	Thermal-induced stresses		N/A				1.36E+14		757.50		5260.42	2.27E-01		1.58
	Vibration induced stresses		N/A				1.36E+14		757.50		5260.42	2.27E-01		1.58
Fluid Dynamics	Flow Distribution (steady state)		N/A		1000		9.80E+13	1,000,000 grid points per 49000 floating point operations per 500 iterations to convergence	544.44	10000	3780.86	1.63E-01	10000	1.13
	Engine Performance	3.00E+12	6 rpms per 10 cycles to convergence per 50 Tflop per rpm per cycle (based on recorded execution time)	16.67		11.57	6.00E+13		333.33		231.48	1.00E-01		0.69
CAD	CAD Operations	1.50E+11		0.83		0.58	1.50E+12		8.33		5.79	2.50E-03		0.02
	Total	3.71E+12		20.60		14.31	5.69E+14		3158.61		19799.38	9.48E-01		6.58

Tab. 11: Estimated FLOPs required by the current prototype and future EDF

The increase in complexity required for a real business development translates in a significantly augmented computational effort:

- Structural assessment requires more accurate meshes and more complex analyses
- CFD is added to the analyses suite
- Engine performance is executed in a refined mode: from 6 rpms (from 1000 to 6000 rpm at 1000 rpm intervals), to 12 rpms (reducing the intervals to 500 rpms)
- Optimization process: with an increase in the number of design variables, a considerable increase in the number of iterations required to extract a Pareto hyper-surface is expected if reasonable completeness is to be kept (from 1,000 to 10,000).

All this multiplies by a factor of 1,000 the number of total FLOPs required for a single maniverter development run.

If the calculations were to be performed on the current laptop (which is estimated to be capable of sustained 3 GFLOPS), each development run would require 19,800 days, i.e. more than 54 years! The use of approximation methods could greatly improve this performance, but it's self-evident that a convenient powerful hardware platform needs to be planned. A 10 TFLOPS machine (more than 3000 more powerful than the current laptop) is estimated to be suitable for the purpose since it's projected to perform the required calculation in 7 days, i.e. one working week (considering that the computer will run also Saturdays and Sundays).

If today, in 2004, the cost of a 10 TFLOPs machine is high (about $1M), conservative estimates forecast it to fall to $200,000 in 2007, to $100,000 in 2008, to $70,000 in 2009 and 30,000 in 2010 (see Fig. 137, drawn from Fig. 34).

So, by the time the enhanced development tool is expected to be ready, the hardware cost should not be a hurdle.

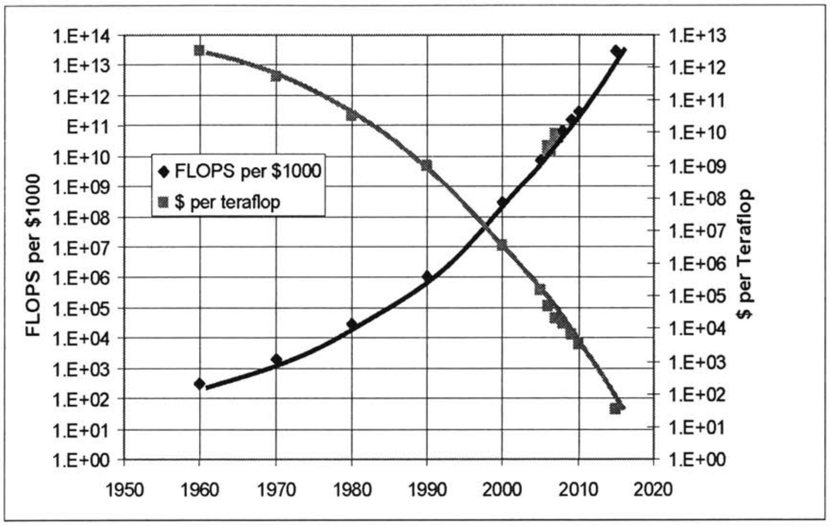


Fig. 137: Projected Cost per TFLOPS

5.4. The envisioned change in the product development process

The Enhanced Development Framework is expected to change radically the way a company like ArvinMeritor designs products and does business. In Fig. 138, a typical virtual product development process is outlined and compared with what is expected to be the new process enabled by the EDF.

Currently, an average of three development loops are required before Design Freeze. The development starts with a CAD model made by an experienced designer that fits in the available space in the engine compartment. The model is usually handed over to the CAE Dept. that performs the basic structural analyses, namely the natural frequency and thermal stress. Usually these analyses highlight either that stiffening is required or that some areas of high stress exist. Therefore, a CAD rework is usually necessary. After that, the fluid dynamic analysis is done to evaluate the design from the flow distribution and the engine performance standpoints. In all these analyses, 70% of the time is actually manual operations (data translation and meshing) and only 30% is computing time. More often than not, in this first phase which lasts approximately three to four months, despite the experience of the designer, the flow uniformity and the engine performances are not on target and a significant rework needs to be done. Usually the OEM also provides some inputs for the next iteration loop.

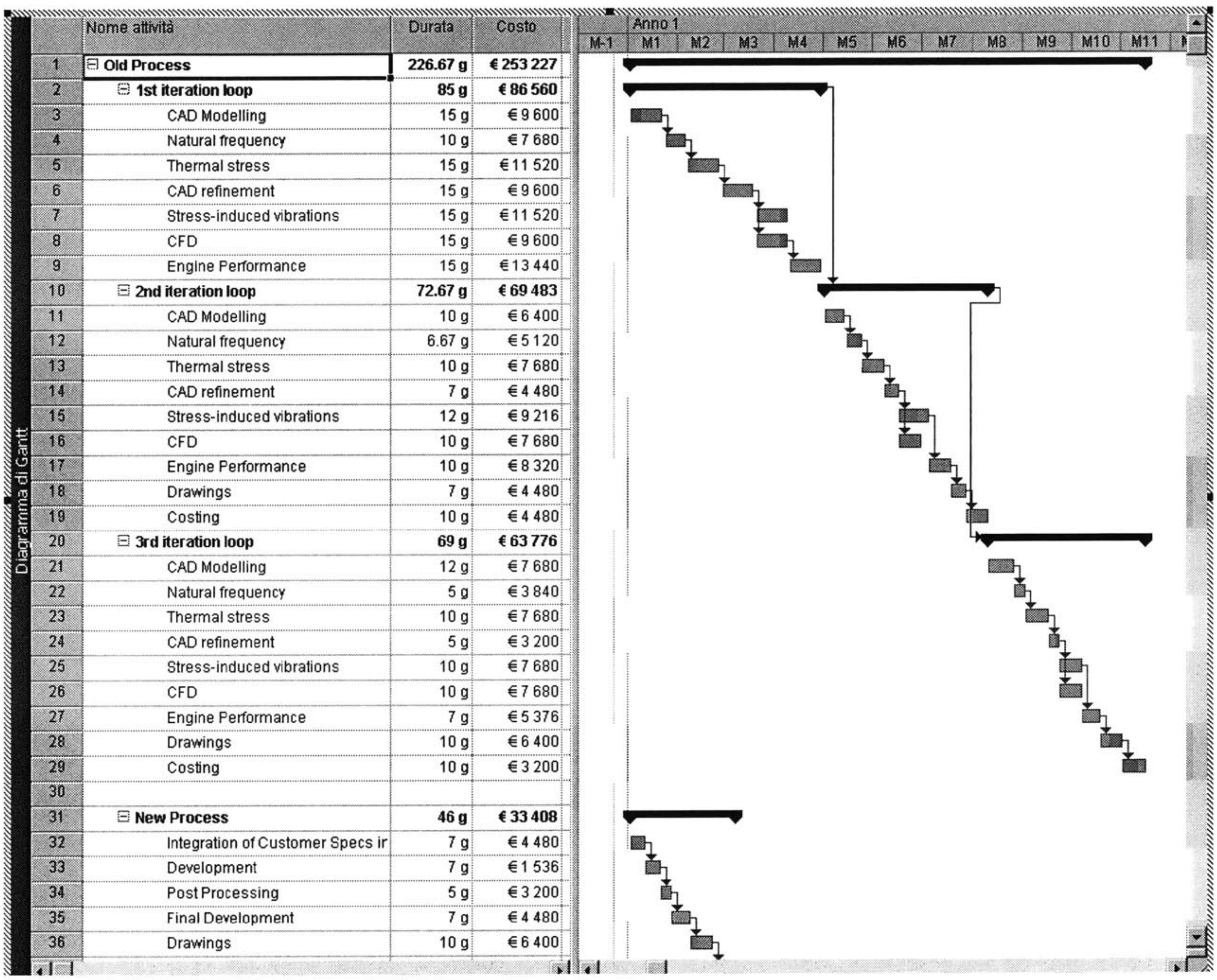
The second iteration loop proceeds similarly to the first. A new CAD model is made which includes the fluid dynamics inputs from previous analyses. Since the design is usually significantly different from the previous one, structural analyses are also repeated. At the end of the loop, even if performance attributes are not completely aligned to the targets received in the OEM's product specification sheet, the design is considered at a sufficiently high definition level that 2D drawings are made and costing follows to get a preliminary feeling for product cost. The design is also presented to the Customer for them to check the results of the design effort and to evaluate the performance level reached up to that point. By now, seven-eight months have passed since the beginning of the development. At this time, the Customer has a clearer picture of vehicle / engine requirements and quite often they are able to express preferences for performance attributes, setting priorities for the next development phase. Usually prototypes are built – sometimes called alpha prototypes – for ArvinMeritor and the car manufacturer to test the design and, therefore, to get an experimental confirmation of the performances predicted with the analysis models (proto building and testing phases are not represented in the timing plan).

The third loop is intended to fine tune the design and reach the definitive performance attribute levels according to the preferences expressed by the Customer; cost and durability are usually the major drivers in this phase. After a final assessment with the Customer, a second round of prototypes – sometimes called beta prototypes – is built. In addition to the confirmation of the final functional performance exhibited by the design, durability is also experimentally verified. Some years ago this phase was considerably painful and costly because durability failures were often encountered and major re-designs needed with a repetition of a full design loop. Nowadays, with the improvement of prediction codes, while not flawless, this phase is considerably smoother.

The total development lasts about 10 months. However two factors invariably enter and considerably lengthen the total development time (and money spent): resource allocation and customer driven changes.

ArvinMeritor, as most of the automotive companies, has resources allocated for more than 100% of their available time. In this resource-constrained environment, only careful organizational management allows ensuring that the scheduling of the analyses and design activities required by the different projects is respected. However, "emergency" situations and unexpected rework on one or more projects drain resources and create bottlenecks which induce delays on other projects. This has, consequently, the effect of inducing gaps in the activities: an average of 20% leadtime increase is expected. The development time, therefore, realistically totals approximately 1 year.

While ArvinMeritor is developing the maniverter system, the OEM develops the vehicle and or the engine. They experience the same issues of design iterations and scheduling delays of any product development. In addition, OEM marketing inputs can change over time because user tastes shift or because management sets a different market penetration strategy. As a result, invariably at least once in the maniverter development timeframe, the customer asks for a major modification of the design to address the new set of issues. This implies for ArvinMeritor to scrap part of the work done and to re-execute a certain portion of the development. A coarse estimation indicates that 50% rework is a sensible figure in this case. Fortunately, in most cases, the OEM is willing to pay for the extra effort, but the net result is that a typical maniverter development leadtime settles at approximately 18 months with 15 months of effective virtual development work.



	Nome attività	Durata	Costo
1	⊟ **Old Process**	**226.67 g**	**€ 253 227**
2	⊟ **1st iteration loop**	**85 g**	**€ 86 560**
3	CAD Modelling	15 g	€ 9 600
4	Natural frequency	10 g	€ 7 680
5	Thermal stress	15 g	€ 11 520
6	CAD refinement	15 g	€ 9 600
7	Stress-induced vibrations	15 g	€ 11 520
8	CFD	15 g	€ 9 600
9	Engine Performance	15 g	€ 13 440
10	⊟ **2nd iteration loop**	**72.67 g**	**€ 69 483**
11	CAD Modelling	10 g	€ 6 400
12	Natural frequency	6.67 g	€ 5 120
13	Thermal stress	10 g	€ 7 680
14	CAD refinement	7 g	€ 4 480
15	Stress-induced vibrations	12 g	€ 9 216
16	CFD	10 g	€ 7 680
17	Engine Performance	10 g	€ 8 320
18	Drawings	7 g	€ 4 480
19	Costing	10 g	€ 4 480
20	⊟ **3rd iteration loop**	**69 g**	**€ 63 776**
21	CAD Modelling	12 g	€ 7 680
22	Natural frequency	5 g	€ 3 840
23	Thermal stress	10 g	€ 7 680
24	CAD refinement	5 g	€ 3 200
25	Stress-induced vibrations	10 g	€ 7 680
26	CFD	10 g	€ 7 680
27	Engine Performance	7 g	€ 5 376
28	Drawings	10 g	€ 6 400
29	Costing	10 g	€ 3 200
30			
31	⊟ **New Process**	**46 g**	**€ 33 408**
32	Integration of Customer Specs ir	7 g	€ 4 480
33	Development	7 g	€ 1 536
34	Post Processing	5 g	€ 3 200
35	Final Development	7 g	€ 4 480
36	Drawings	10 g	€ 6 400

Fig. 138: Current Maniverter Development Process and the New EDF enabled

The new paradigm of product development radically changes this scenario.

Supposing that the EDF is fully deployed and the users have already gone through the initial learning phase, one week is forecasted to be taken by design engineers (later on called Product Managers) to introduce all customer and company requirements (engine performance, cost, flow distribution, tooling access, etc), constraints and input data (e.g. Engine compartment CAD file, GT-Power models, etc) in the framework. Then 7 days of execution time are taken to find out the Pareto optimal solutions. One working week is scheduled to post-process the results and format them for the final decision making together with the customer.

Once the design is selected, about three weeks of CAD activity are still considered to bring the CAD model delivered by the framework to the final detailed definition level and for making the necessary (still!) 2-D drawings for costing purposes. Then detailed costing is performed and from that moment on the prototype building phase can start. CAD detailing is supposed not to alter product performance and the cost model in the EDF is supposed to be sufficiently accurate. These two assumptions are important to have a design solution that does not necessitate of additional design iterations.

The total development leadime is now reduced to 2 months, compared to the previous 10, with development costs reduced accordingly to about 1/5. If the customer changes the design requirements, no re-development is needed, since all the solutions are extracted in one shot. Only the model refinement and the drawings are necessary.

We anticipate, however, that if a Tier 1 supplier is capable of such a fast and inexpensive development process, the OEM is induced to make changes more often; in addition changes in the layout or in the input data are still possible[7]. To take this into account, a doubling of the development time compared with what a single development would take is planned, i.e. 4 months, still way shorter if compared with the 15 months of the current status.

Development costs reduce in a similar proportion. A summary and comparison of development project financials is presented in Tab. 12.

	Current Process			New Process			Difference		
	Activity Duration [months]	Cost [k€]	FTE	Activity Duration [months]	Cost [k€]	FTE	Activity Duration [months]	Cost [k€]	FTE
1st development	10	250	2.74	2	33	1.81	8	217	0.93
impact of customer change	5	125	2.74	2	33	1.81	3	92	0.93
Total	**15**	**375**	**2.74**	**4**	**66**	**1.81**	**11**	**309**	**0.93**

Tab. 12: Development Project Costs (current and EDF enabled)

[7] We can also envision that customer data within ArvinMeritor are synchronized in real time with those at the OEM's site using engineering collaboration tools.

In the same table the number of people continuously engaged in the project is also reported expressed in Full Time Equivalent (FTE) units.[8]

The bottom line is that the use of the proposed Enhanced Development Framework process has the potential to reduce the maniverter development costs by more than €300,000 ($400,000+) per project and to free nearly one resource on a project basis and more than 2 on a yearly basis (project duration, in fact, is 4 months compared with the 15 months of the standard project).

5.5. Cost Benefit Analysis

Building upon the arguments discussed in previous Sections, hereafter follows a preliminary Cost Benefit analysis of the EDF.

Cost-Benefit Analysis (CBA) estimates and totals up the equivalent money value of the benefits and costs of a project to establish whether it is worthwhile. The purpose is to assess whether the development of such a tool is economically sound or not.

In case of our EDF development project, costs are expected to be related to:

- Hardware
- Additional software
- Tool development, implementation and maintenance

Direct Benefits are expected to be:

- Reduced development costs
- Higher margins due to a more cost effective development of products
- Increased market share and turnover due to the higher flexibility, shorter development time and lower development costs

In what follows a brief discussion of each element is done. At the end of the discussion the analysis is presented.

Costs

- Hardware Costs:
    - During the entire development project leverage of existing hardware is planned. If extra CPU power is required, it's planned to be purchased for the number of computing minutes required for a run; this is already taken into account in the project budget. It's only in 2009, when the platform is deployed that the

[8] FTE is a measure of staff hours equal to those of a full-time employee working 80 hours per pay period over the course of a fiscal year. Source: ohr.gsfc.nasa.gov/wfstatistics/definitions.htm

forecasted 10 TFlops machine is really required. By that time the cost of such a computer is estimated to be $70,000 or less. The utilization of the machine for maniverter development, at the beginning, is expected to be only 5% and, even at full regime, not to exceed 50%. For this reason, several other conventional analyses are forecasted to be executed on the same machine. Resulting savings on otherwise purchased desktop workstations may offset the cost of the new supercomputer. On the other side, a detailed study of current workstations utilization may also highlight that setting up a computing grid using existing hardware can make the purchase of a new supercomputer unnecessary. Since, in any case, savings are expected to balance costs, hardware costs are not charged to the project in the present CBA.

- Software Costs:
  - No additional CAE license fee is required as current available licenses are deemed to be sufficient to cope with existing and even increasing workload. The fast clockspeed of the EDF will maximize the efficiency in licenses usage.
  - CAD licenses: KBE additional licenses are expected to be required. One additional development toolkit plus 5 additional client licenses are estimated to amount 37,000 $ plus $1,600 for yearly maintenance.
  - Optimizer software: this is a new component of the software suite and needs to be taken into account. Annual fee is estimated to be $13,000.
- Development Cost and implementation:
  - The resource requirement associated with the development plan presented in Fig. 136 entails a total of about €3M or $4M. Assuming that the partners of the development be the same of the present work, the following breakdown can be outlined (Tab. 13):

	2005	2006	2007	2008	2009	Total I [€]	Total [$]
CRF	€ 230 656	€ 361 664	€ 353 792	€ 313 600		€ 1 259 712	$1 700 611
ArvinMeritor	€ 256 512	€ 137 472	€ 201 408	€ 156 160	€ 245 760	€ 997 312	$1 346 371
MSC.Software	€ 31 232	€ 58 816	€ 109 376	€ 16 000		€ 215 424	$290 822
AVL	€ 31 232	€ 69 760	€ 132 992	€ 16 000		€ 249 984	$337 478
Engineous	€ 31 232	€ 22 336	€ 37 696	€ 32 000		€ 123 264	$166 406
CD	€ 25 472	€ 58 816	€ 109 376	€ 16 000		€ 209 664	$283 046
Total	€ 606 336	€ 708 864	€ 944 640	€ 549 760	€ 245 760	**€ 3 055 360**	**$4 124 736**
Total spent (with 50% European funding) [€]	€ 303 168	€ 354 432	€ 472 320	€ 274 880	€ 122 880	**€ 1 527 680**	
Total spent (with 50% European funding) [$]	**$409 277**	**$478 483**	**$637 632**	**$371 088**	**$165 888**	**$2 062 368**	

Tab. 13: EDF Development Costs

- The scenario of getting 50% of development costs funding from the European Community is included. Funding is planned to be obtained by participating to the The European Union 6th Framework research Programme 2002-2006 (http://europa.eu.int/comm/research/fp6/index_en.html). The project is planned to be inserted in the research theme for Sustainable Surface Transport, which calls for projects on "*Advanced design & production techniques*" aimed at "*Integration and standardisation of enhanced product development tools for design, simulation, prototyping, testing and risk management that would reduce product development time and all associated costs and resources*".
- Maintenance: hardware and software yearly maintenance costs are taking into account by considering a lump sum of $100,000 per year. The value decreasing time effect is considered to be balanced by the actual maintenance cost for a given service level that it is expected to occur.

Benefits

- Higher margins due to more cost effective development of products. Automotive profit margins are very low. A 2% increase is conservatively forecasted.
- Increased market share and turnover due to the higher flexibility, shorter development time and lower development costs. If an organization has an efficient design process, the chances of developing a winning proposal for the customer are increased. A 20% increase in turnover is projected.
- Reduced development costs. While currently 2.7 persons are required to work 15 months to develop a maniverter, with the new process 1.8 persons are estimated to be able to develop a product in 4 months. This represents a huge saving for the company.
- Spillovers of the same technology to different products within the same company and research on new products, enabled by the availability of freed resources. Since these are not easily predictable, they are neglected.

A quantitative estimation of the expected increase in net income is presented in Tab. 14.

Starting from publicly available financial data of ArvinMeritor turnover and net income for the entire corporation and for the Light Vehicle Systems (LVS) division (ArvinMeritor 2003 Annual Report), figures are drawn related to the manifold/maniverter business segment only, assuming that this constitutes 10% of the total LVS business. These data are used as a basis to estimate the increase in net income due to 20% additional turnover and due to 2% margin increase.

Ten different projects are supposed to run concurrently. Consequently, based on the resource requirements, project duration and money expenditure for one single project (Tab. 12), the total number of FTE required yearly for the overall manifold business is estimated. The calculation is done for the current PDP and the new EDF enabled one. The difference is then calculated both in terms of freed resources and of cost savings. Additional FTE resources required by the increased turnover is also considered.

	Quantity (money figures are in k$)	Notes
Total ArvinMeritor Turnover	$7 788 000	From ArvinMeritor 2003 Annual Report
Total A&ET Turnover (30%)	$2 336 400	30% of the total, as per Annual report
Total Manifolds Turnover (10%)	$233 640	Assumed to be 10% of the A&ET
Average Number of concurrent projects	10	
R&D as percentage of Turnover	1.73%	Virtual Development only
Total ArvinMeritor Net Income	$136 000	From ArvinMeritor 2003 Annual Report
A&ET New Income (30%)	$40 800	From ArvinMeritor 2003 Annual Report
Manifolds Net Income (10%)	$4 080	
Profit margins (after tax)	1.75%	
Increased margins	**$81.60**	Assumed 2% increased due to more cost effective produc development
Percentage increase in turnover	20.00%	20% (due to greater flexibility and reduced R&D costs), i.e. 2 additional projects
Increased Turnover	$46 728	
Increased net income due to turnover increase	**$816**	
FTE current process	27.4	
FTE new process	4.8	
No. Of freed FTE	22.6	
Increased FTE due to additional turnover	1.0	
Annual FTE savings	**$3 195**	
Higher Net Income	**$4 092**	

Tab. 14: Financial Benefits of the application of the EDF

The EDF related costs and expected savings are combined to compute several project financial indicators: Net Present Value (NPV), (Tab. 15), Discounted Payback Period (DPP) and Internal Rate of Return (Tab. 16). A temporal horizon of 6 years of utilization of the tool (up to 2015) is considered.

Interest/discount rate	7.000%
Yearly increased income:	$4 092 240
NPV=	**$11 845 233**

Year	Year	Cash flow		Present Value		NPV at year
		Development costs	Increased income	Development costs	Increased income	
2004	0	$2 662 368	$24 553 440	**$2 062 150**	**$13 907 383**	
2005	1	$409 277		$382 502		-$382 502
2006	2	$478 483		$417 926		-$800 427
2007	3	$637 632		$520 498		-$1 320 925
2008	4	$371 088		$283 101		-$1 604 026
2009	5	$165 888		$118 276		-$1 722 302
2010	6	$100 000	$4 092 240	$66 634	$2 726 832	$937 896
2011	7	$100 000	$4 092 240	$62 275	$2 548 441	$3 424 062
2012	8	$100 000	$4 092 240	$58 201	$2 381 721	$5 747 582
2013	9	$100 000	$4 092 240	$54 393	$2 225 907	$7 919 096
2014	10	$100 000	$4 092 240	$50 835	$2 080 287	$9 948 549
2015	11	$100 000	$4 092 240	$47 509	$1 944 194	$11 845 233

Tab. 15: EDF Development Project: NPV and DPP

IRR	**56.443%**
Yearly increased income:	$4 092 240

Year	Year	Cash flow		Present Value	
		Development costs	Increased income	Development costs	Increased income
2004	0	$2 062 368	$24 553 440	**$720 922**	**$720 922**
2005	1	$409 277		$261 614	
2006	2	$478 483		$195 504	
2007	3	$637 632		$166 534	
2008	4	$371 088		$61 952	
2009	5	$165 888		$17 703	
2010	6	$100 000	$4 092 240	$6 821	$279 142
2011	7	$100 000	$4 092 240	$4 360	$178 431
2012	8	$100 000	$4 092 240	$2 787	$114 055
2013	9	$100 000	$4 092 240	$1 782	$72 905
2014	10	$100 000	$4 092 240	$1 139	$46 602
2015	11	$100 000	$4 092 240	$728	$29 788

Tab. 16: EDF Development project: IRR

The NPV is solid positive, the DPP is less than one year and the IRR features an impressive 56%. All indicators strongly mark that developing the EDF is an incredibly profitable project, even if worsening factors that may have been neglected in the current analysis, intervene.

The major saving comes from headcount reduction. The power of the tool, in fact, allows reducing the number of development engineers from slightly more than 27 people to 4 to which one is added to cope with the additional workload due to increased turnover. Incidentally, we note that 27 people is the approximate current size of the Center Of Competence Manifolds – the functional group currently in charge of manifold development at ArvinMeritor – plus few other analysts outsourced for calculation.

The net organizational result is that 22 people are freed up. We argue, however, that the company should not view this as a mere headcount reduction, but as a big opportunity to utilize the resources for innovation and long-term growth. The organizational issues related to the development and introduction of the new tool are discussed in detail in Section 5.7.

5.6. Benefits amplify in a virtous spiral

The application of the EDF is expected to have a direct and significant impact on the financial statement. In addition to the obvious increase in the revenue stream (due to more business captured) and in earnings (due to a more efficient execution), the following effects are anticipated [75] :

- Valuation Impact. Valuation is the price that investors place on the company stock based on financial performance and anticipated performance. Typically, valuation will follow market and industry segment trends. A company will be rewarded over the long term with a higher valuation if it demonstrates that it can outpace growth in the market and utilize assets efficiently to create earnings. Short-term fluctuations in valuation are impacted by market conditions while longer-term trends are more indicative of the company's financial health. To this end, process improvements such as those enabled by the EDF will have the greatest chance to impact valuation over a long period of time.
- Balance Sheet Impact. The effect of debt is important in understanding the dynamics of process improvements. Companies with poor processes or engineering teams that do not respond well to the changes of the program run the risk of compounding their problems with a surge in debt. Short and long-term liabilities accumulate quickly when an organization is attempting to complete work on a given schedule, or within a given market price point. When market demands drive the cost of the product down, less engineering R&D costs can be absorbed, offsetting profits. If the engineering and R&D costs are high to begin with, then the organization must cover more expenses with either overhead charged to the customer, or extra liabilities. An organization that becomes burdened in debt is a punished by investors. Hence, process improvements make the likelihood of debt run-up lower, and reduce the risk of an organization's stock becoming penalized due to over-leveraging.

- The Effect of Interest. Finally, the "cost of money" can be an indirect effect for an organization planning the rollout of a product to market. Delaying operations by weeks or months will compound short-term debt problems and give warning to analysts. Another undesirable side effect is the cost in terms of interest payments. Not only delays are eroding the balance sheet, but also the costly fees associated with high borrowing are lost forever. Streamlining the process so that delays are avoided will save organizations costly interest fees.

But the application of an EDF will yield other strategic advantages:

- Speed. In addition of the obvious advantage of low cost, the shortened development time is, in itself, a source of competitive advantage
- Innovation: resource freed up can be used for medium-long term research projects and to tackle new product segments

5.7. Implementation and organizational changes

"The goal is to use human-centered design processes that will result in efficient, effective, user acceptable system interfaces that will be simple to train, use, and maintain".

We hope that by now it is evident to the reader how big are the potential advantages that the MDO approach within an ICE platform could bring to a company that embraces it. However, as in car races a great car without a great driver is going to fail delivering the expected results, to fully reap these benefits, a company needs to shape the organization and its processes in order to adequately exploit the capabilities that the EDF offers. This Section is intended to briefly discuss the organizational implications of the adoption of the EDF and to outline a transition plan from a traditional matrix structure to a novel Application / Product Knowledge / Function Expertise structure. What described applies strictly to the part of the organization that currently performs the activities of product design and development (i.e. excluding HR, Finance, Administration, etc). However, we know, people are resistant to change. The more radical the change the greater the inertia we can expect. It's demonstrated that any improvement program, no matter how good it is, can fail because it's introduced in the wrong way. Many are the examples that we can cite: TQM, Six Sigma, etc. The difference between an outstanding success and a fiasco lies invariably in the early phases of their implementation. Therefore, we want to conclude this Chapter highlighting some of the common pitfalls of the implementation of any improvement program – including the EDF – to raise the awareness of an organization that would embark in the task of changing its development paradigm so that they can avoid stumbling and falling during the deployment.

5.7.1. Reshaping the Organization

Many automotive manufacturing and Tier 1 supplier firms are characterized nowadays by a traditional matrix structure. In what follows, we will therefore assume that a matrix structure is the baseline organizational form of a Company (such as ArvinMeritor) which decides to develop the EDF and plans its deployment. After a short description of the principal characteristics of the matrix structure, we outline the organizational evolution that should take place prior or, at least, in parallel with the adoption of the new development paradigm.

Matrix structures exist in various forms across a wide range of organizations. One of the most common characteristics associated with the matrix is the "mixed" or "overlay" organizational texture in which traditional, vertical hierarchy is overlaid by some form of lateral authority or influence [76] . As depicted in Fig. 139, the vertical hierarchy is traditionally functional and the horizontal "overlay" typically consists of projects, products, or business areas. In a matrix organization, usually, the Project or Programme or Product Manager (PM) leads the development effort, he/she is the primary customer interface and the focal point of the project, responsible for product performance, project timing and budget. The PM relies upon the expertise and the support of a cross-functional team whose members are part of individual functional groups, each one managed by a Functional Manager (FM). Customarily, PMs are dedicated to one or few projects, while functional members work on several different projects as far as their domain of expertise is concerned. For example, the CAE department is in charge to perform structural analyses on all types of components and subsystems. Usually, the Functional Manager sets working priorities for the members of his/her departments.

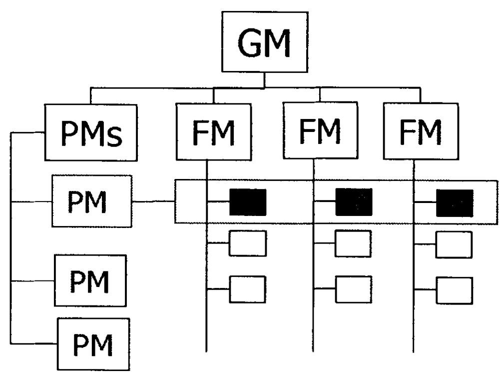


Fig. 139: Matrix Organizational Structure

The matrix structure exhibits a characteristic dual line of authority, responsibility, and accountability. As such, even it it's seen as a coordinative structural device which constructively blends the program orientation of project staffs with the specialty orientation of functional personnel in a synergistic relationship, it violates the traditional "one-boss" principle of management, according to which, for an effective management, no person should report to more than one boss.

The EDF crosses the boundaries of traditional matrix structure tapping into different areas of the organization. In some sense, it can be seen as a harmonized cross-functional Integrated

Product Team (IPT) that works with an incredible efficiency twenty four hours a day, seven days a week. And, as an IPT, its task is to manage the design complexity along its two main dimensions: 1) individual design analyses (traditionally the domain of the functional groups), 2) trade-offs between conflicting requirements on the different performance attributes (traditionally resolved through multiple design loops and cross-functional team meetings).

The "virtual IPT" is expected to impact significantly both PMs and Functional Groups. The following consequences are anticipated:

- Functional Groups: Dramatic reduction of standard work, mainly CAD and CAE. Since CAD models are generated automatically and handled by the geometry handler, traditional CAD modelling is no longer needed. CAD activity is indeed required only for models refinements and 2-D drawings. The same is true for CAE analyses: standard CFD, GT-Power, Finite Element analyses are, in fact, performed automatically without human intervention.
- Program Managers: A different decision making environment and different skills required. Results are no longer provided in a small set by the different functions, which also support the PM in interpreting the data. Instead, they are presented by the tool as a large set of numerical values (one or two order of magnitude higher than what currently happens) in a form of Pareto hyper-surface. On one side, this eases the role of a PM because the intrinsic trade-offs of the system have already been resolved and what needs to be done is "just" to select the most appropriate solution for the specific application. On the other side, however, the PM is required to be able to navigate comfortably through the results and to have a solid and deep knowledge of system's behavior to interpret the numeric data and to drive a sound engineering decision.

On the other hand, the new product development framework poses a new set of needs:

- Maintenance of the development platform
- Improvement of the platform
- Improvement of the individual modules

These issues can be managed only by a different organization or, better, by an organization with different foci. The main transformations that are anticipated are the following:

- Program Managers. The new development tool relieves them from the team coordination workload, which is now one of their major tasks, but simultaneously raises the importance of their role and their responsibility as decision makers. This requires that PMs' professional profile include a deep knowledge of all the different engineering domains involved as well as sound business skills. They appear to be more and more Product Managers rather than Project Managers.

- Functional Groups. They are probably the most impacted by the EDF. Greatly reduced in size compared with the current practice, their main task is envisioned to be advancing in the knowledge of their respective fields. They are responsible of the evolution of the different analysis modules to better and better reproduce with simulation the physical phenomena that are observed experimentally. Under the guidance of members of the Product Knowledge Teams (see later) they will responsible to identify better prediction models and to enable their execution within the development platform. As such they are supposed to be highly specialized and skilled on the individual engineering domains and analysis packages. A strong connection with the testing areas is highly desirable.
- Product Knowledge Team. This is a new group. It has a system focus, differently from the PMs who have a specific application and customer focus. One team is needed for each system the EDF applies to. This group is in charge of two main tasks: 1) to develop the knowledge on the particular system, 2) to maintain the development platform and to drive enhancements of its capabilities. Enhancements can come from either better analyses modules, from additional analyses modules (i.e. feasibility of stamped components) or additional system analysis capabilities (i.e. robust design). These professionals combine the expertise on the product with the competence in the design process. They develop the knowledge at the system level on the product/process (including the knowledge generated by the functional groups), they standardize and they embed it in the development framework that product managers will use as an off the shelf tool. Their role is crucial for the success of the company: leveraging the individual domain expertise, they will create the system's knowledge and pre-package it to enable the most efficient execution of individual projects.

In Section 5.5, we've seen that the application of the EDF reduces dramatically the number of people that are needed to develop the products: in the specific maniverter example, for the same workload, 5 FTE are estimated to be sufficient, compared with the current 27, i.e. more than 80% reduction. The great majority of this headcount cut is expected to occur in the functional groups together with a significant portion in the PMs group.

We advise, however, that the application of the EDF is not interpreted as a cost reduction operation, but as a transformation that has the potential to free up to 80% of the engineering development resources, which can be then "re-invested" to boost innovation for the medium-long term prosperity of the company.

In Chapter 1, we have seen as tomorrow's development will have to cope with engineering challenges of increasing difficulty with no correspondent increase in R&D budgets. Innovation appears as the main key to survive and prosper in today's and tomorrow's tough automotive industry environment. The engineering resources that are no longer entrenched with the day-by-day work can be adequately trained and re-directed to higher added-value activities such as:

- Developing next generation products
- Tackling systems of bigger scope
- Entering new business segments and/or creating new businesses
- Pursuing further cost reduction opportunities on existing products to boost income and as a marketing strategy

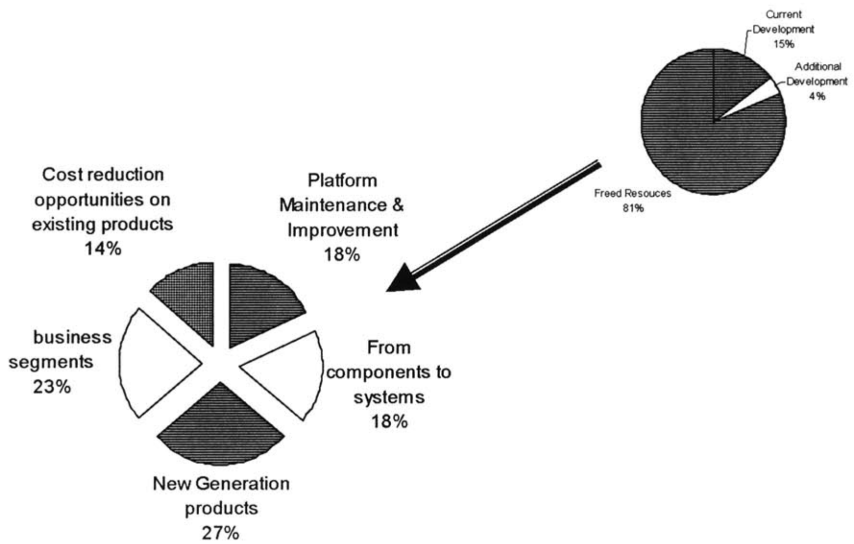


Fig. 140: Re-allocation of resources freed-up by the use of the maniverter EDF

Fig. 140 illustrates a scenario of re-allocation of the 22 resources that could be freed by the EDF.

ArvinMeritor has recently announced the decision to "separate the technology engineering from the application engineering", creating two new organizational entities that have been given the names: Centers of Competence (COCs) and Centers of Applications (COAs).[9]

This shift is welcome and goes in the direction of the proposed change. If we wanted to draw a correspondence between ArvinMeritor's implementation and the proposed organizational scheme, we can acknowledge that the PMs can be inserted in the COAs while Product Knowledge experts can be part of COCs. As far as traditional functional groups, they may splitted among the COCs if the analyses are very specific or, better, grouped in a service group.

[9] ArvinMeritor's Exhaust Strategy, http://www.autofieldguide.com/columns/0904euro.html

5.7.2. Be aware: it's going to get worse before getting better

The projected benefits and advantages of the application of the EDF must not induce in the management the false belief that as soon as the tool is ready, the company will be able to reap them, even if the adequate organizational structure is put in place, Fig. 141.

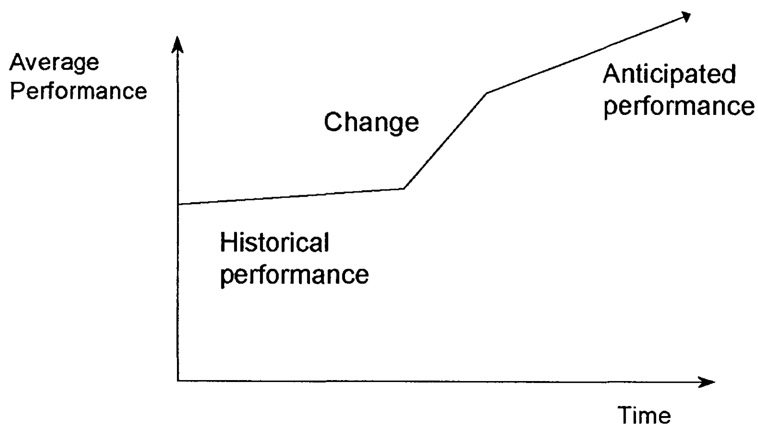


Fig. 141: Change: common expectations

Although successful deployment of improved tools and processes would unequivocally help the organization, they require the development of knowledge and experience. Consequently, introducing the tools actually lowers productivity in the short run while people learn and incorporate them into the normal practice, Fig. 142.

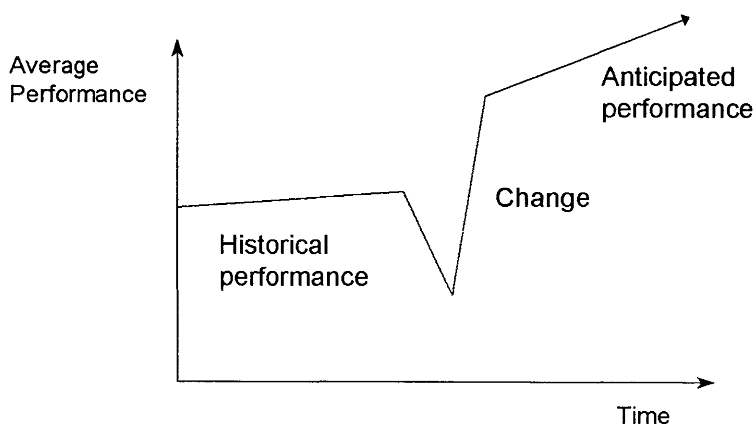


Fig. 142: The Reality of Change

Problems arise when managers ignore this worse-before-better trade-off [13] . The increase in workload arising from the additional training, learning, and practice time required to use the tools proficiently further raises resource utilization. Thus, if new tools are not accompanied by a reduced workload, their introduction is likely to lead to more fire fighting and to a further decline in process capability.

Therefore any organization that decides to embrace the new EDF has to allow extra time in the first couple of projects that use the new technology to learn it and work the kinks out of it.

5.8. The Following Generation: Modeling Uncertainty to Manage Risk

In Section 5.2 we have discussed a possible 4Y development plan of the first generation of the Enhanced Development Framework for a maniverter application and in the subsequent Sections we have projected ourselves in 2009 discussing the new operating environment and the related set of hardware/software/organizational issues.

In this Section we want to take a step even further and project a glimpse on what could be the second generation of the EDF, besides the incremental normal enhancement. We deem that the following leap will be made by transforming the analyses model to include the variability of the man-made products.

Materials are not homogenous, boundary conditions are not ideal, geometry is not perfect, and loads are subjected to unexpected fluctuations. Uncertainty originates from the very heart of physics and is deeply rooted in the nature of matter. Not surprisingly, therefore, it accounts for a huge chunk of the phenomena we observe.

Uncertainty is customarily accommodated in engineering via safety factors. This simple stratagem transforms a stochastic problem into a deterministic one. However, as we know, there is always something that is left unmodeled, some simplifying assumption that proves wrong, some unfortunate and unanticipated combination of factors that finally lead to an expensive law suit or recall or perhaps to catastrophic collapse or even loss of life.

With spectacular advances in computing technology we can expect to be able to take uncertainty into account in the very way it manifests itself in nature. The tremendous advantage of doing this is that models incorporating uncertainty become extremely realistic so that they allow us to understand and manage uncertainty.

For the first time, models will be realistic. The inclusion of elements of uncertainty in computer models will boosts the realism of these models to unexpected and unthought-of levels. Model precision will loose the meaning it has today

The scenario we envision is that, after EDF has provided the Pareto optimal set of solutions and the PM and the OEM have jointly chosen the configuration which gives the "right" performance attributes levels, a fine tuning sensitivity and optimization analysis is performed with the goal of maintaining those performance attributes at Six Sigma level. We can even stretch our vision and think that the algorithms for Pareto front extraction can be made so efficient and the computing power be so high that uncertainty can be included in the mainstream development process. In this case, the result of the design work will not be a Pareto hyper-surface but rather a collection of iso-probability surfaces. In that case, data presentation issues are amplified, but once the correct visualization approach is put in place, the decision making capabilities will be significantly augmented.

Examples of robust design/optimization and uncertainty analysis are already available now: optimization including uncertainty is at the heart of the Stochastic Design Improvement pioneered by MSC [21] and research projects multiply (for some automotive examples, see [77] , [78]).

Depending on the advancement of the knowledge on the specific field, the inclusion of the uncertainty management could be already inserted in the proposed EDF development, instead of starting five years from now. Critical uncertainties to be included are anticipated to be, in addition to material properties, dimensions of the components, assembly tolerances and vibrational loads.

6. CONCLUSIONS AND RECOMMENDATIONS

> *"Systems engineering is an interdisciplinary approach to evolve and verify an integrated and optimally balanced set of product and process designs that satisfy user needs and provide information for management decision making"*[10].

The analysis of Chapter 1 was hopefully successful in demonstrating that the automotive industry, after an extensive cost cutting phase, has by now reached a turning point where only a major evolutionary step is going to ensure sustained profitability. Significant contribution to this transformation will come from the application of systems engineering principles to product development and organizational design. New tools and approaches enabled by the advancement in computing technology and simulation capabilities will break the development speed-cost-quality iron triangle, bringing throughput to unprecedented levels. Multi-disciplinary Design Optimization is believed to be one of the most important elements of this (r)evolution and in the present work we tried to highlight its tangible huge potential.

In this last Section, we attempt to cast some light on what is expected to happen in the next decades. Our vision of the future of global automotive enterprises is the one of agile entities, capable of designing and delivering products quickly to a more demanding social community. In this scenario MDO is expected to be a standard product development practice.

We recommend, therefore, that automotive companies – including ArvinMeritor – shift their focus from cost cutting, which, if pursued further, could actually endanger the long-term sustainability of the company, to the development of new processes and tools. In particular, we suggest to start the exciting MDO journey as soon as possible to be the market leader in the 21st century.

6.1. A vision for the 21st century automotive industry: network of adaptive, agile, lean, self-controlling organizations

> *"The successful company of the future must understand how people really work and how technology can help them work more effectively. It must know how to create an environment for continual innovation on the part of all employees. It must tap the latent needs of customers. It must use research to reinvent the corporation."*

[10] Definition of systems engineering contained in the May 15, 1991 Pre-coordination Draft of Mil-STD-499B Systems Engineering.

Predictable markets, product lines, and business models – key assumptions in long-range business planning – are a fading memory today. Efficient channels of communication and distribution, accelerated product development, define a new business environment.

6.1.1. Change as a Mode of Operation

Whether deliberately introducing change into markets, or reacting to external change before competitors could, businesses will rely on speed to secure competitive advantage. The organization's ability to swiftly and easily accommodate and even anticipate change is therefore going to be a core requirement for business success in the 21st century.

Business agility[11] is believed to have three main dimensions [80] :

- Time needed to implement or react to a change in the business environment
- Range of implementation across geographies, business processes, or operating units
- Ease of change deployment, measured as labor or expense

This new level of business agility requires a flexible, adaptive enterprise that allows business to manage, control, and optimize the impact of change to the advantage of the corporation. The more flexible organizations will lower the change impact's gradient (number of changes and impact are, generally, linked by a non-linear function) (Fig. 143 [78]) and will perform better regarding time, quality and cost.

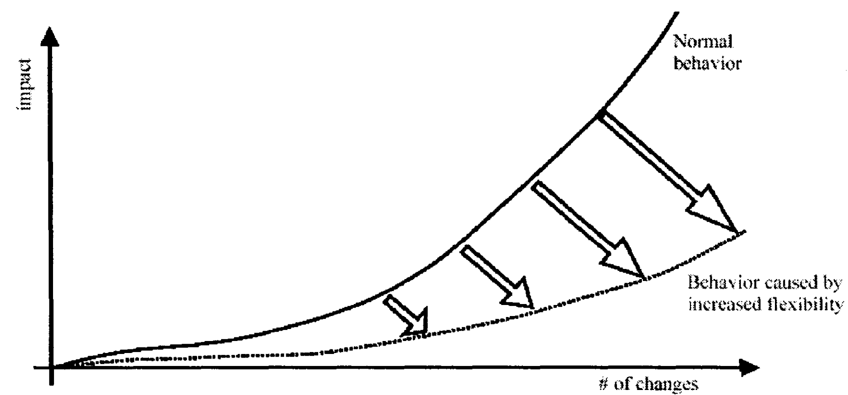


Fig. 143: Correlation of the number of changes to their impact

The economic consequences are expected to be significant. Automotive and Supply Chain experts at CapGemini Ernst&Young estimate that the achievement of an "adaptive" state has the potential to reduce the cost to build a vehicle by up to $1,050 per vehicle and compress the time from order-to-delivery from months to weeks or less.

[11] Agility can be loosely defined as "the ability of an organization to thrive in a continuously changing, unpredictable business environment."

Agility targets reducing unnecessary costs effectively by shifting emphasis away from cost reduction to increased throughput, market segmentation and rapid creation of new services and products.

Following Dove [81] to be agile, organizations must be able to both manage and apply knowledge[12] effectively. In the agile organization, in fact, knowledge management must be accompanied by change proficiency. Value from either capability is impeded if they are not in balance: Knowledge management without change proficiency leads to a catatonic organization, unable to move and follow the market, whereas change proficiency without enough knowledge management produces a spastic organization, Fig. 144, prey of fire fighting.

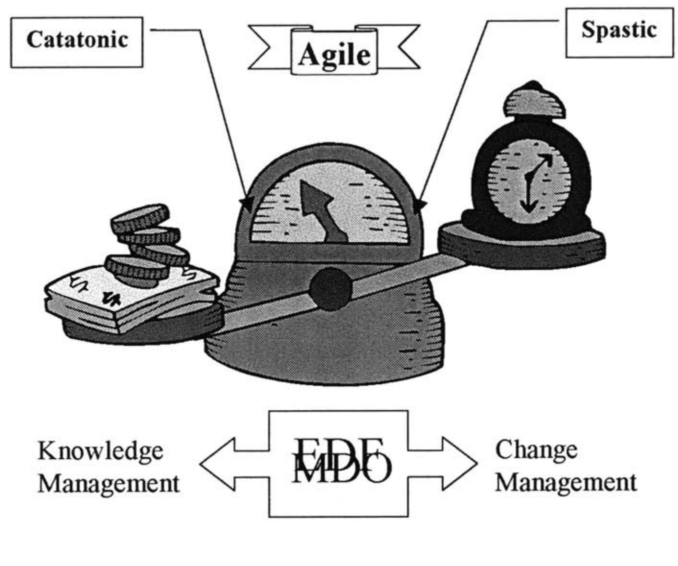


Fig. 144: The EDF as a tool to achieve business agility

Companies nowadays are finding it more difficult to stay in synch with the pace of change in their operational and competitive environments, because the increase of the knowledge content that they have experienced in these years – due to products and organizational complexity – hasn't been balanced by adequately change management processes.

6.1.2. Organizational Implications

Executives cannot simply take the organization and its current operations for granted including all existing functions, business processes, and information systems but they must rethink the very essence of each process, in order to better adapt to the ever-more volatile changing economic circumstances. The new digital and agile enterprise, however, has to be paired with new organizational designs.

Over the past decades, it has become common for companies to describe their organization in the form of an organization scheme where the structure of the company is drawn up in the form of divisions, staff organization, and other functional or geographical sub-divisions within the company. Companies have followed the model introduced by A. P. Sloan at General

[12] Knowledge: the body of truth, information, and principles acquired and interpreted information that can be used. Source: www.iteawww.org/TAA/Glossary.htm

Motors in the early 1920's. Within the company, work for the employees have been described in service and working instructions.

However, the structural constraints of the formal organization, with its reliance on standards, norms and rules for its operation, provide a poor foundation for change and adaptation to new conditions. In addition, we also have to recognize that strong resistances to change are inborn. Organizational behaviour is resistant to change due to cognitive processes and defensive routines: people make sense of past behaviour by forming beliefs that rationalize them and by escalating commitment to them; they also avoid embarrassment and threat to self and others. The heavy demands of today and tomorrow on the ability of a company to be flexible and constantly improve its performance, accent therefore the importance of new forms of coordination.

Networked self-organized entities are believed to provide the required human fabric of the agile enterprise, where the formal and the informal structures and the mental models of the employee and the corporation fuse together. Temporarily formed groups or teams, in common with other forms of flexible organization that adapt to operational changes, will become more prevalent.

This is a field of research on its own and further discussion at this point will bring us too far off track. The interested reader is suggested to consult the resources cited in Appendix 7.7.

6.1.3. The Role of Technology

In this business process re-engineering, technology will play a key role. In the 21st century, computers and information technology will be "power tools" to augment creative humans in product design by automating routine tasks and providing easy access to appropriate information, tools, and knowledge. The tools and design environment will stimulate the innovative process so that ideas are converted to wants and needs, needs become product requirements, and requirements drive design – all of this in a tradeoffs environment

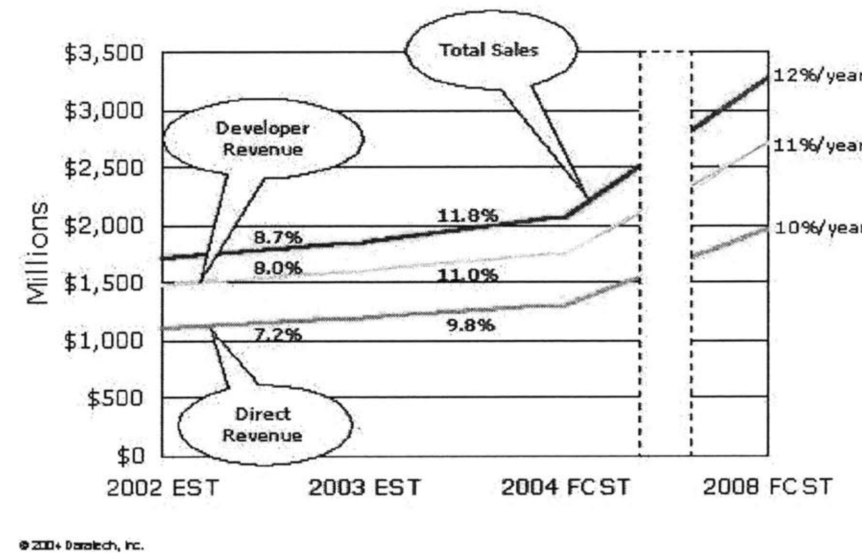


Fig. 145: Investments in Digital Simulation Technologies

allowing the best decisions to be made. Computers will not replace human creativity, but instead will enable creation of far better designs orders of magnitude faster than with today's systems. Computers will also expand the range of collaboration that is practical, as remote telecommunications and interactive application sharing will become commonplace.

The prodromes of this transformation are already evident, witnessed by the high ferment around digital simulation tools. As a Daratech study shows, investment in CAE technologies topped $2.1 billion in 2004, an increase of 12% over last year. When one considers spending on product lifecycle management reached $8.6 billion in 2004 and grows 8% each year through 2008, a clear market picture of CAE's importance emerges. Approximately 25% of PLM investments came from digital simulation in 2004 and, over the next five years, digital simulation will be the growth engine of PLM, rising 12% annually over that time (see Fig. 145). Driving growth will be a combination of advances in high-performance computing along with the increasing recognition of digital simulation's ability to generate higher quality and more innovative products faster and at lower costs than is possible with traditional methods.

Today, in fact, advanced CAx is promising much more than increased productivity. It promises faster times-to-money, lower warranty costs and above all, products that outperform, work better, are safer and fail less often.

The themes of evolution of digital simulation are expected to be[13]:

- Integrating CAE with Product Lifecycle Management (PLM)
- **Simulation-based design**
- Multi-domain integration
- **Automating complex work processes**
- Enterprise-level drivers for wider CAE deployment
- Long-term role and outlook for physical test
- Simulation data & process management: collaborating, archiving, re-use
- Advances in analytical-to-physical correlation
- **Multi-disciplinary and multi-objective optimization**
- Design for Six Sigma strategies
- Organizational & cultural challenges: management best practices
- Strategies for simulation of multi-domain systems: electronics, mechanics, hydraulics, controls
- **Supercomputing, HPC, grid computing**, clustering, 64-bit: maximizing performance
- Management strategies for regulatory compliance

[13] Top 15 priorities declared at the 2004 Daratech Digital Product Simulation and PLM conference

- Strategies for managing and integrating in-house product development with outsourced product development

6.2. MDO: a fundamental knowledge management tool for high performance product development process of the new agile enterprise

> *"Research on new work practices is as important as research on new products"*
> John Seely BrownFormer Chief Scientist of Xerox CorporationFormer director, Xerox PARC

We believe that the "computerization" of product development embedding an MDO approach has the capability to improve knowledge management but, at the same time, to greatly increase change proficiency, thus bringing the company to unprecedented levels of agility.

Our vision for the future of engineering design, and for automotive systems design in particular, is that of a Multidisciplinary Design Optimization (MDO) environment where it is possible to perform the design optimization of complex engineering systems using computational tools. The MDO approach, powered by the ICE platform, will automate much of the design configuration process and put product engineering at the heart of the design process. This automation, however, will not be "black-box" engineering, but rather the execution of known engineering steps to evaluate design alternatives, providing engineers with information to make better decisions and to rapidly respond to defined and projected needs at manageable cost.

These are some of the perspectives in this futuristic environment:

- Scenario-based conceptual models will allow the customer to evaluate and understand their preferences and the results and implications of those preferences
- Intuitive systems will provide physically and mathematically accurate visualizations that support trade-offs for optimization based on performance parameters and preferences
- All systems will be seamlessly interoperable
- All information needed for design and other applications will be contained in an accessible repository and readily useable by any system. The maintenance and usability of the data is independent of any specific system or format
- Real life modelling will include uncertainty and provide the foundations for more robust designs
- "Automatic" calibration of integrated models with experimental data will be implemented (self-learning)

The ability to re-engineer products rapidly and the emphasis on design assessment, comparison and improvement will almost inevitably lead to better engineering solutions to product design problems and to solutions configured instantaneously to meet fast changing customer needs.

The optimization algorithms, not constrained within the well-known ridges of common sense and practice could adventure safely in new areas of the design space leading to innovative, high performance designs.

The freeing of experts from team supervision, teaching and routine engineering work will further enhance their ability to discover engineering improvements and allow them to devote to research and innovation.

This new product development environment enabled by pervasive computing has at its heart knowledge but it is also man-centric. Its creation stems from the deliberate analysis of the strengths and weaknesses of the sensorial and cognitive capabilities of man and it is developed by man to exploit the first and to complement the latter.

6.3. The big risk is delayed action: take a bold-face decision

> *"Even the longest journey starts with a single step"*
>
> Old Chinese saying

Once upon a time, 35 years ago, Computer-Aided Design or CAD was touted by the National Science Foundation as having the greatest potential to improve productivity since the advent of electricity. The visionary executives of the day pushed through the adoption of this emerging technology against enormous odds because they recognized and believed in CAD's potential. People with no familiarity of computers were asked to change their ways and set aside a lifetime of training. A highly unionized work force was persuaded to adopt technologies that promised to eliminate jobs. And a re-education was necessary that did not get people up to full speed for approximately a year. Nonetheless, top management was sold on the big picture and this made it easy for people in the middle to take on challenges inherent in any revolutionary changes. This is not yet happening with frameworks like the one described in the present work, but it will.

Yet many companies will fail to make the shift, because the technology is believed to be immature and it's not trusted. Many companies will fail due to the inadequacies of their leadership.

Taking a boldface act, we recommend that EDFs' development projects be started selecting areas (products, divisions, etc) which could benefit most from the application of such an approach. We've provided elements to prove that, even if neglecting any strategic implication, these project are self-justified by a sound positive business case. By taking a system's perspective, we also suggest that these projects be not narrowedly limited to the investigation of optimization algorithms or simulation models, but consider the MDO approach in its full articulation that includes design problem formulation and solution, the information flow and management, and the organization and culture aspects, Fig. 146 [82]

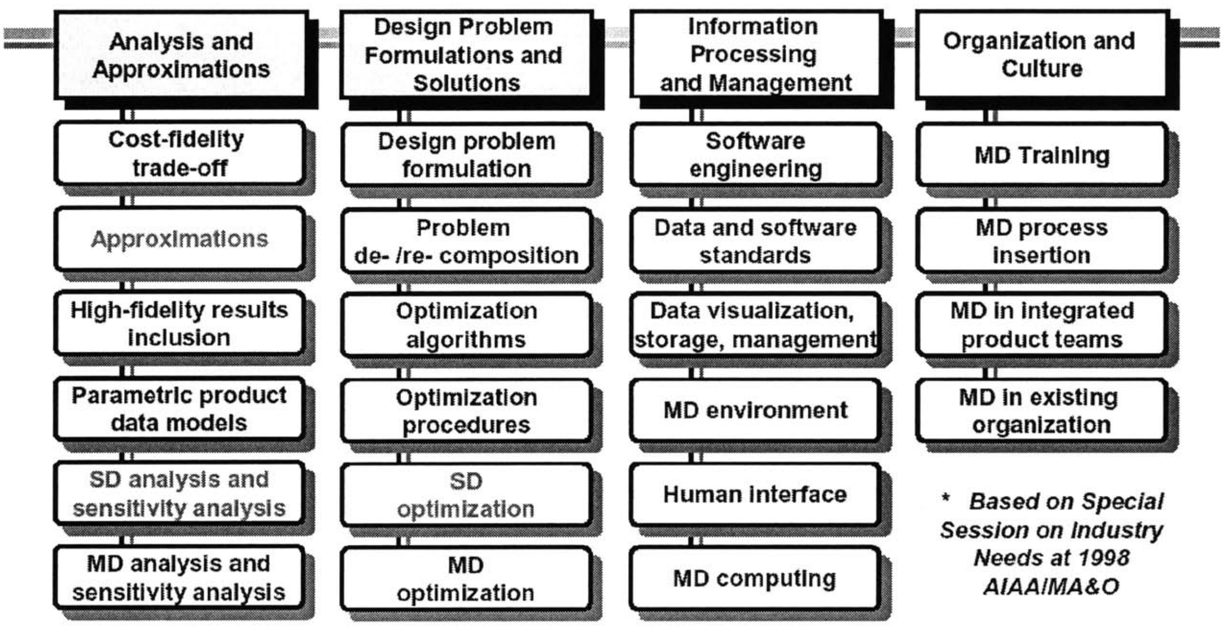


Fig. 146: The Articulation of MDO

"Those who fail, or refuse to adjust to it [change], are condemning themselves to professional obsolescence. How can you adapt to change? First, try to understand it...With understanding comes confidence. That's why the first step toward coping with change is understanding it, the whys, hows and whats of it.... And, face the change with confidence."
[Electric Light and Power editor Ted Pollock, July 2004.]

The combination of demanding customers, pressured profitability, troublesome environmental regulations, worldwide competition and intractable labor agreements is breaking the mold of the 20th century model of global automotive organizations.

Cost cutting as the sole approach to fattening margins results invariably in a reduction of operational capabilities which is likely to result in a decline in sales volume that leads to further cost reductions in a continuous death spiral [83] . Cost cutting has a short-term focus. It can only be done for the current situation and is not future-oriented.

Long-term profitable growth requires, instead, a continuous flow of innovative products and processes that align with customer needs. In the long run, it is much cheaper to design organizational processes that allow mastering fast, low cost change and creating a growth cycle. The key to success derives, indeed, from our ability to take advantage of change. Evolving from the assembly line and scientific management, which taught people to think and function in machine-like ways, our task is to create organizational structures based on systems thinking and change which are agile and can adapt to the new fast-paced morphing environment.

'Multidisciplinary optimization' (MDO) – an emerging discipline that stems from Systems Engineering and that relies on mathematics, statistics, operations research and computer science – integrated in a High Performance Computing Integrated Concurrent Engineering Platform is going to be the cornerstone of the new flexible product development paradigm.

Nobody will be able to immediately start a fully operational new agile and adaptive product development environment. However, the winner will be the one who has a clear vision of the final agile state, start earlier on the journey to achieve this vision, and implement it piece by piece.

"It's not the strongest of species that survive, nor the most intelligent, but the one most responsive to change".

Charles Darwin

7. APPENDIX

7.1. Geometrical Model Indipendent Parameters List

The table that follows report the geometrical model independent parameters. The first block (identified with the green color) refers to independent parameters that can be varied during the optimization, the second (identified with yellow) to those that are fixed with the application. Units are [mm] for dimensions and coordinates, [°] for angles.

No.	Parameter	Value (Baseline)	Description
1	C2a_x	40	Pipe A, Control Point 2, x coordinate
2	C2a_y	-4	Pipe A, Control Point 2, y coordinate
3	C2a_z	9	Pipe A, Control Point 2, z coordinate
4	C2b_x	90	Pipe B, Control Point 2, x coordinate
5	C2b_y	77	Pipe B, Control Point 2, y coordinate
6	C2b_z	7	Pipe B, Control Point 2, z coordinate
7	C2c_x	110	Pipe C, Control Point 2, x coordinate
8	C2c_y	154	Pipe C, Control Point 2, y coordinate
9	C2c_z	9	Pipe C, Control Point 2, z coordinate
10	C2d_x	33	Pipe D, Control Point 2, x coordinate
11	C2d_y	230	Pipe D, Control Point 2, y coordinate
12	C2d_z	7	Pipe D, Control Point 2, z coordinate
13	C3a_x	71	Pipe A, Control Point 3, x coordinate
14	C3a_y	-2	Pipe A, Control Point 3, y coordinate
15	C3a_z	50	Pipe A, Control Point 3, z coordinate
16	C3b_x	130	Pipe B, Control Point 3, x coordinate
17	C3b_y	27	Pipe B, Control Point 3, y coordinate
18	C3b_z	40	Pipe B, Control Point 3, z coordinate
19	C3c_x	145	Pipe C, Control Point 3, x coordinate
20	C3c_y	190	Pipe C, Control Point 3, y coordinate
21	C3c_z	60	Pipe C, Control Point 3, z coordinate
22	C3d_x	44	Pipe D, Control Point 3, x coordinate
23	C3d_y	198	Pipe D, Control Point 3, y coordinate
24	C3d_z	60	Pipe D, Control Point 3, z coordinate
25	C4a_x	100	Pipe A, Control Point 4, x coordinate
26	C4a_y	91	Pipe A, Control Point 4, y coordinate
27	C4a_z	110	Pipe A, Control Point 4, z coordinate
28	Dinta	26	Pipe A, Inner Diameter
29	Dintb	26	Pipe B, Inner Diameter
30	Dintc	26	Pipe C, Inner Diameter

No.	Parameter	Value (Baseline)	Description
31	Dintd	26	Pipe D, Inner Diameter
32	diametro_max	30	
33	diametro_sf	102	Hemy-sphere Inner Diameter
34	rot_dir_cil_x	-6.5	Brick Angle, X
35	rot_dir_cil_y	8	Brick Angle, Y
36	sp_cono_uscita	2	Outlet Cone Thickness
37	sp_ext_cilindro	2	Converter Can Thickness
38	sp_sf_cono_ingresso	2	Inlet Cone thickness
39	sp_staffetta_cil_fisso	2	Bracket Thickness
40	spa	2	Pipe A, Thickness
41	spb	2	Pipe B, Thickness
42	spc	2	Pipe C, Thickness
43	spd	2	Pipe D, Thickness
44	x_2	100	Point 2, x coordinate
45	y_2	90	Point 2, y coordinate
46	z_2	144	Point 2, z coordinate
47	C1a_x	0	Pipe A, Control Point 1, x coordinate
48	C1a_y	0	Pipe A, Control Point 1, y coordinate
49	C1a_z	0	Pipe A, Control Point 1, z coordinate
50	C1b_x	0	Pipe B, Control Point 1, x coordinate
51	C1b_y	77	Pipe B, Control Point 1, y coordinate
52	C1b_z	0	Pipe B, Control Point 1, z coordinate
53	C1c_x	0	Pipe C, Control Point 1, x coordinate
54	C1c_y	154	Pipe C, Control Point 1, y coordinate
55	C1c_z	0	Pipe C, Control Point 1, z coordinate
56	C1d_x	0	Pipe D, Control Point 1, x coordinate
57	C1d_y	231	Pipe D, Control Point 1, y coordinate
58	C1d_z	0	Pipe D, Control Point 1, z coordinate
59	diametro_cil	106	Brick Diameter
60	diametro_cil_fisso	45	Outlet Pipe Inner Diameter
61	diametro_foro_staffetta	10	Bracket Hole Diameter
62	estensione_labbro_inf	40	Bracket Lower Wing Length
63	estensione_labbro_sup	40	Bracket Upper Wing Length
64	lunghezza_cil	127	Brick Length
65	lunghezza_cil_fisso	280	Outlet Pipe Length
66	semi_apertura_inf_staffetta	3	
67	semi_apertura_sup_staffetta	3	
68	sp_cil_fisso	2	Outlet Pipe Thickness
69	sp_mat	6	Mat Thickness
70	spessflangia	10	Inlet Flange Thickness
71	staffetta_fine	187.5	Bracket Leading Edge, x coordinate

No.	Parameter	Value (Baseline)	Description
72	staffetta_inizio	153.5	Bracket Trailing Edge, x coordinate
73	x_5	-23.5	Point 5, x coordinate
74	x_6	-263	Point 6, x coordinate
75	y_5	30	Point 5, y coordinate
76	y_6	13.5	Point 6, y coordinate
77	z_5	367	Point 5, z coordinate
78	z_6	374	Point 6, z coordinate

7.2. KEFAOptimizer

7.2.1. Command syntax

KEFAOptimizer is launched with the following command:

```
KEFAOptimizer2.exe -p <PART_FILE> -e <EXP_FILE> -o <OPT_FILE> -s <ParasolidFile> [-clean]

   -p <PART_FILE>       : part to be modified, eg: "-p part.prt"
   -e <EXP_FILE>        : file with expressions, eg:"-k tube.exp"
   -o <OPT_FILE>        : required params, eg: "-o opt.txt"
   -s <ParasolidFile>   : parasolid file name, eg: "-s outParasolid.x_t"
   -clean               : delete comments from expression file
```

7.2.2. Operating System Requirements

KEFAOptimizer works on Microsoft Windows 2000 and later.

It requires the UG NX2 software with licenses for Open API and Knowledge Fusion

Other OS requirements are the same of UG NX2.

7.3. BOOST Application Automation Input File

The interface definition consists of 2 parts: the first defines the data that should be updated inside the BOOST-model, the second describes the requested results:

Example:

```
<boost_automatization_interface>
   <boost_input>
         .
         .
   </boost_input>
   <boost_output_request>
         .
```

```
        .
    </boost_output_request>
</boost_automatization_interface>
```

The input section can contain any arbitrary number of input data definitions. The definition consists of two types of information: the name of the parameter, which should be updated, and the value of it (absolute value, not incremental). The condition, which has to be fullfilled for the update to be successful, is that parameters have to be so-called "Workspace"-Parameters. "Workspace"-Parameters are input data inside BOOST which are declared as a parameter. The declaration of the parameter is done interactively from the creator of the BOOST model.

A sample of the input section is given below:

```
<boost_input>
      <parameter>
            <name>Amb_T</name>
            <value>303.000000</value>
      </parameter>
      <parameter>
            <name>Brick_dia</name>
            <value>106.000000</value>
      </parameter>
<boost_input>
```

The iSIGHT parser is instructed to write the values of the parameters in the appropriate fields (see 3.10.2).

The results definition is also totally flexible. Here, in addition to specifying the requested results the following declarations have to be done:

- Name of the result export file
- If a post processing step of BOOST should be run.

Torque curves are obtained by combining the results from different single rpm operations. BOOST itself, adequately instructed by the user, is able to run multiple calculations. The user then makes the combination in the post-processing phase. The automation interface is developed with the same capabilities and therefore it needs the information regardless if the creation process of the combined results are necessary or not.

Example:

```
<boost_output_request>
      <filename>results_boost.dat</filename>
      <create_series_results>YES</create_series_results>
      .
      .
</boost_output_request>
```

After the definition of the output file and the post processing step, the results are defined. Also here, there is no limitation about the number of results.

The information which allows the automation layer to extract the requested results are:

- Name of the output value (which is written into the result file)
- Result file containing the data of interest (BOOST .gid file)
- Result column(s) inside the previous defined result file
- Extraction method: "LAST" or "ALL"

  In the case of "LAST" the last value of the result column is taken and written

  In the case of "ALL" the complete data column is written

Example

```
<boost_output_request>
      <filename>results_boost.dat</filename>
      <create_series_results>YES</create_series_results>
      <parameter_table>
            <parameter>
                  <name>RPM</name>
                  <result_file>ex1.cs1/simulation.dir/sEG1___1.gid</result_file>
                  <result_column>En_speed</result_column>
                  <result_methode>ALL</result_methode>
            </parameter>
            <parameter>
                  <name>TORQUE</name>
                  <result_file> ex1.cs1/simulation.dir/sEG1___1.gid</result_file>
                  <result_column>TORQUE</result_column>
                  <result_methode>ALL</result_methode>
            </parameter>
      </parameter_table>
</boost_output_request>
```

7.4. Resources for Advances in Product Development

- New Product Development Project Innovation International Conferences, on www.managementrounddtable.com
- Methods and Tools for Co-operative and Integrated Design, on http://cirp-dn2003.hmg.inpg.fr/scope.html

- International Symposium on TOOLS AND METHODS OF COMPETITIVE ENGINEERING, on http://dutoce.io.tudelft.nl/~jouke/tmce2004/
- Product Development and Management Association, on http://www.pdma.org/
- NEW PRODUCT DEVELOPMENT SOLUTIONS on http://www.npd-solutions.com/index.html

7.5. Resources for Multidisciplinary Design Optimization

- MDO TECHNICAL REPORTS ONLINE http://www.soton.ac.uk/~pbn/MDO/mdo_pubs.html (and also http://www.soton.ac.uk/~pbn/MDO/mdo_links.html)
- NASA Langley Research Center Multidiscipline Optimization Branch http://mdob.larc.nasa.gov/
- MULTIDISCIPLINARY DESIGN OPTIMIZATION TECHNICAL COMMITTEE http://www.aiaa.org/portal/index.cfm?GetComm=80
- List of MDO and Optimization-Related Web Sites: http://www.ae.msstate.edu/~masoud/Research/optsites.html
- http://www.sgi.com/industries/manufacturing/mdo/#overview
- Association for Structural and Multidisciplinary Optimization in the UK

7.6. Resources for Knowledge Management:

- http://www.parshift.com/library.htm
- http://www.viktoria.se/results/result_files/171.pdf

7.7. Resources for Complex Systems and Chaos Theory:

- http://www.brint.com/Systems2.htm

8. BIBLIOGRAPHY

[1] David E. Cole, "The Auto Future: Fast, Fun & Scary", Automotive News World Congress Detroit, Michigan, January 15, 2003

[2] Gilles REINAUD, "Hard times for the automobile industry? Overcapacity and downturn in the market", Conjoncture, June 2001, pp. 24-31

[3] L. Yost , "ArvinMeritor – Analyst Day", 30 Sept 2003

[4] Accenture and The Automotive Supplier Industry; Exploring New Paths of Profitability, 2002

[5] Roland Berger – Strategy Consultants, "Automotive Engineering 2010 - Achieving More From Less", Detroit, July 20th 2004

[6] "GM springs for IBM supercomputer" (21st Apr 2004); Available from: http://marketwatch-cnet.com.com/GM+springs+for+IBM+supercomputer/2100-1001_3-5197178.html, [Accessed 30th Apr 2004]

[7] Bob Lewis, "When your business income and expenses are out of balance, don't cut costs"; Available from: http://www.infoworld.com/articles/op/xml/00/10/23/001023oplewis.html, [Accessed 30th Sept 2004]

[8] Michael Burkett, "Long-Term Profit Margins Demand Innovation, Not Just Cost Cutting", AMR Research Alert 17490, Wednesday, August 04, 2004

[9] N. P. Repenning, "Understanding Fire Fighting in New Product Development", March 2001

[10] N. Repenning, Paulo Gonçalves, Laura J. Black, "Past the Tipping Point: The Persistence of Firefighting in Product Development", 2001

[11] Sven Bertelsen, Lauri Koskela, "AVOIDING AND MANAGING CHAOS IN PROJECTS", International Group of Lean Construction 11th Annual Conference July 22 - 24, 2003, Virginia Tech, Blacksburg, Virginia, USA

[12] Brooks, F. P. Jr. (1995). The Mythical Man-Month. Addison-Wesley.

[13] Rebecca Henderson, "Developing and Managing a Successful Technology & Product Strategy", Executive Course, 2003

[14] R. Henderson, "Generating Growth in the Operationally Driven Company", *Leaders for Manufacturing Conference 2003: Manufacturing and Beyond: Leadership, Innovation and Bottom Line, October 23 and 24 2003, Massachusetts Institute of Technology*

[15] Dörner, Dietrich (1996). The Logic of Failure. Perseus Books, Cambridge, MA. 222 p.

[16] Elizabeth C. Lempres, "A product is born", The McKinsey Quarterly, 2003 Number 3

[17] P. G. Smith, D. G. Reinertsen, "Developing Products in Half the Time", John Wiley and Sons, 1998

[18] Blanchard and Fabrycky, "Systems Engineering and Analysis", 3rd Edition, Prentice Hall, 1998, page 561, Fig. 173

[19] Gregory Roth, "The Value of Early Analysis (Part 1)", BENCHmark, July 2000, pp. 15-18

[20] Garreth Evans, "The Knowledge Revolution - A New Era", 2004

[21] Gene Allen & Jacek Marczyk, "Stochastic Simulation – A New Tool for Engineering", NDIA 6th annual Systems Engineering Supportability & Interoperability Conference, 22nd Oct 2003

[22] Ali Yassine, Nitin Joglekar, Dan Braha, Steven Eppinger, Daniel Whitney, "INFORMATION HIDING IN PRODUCT DEVELOPMENT: THE DESIGN CHURN EFFECT", MIT Sloan School of Management Working Paper 4333-02, January 2002

[23] Thomas A. Roemer and Steven Eppinger, "Tools For Innovation: The Design Structure Matrix", Summer 02, 15-761

[24] Filippini, Roberto, Salmaso, Luigi & Tessarolo, Paolo (2004), "Product Development Time Performance: Investigating the Effect of Interactions between Drivers", Journal of Product Innovation Management 21 (3), 199-214. doi: 10.1111/j.0737-6782.2004.00070.x

[25] "A New Approach to Rapid Product Development", National Institute of Technology Management & Fraunhofer Institute, 2002

[26] Monica Schnitger, "Digital Simulation to Meet Today's Product Development Challenges", Daratech White Paper, December 2003

[27] Phil Martens, "Making the Elephant Dance – The transformation of Ford's North American Product Development Operations", 2003

[28] Kodiyalam, S. and Sobieszczanski-Sobieski, J.(2001) 'Multidisciplinary design optimization – some formal methods, framework requirements, and application to vehicle design', Int. J. Vehicle Design (Special Issue), pp. 3–22.

[29] Current State of the Art on Multidisciplinary Design Optimization, An AIAA White Paper, September 1991

[30] Srinivas Kodiyalam, "Evaluation of Methods for Multidisciplinary Design Optimization (MDO), Phase I", NASA/CR-1998-208716

[31] Srinivas Kodiyalam, "Evaluation of Methods for Multidisciplinary Design Optimization (MDO), Phase II", NASA/CR-2000-210313

[32] PANOS Y. PAPALAMBROS, "DESIGN OPTIMIZATION PRACTICE IN PRODUCT DEVELOPMENT", Invited presentation at the Engineering Design in Integrated Product Development, International Workshop and Seminar 2002, (EdiProD2002), Lagow, Poland, October 10-12, 2002.

[33] O. de Weck, K. Willcox, "Multidisciplinary System Design Optimization (MSDO)", MIT Course 16.888/ESD.77J, Spring 2004

[34] T. Simpson, D. Lin and W. Chen, "Sampling Strategies for Computer Experiments: Design and Analysis", International Journal of Reliability and Applications, 2001

[35] Timothy W. Simpson, Andrew J. Booker, Dipankar Ghosh, Anthony A. Giunta, Patrick N. Koch, and Ren-Jye Yang , "Approximation Methods in Multidisciplinary Analysis and Optimization: Panel Discussion", 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis & Optimization in Atlanta, GA, September 2-4, 2002.

[36] Miller, George, The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. The Psychological Review 63:81-97, 1956

[37] Preston G. Smith, "Field Guide to Project Management", chapter 32: Concurrent Engineering Teams, David I. Cleland editor, John Wiley & Sons, 1998,

[38] Exploiting eManufacturing: Interoperability of Software Systems Used by U.S. Manufacturers, NACFAM, February 2001

[39] Automated Generation of Models for Design and Manufacturing - A White Paper, 30 September 2002, Prepared in Support of the Modeling & Simulation for Affordable Manufacturing Technology Roadmapping Initiative

[40] "Moore's Law", Available from: http://www.intel.com/research/silicon/mooreslaw.htm, [Accessed on 23rd Oct 2004]

[41] John Dorfner, Wesley Jones, and Eric Ng, "Supercomputer Benchmarking", 2004, Available from: http://www.academic.marist.edu/~jzbv/architecture/Projects/projects2004/Supercomputer_Benchmarking.ppt [Accessed on 04th Nov 2004]

[42] 24th Edition of TOP500 List of World's Fastest Supercomputers Released: DOE/IBM BlueGene/L and NASA/SGI's Columbia gain Top Positions, Nov 9, 2004, 15:08, Available from: http://www.top500.org/news/articles/article_51.php [Accessed on 20th Nov 2004]

[43] Dr. Lisa T. Su, "Innovations for the On Demand Era: Breakthroughs Enabled by Technology", NEDA Executive Conference, October 18, 2004

[44] Cray Inc, Cray Update - SuperComputing 2004

[45] "The Robots Are Getting Closer", Jan 2003, Available from: http://hiqnews.megafoundation.org/2003-1-9_Robotics_Update.htm, [Accessed on 15th Nov 2004]

[46] Srinivas Kodiyalam, R. J. Yang, Lei Gu, Cheng-Ho Tho, "Large-Scale, Multidisciplinary Optimization of a Vehicle System in a Scalable, High Performance Computing Environment", White Paper, 2001

[47] G. Lombardi, A. Vicere, H. Paap, G. Manacorda, "Optimized Aerodynamic Design for High Performance Cars", AIAA-98-4789, MAO Conference, St. Louis, 1998

[48] Zhijun Li, Panayotis Georgiopoulos, Panos Y. Papalambros, Zoran Filipi, Guangquan Wu and Xiaodong Yang, "Model Based Analysis of Performance-Cost Tradeoffs for

Engine Manifold Surface Finishing", SAE 2004-01-1561, 2004 SAE World Congress, Detroit, Michigan, March 8-11, 2004

[49] G.A. Hazelrigg, "A framework for decision based engineering design", Journal of Mechanical Design, 120:653–658, 1998.

[50] Berry S., Levinsohn J., and Pakes A., "Automobile Prices in Market Equilibrium", Econometrica, 63:841–890, 1995.

[51] Mattias Nystrom, "Multidisciplinary Optimization with Application to Exhaust System Design", AIAA-2000-4749

[52] I. Arbuckle, S. Naylor, M. Worthington, "Optimization Strategies Applied to Exhaust System Design", SAE Paper No. 2002-01-0812

[53] M. Kanazaki, S. Obayashi and K. Nakahashi: "Multiobjective Design Optimization of Merging Configuration for an Exhaust Manifold of a Car Engine", *Proc. PPSN, the 7th international conference on parallel problem solving from nature, (2002), p.p.281-287.*

[54] Alvin Teo, "Exhaust Flow in an Auotmobile", Available from: http://www.quickhonda.net/exhaust.htm#Analysis%20of%20the%20Exhaust%20System%20in%20an%20Average%20Car, [Accessed on: 03rd Dec 2004]

[55] Donald H. Liles, Mary E. Johnson, Laura Meade, "Ryan Underdown, "Is Enterprise Engineering: A Discipline?", Society for Enterprise Engineering Conference Proceedings, June, 1995

[56] History of CAD - http://cadlab.daltonstate.edu/historyofcad.pdf

[57] Rohl, P.J. He, B. Finnigan, P.M., A Collaborative Optmization Environment for Turbine Engine Development", AIAA-98-4734

[58] Jami J. Shah, Martti Mäntylä, "Parametric and Feature-Based CAD/CAM : Concepts, Techniques, and Applications", John Wiley & Sons, 1995

[59] Dr. H. Felix Lee and Mr David W. Fulton, "Computer Aided Design with Unigraphics NX2", 4th edition

[60] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy, Uniformly high order essentially non-oscillatory schemes III, J. Comput. Phys. 71, 231 (1987)

[61] AVL BOOST, Version 4.0.4, Primer, June 2004

[62] AVL BOOST, Version 4.0.4, User's Guide, June 2004

[63] AVL BOOST, Version 4.0.4, Examples, June 2004

[64] D. Frey, P. Lagace, C. Magee, ESD.33 - Systems Engineering Course, Summer 2003

[65] SEG - Systems Engineering Glossary, Search for Search for a definition: Parametric cost estimate, Available from: http://www.seecforum.unisa.edu.au/glossary/display_list_definition.asp?term=58059761459760960161661460559953259961161561653260161561660560959761660l, [Accessed on 16th Dec 2004]

[66] iSIGHT Product Overview, Version 9.0, 2004

[67] C. Ware. Information Visualization – Perception for Design. Morgan Kaufmann Publishers, 2000

[68] G. M. Stump, M. Yukish, J. D. Martin, T. Simpson, "The ARL Trade Space Visualizer: An Engineering Decision-Making Tool", 2004

[69] M. J. Pastizzo, R. F. Erbacher, L. B. Feldman, "Multi-Dimensional Data Visualization", 2002

[70] H. Pohlheim, "Visualization of Evolutionary Algorithms - Real-World Application of Standard Techniques and Multidimensional Visualization", Proceedings of the Genetic and Evolutionary Computation Conference – Workshop Program, San Francisco, CA, 1999

[71] Hooke, R. and T.A. Jeeves, Direct Search Solution of Numerical and Statistical Problems, J. Assoc. Comp. Mach., 8, 212 (1961)

[72] Shinya Watanabe, Tomoyuki Hiroyasu, Mitsunori Miki, "NCGA: Neighborhood Cultivation Genetic Algorithm for Multi-Objective Optimization Problems", IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, 2002

[73] "Sound", Available from: http://www.science.tamu.edu/CMSE/powerpoint/Sound.ppt, [Accessed on: 04th Jan 2005]

[74] R. T. Biedron, P. Mehrotra, M. L. Nelson, F. S. Preston, J. J. Rehder, J. L. Rogers,D. H. Rudy, J. Sobieski, and O. O. Storaasli, "Compute as Fast as the Engineers Can Think!", Sept 1999, NASA/TM-1999-209715

[75] Brett Malone, "ON THE FINANCIAL IMPACT OF MDO ON THE CORPORATION", 2002

[76] Robert C. Ford, W. Alan Randolph, "Cross-functional structures: a review and integration of matrix organization and project management", Journal of Management, June, 1992

[77] Zhijun Li, Michael Kokkolaras, Dohoy Jung, Panos Y. Papalambros and Dennis N. Assanis, "An Optimization Study of Manufacturing Variation Effects on Diesel Injector Design with Emphasis on Emissions", SAE Paper 2004-01-1560

[78] Jan Jürging, "Flexibility: Bringing the one constant factor into projects", White Paper, 2003

[79] GOBBI, M., MASTINU, G., CATELANI, D., GUGLIELMETTO, L., BOCCHI, M., "MULTI-OBJECTIVE OPTIMAL DESIGN OF ROAD VEHICLE SUB-SYSTEMS BY MEANS OF GLOBAL APPROXIMATION", Adams User Conference Paper, 2000

[80] "Adaptive enterprise for business agility – A white paper from Hewlett-Packard Company", 2003

[81] Dove, Rick, "Knowledge Management, Response Ability, and the Agile Enterprise", Journal of Knowledge Management, 1999, Vol. 3, Issue 1, pages 18-35

[82] Sharon L. Padula, "Numerical Methods Overview", ASCAC Methods Development Peer Review November 27-29, 2001

[83] Michael Burkett, "Long-Term Profit Margins Demand Innovation, Not Just Cost Cutting", AMR Research Alert, Wednesday, August 04, 2004

[84] Accenture, "AUTO 2010 - An Expert Inquiry on the Future of the Automotive Industry", 2001

[85] D.H Brown, "Industry Perspective on Modeling & Simulation Tools/Issues", NDIA National Defense Industry Association - 5th Simulation-Based Acquisition/Advanced Systems Engineering Environment Conference; Role of Simulation and Advanced Engineering Environments in the Transformation of the Military, 6/24-6/27/02

[86] "A New Approach to a Rapid Product Development", National Institute of Technology Management, Fraunhofer Institute, 2003

[87] KPMG AUTOMOTIVE INDUSTRY SURVEY 2002, Momentum in the Automotive Industry, Vol. 2, No. 1, 2002

[88] Vanderplaats, G. N., Numerical Optimization Techniques for Engineering Design, 3rd edition, VR&D, Colorado Springs, CO, 1999.

[89] Timothy T. Maxwell, M. M. Tanik, "HARNESSING COMPLEXITY IN DESIGN", Transactions of the Society for Design and Process Science SEPTEMBER 2002, Vol. 6, No. 3, pp. 63-74

[90] Sarah A. Sheard, "TWELVE SYSTEMS ENGINEERING ROLES", Proceedings of the INCOSE Sixth Annual International Symposium, 1996

[91] Sven Åke Hörrte, "Social network analysis and Manipulative management", White paper, Institutionen för Industriell ekonomi och samhällsvetenskap, 1999

[92] CONNECTING TO YOUR FUTURE: THE NETWORKING OF EVERY MANUFACTURED THING, Harbor Research, Inc., 2004

[93] Carolyn Conner Seepersad, F. Scott Cowan, Matthew K. Chamberlain and Farrokh Mistree, "Strategic Design: Leveraging and Innovation for a Changing Marketplace", AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, August 30th – September 1st 2002

[94] Ahmed El-Bahrawy, Mattias Nyström, Luleå University of Technology Volvo Car Corporation, "Automated Integrated Finite Element Analysis Using MSC.Nastran and SDRC/I-DEAS"

[95] Dr. Se Il Song, "Multidisciplinary Optimization of an Automotive Door With Tailored Blank", FISITA World Automotive Congress 2004, F2004F422

[96] M. Kanazaki, S. Obayashi and K. Nakahashi, "Exhaust Manifold Design with Tapered Pipes Using Divided Range MOGA",

[97] Lee Ming Wong G. Gary Wang, "DEVELOPMENT OF AN AUTOMATIC DESIGN AND OPTIMIZATION SYSTEM FOR INDUSTRIAL SILENCERS", *Journal of Manufacturing Systems, (Accepted on Oct. 13, 2003)*

[98] G. Gary Wang, "Adaptive Response Surface Method Using Inherited Latin Hypercube Design Points", *Transactions of the ASME, Journal of Mechanical Design, Vol. 125, pp. 210-220, June 2003.*

[99] Nicola Senin, David R. Wallace, Nicolas Borland, "Distributed Object-Based Modeling in Design Simulation Marketplace", Transactions of the ASME, Vol. 125, MARCH 2003

[100] Shaun Abrahamson, Nicola Senin, David Wallace, Nick Borland, "INTEGRATED ENGINEERING, GEOMETRIC, AND CUSTOMER MODELING: LCD PROJECTOR DESIGN CASE STUDY", Proceedings of DETC 99, 1999 ASME Design Engineering Technical Conferences, 12-16 September 1999, Las Vegas, Nevada

[101] Jacob Wronski, J. Michael Gray, "MULTI-DISCIPLINARY SYSTEM DESIGN OPTIMIZATION OF THE F-350 REAR SUSPENSION", AIAA 2004

[102] V. O. Balabanov, C. Charpentier, D. Ghosh, G. Quinn, G. N. Vanderplaats and G. Venter, "VisualDOC: A Software System for General-Purpose Integration and Design Optimization", AIAA 2002-5513

[103] P. Y. Papalambros, "The optimization paradigm in engineering design: promises and challenges", Computer Aided Design 34 (2002= 939-951)

[104] Anthony A. Giunta, Steven F. Wojtkiewicz Jr., and Michael S. Eldred, "Overview of modern design of experiments methods for computational simulations", AIAA 2003-0649

[105] Timothy W. Simpson, Andrew J. Booker, Dipankar Ghosh, Anthony A. Giunta, Patrick N. Koch and Ren-Jye Yang, "Approximation Methods in Multidisciplinary Analysis and Optimization: A Panel Discussion", Approximation Methods Panel that was held at the 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis & Optimization

[106] Benjamin Wilson, David J. Cappelleri, Timothy W. Simpson, and Mary I. Frecker, "EFFICIENT PARETO FRONTIER EXPLORATION USING SURROGATE APPROXIMATIONS", 8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, September 6-8, 2000

[107] Gartmeier, O., "The Role of Structural Optimisation in Automotive Design-Past, Present, and Future", *32nd ISATA proceedings Simulation, Virtual Reality, and Supercomputing, Automotive Applications, 1999.*

[108] P.G. Smith, "Accelerated Product Development: Techniques and Traps", The PDMA Handbook of New Product Developmetn, Second Edition, Kenneeth B. Kahn, Editor, 2004, by John Wiley & Sons

[109] N. P. Repenning, J. D. Sterman, "Nobody Ever Gets Credit for Fixing Problems that Nevel Happened: Creating and Sustaining Process Improvement", California Management Review Vol. 43, No. 4 Summer 2001

[110] Pierre Durand, Michael Wujciak, Fred A. Kuglin, and Colin Wilson, "Creating and Leading the Adaptive Automotive Enterprise - Reshuffling the Global Automotive Leadership", Capgemini Ernst & Young, April 2004

[111] Valerie Belton, "From Design to Decision: An Integrated Approach Linking Discrete Event Simulation, Evolutionary Multi-objective Optimisation and Multicriteria Decision Analysis", Presented at and Published in the Proceedings of the 2nd International Workshop on Simulation (UK Operational Research Society), Birmingham, March 2004

[112] Joseph P. Giesing, Jean-François M. Barthelemy, "A SUMMARY OF INDUSTRY MDO APPLICATIONS AND NEEDS", AIAA White Paper, 1998

[113] K.J. Craigl, Nielen Stander, D.A. Dooge and S. Varadappa, "AUTOMOTIVE CRASHWORTHINESS DESIGN USING RESPONSE SURFACE-BASED VARIABLE SCREENING AND OPTIMIZATION", Engineering Computations, 2004

[114] Atherton Carty, "An Approach to Multidisciplinary Design, Analysis & Optimization For Rapid Conceptual Design", 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, 4-6 September 2002, Atlanta, Georgia, AIAA 2002-5438

[115] J. Bennett, P. Fenyes, W. Haering, M. Neal, "Issues in Industrial Multidisciplinary Optimization", 1998, AIAA 98-4727

[116] Charles D. McAllister, Timothy W. Simpson, Kurt Hacker, and Kemper Lewis, "APPLICATION OF MULTIDISCIPLINARY DESIGN OPTIMIZATION TO RACECAR DESIGN AND ANALYSIS", 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, 4-6 September 2002, Atlanta, Georgia, AIAA 2002-5608

[117] Andreas A. Jobst, "A New Human Resource Strategy Model – How to Reduce Complexity through a Fair Evaluation of the Triangular Relationship between Finite Human Capacity, Technological Progress and Fair Human Resource Management", White Paper, 2000

[118] ArvinMeritor, "A Glossary of Air and Emission Technology Terms", 2001