"Synthesis" and "Computing" in Process Systems Engineering: 45-year Travelogue of an Unindoctrinated Academic

Farewell Lecture

George Stephanopoulos

Department of Chemical Engineering
Massachusetts Institute of Technology

June 2, 2017

Dear colleagues at MIT, colleagues from industry and academia, former students and postdoctoral research associates, personal friends, and family members, thank you for coming to this Farewell Lecture. I am grateful for your presence here today.

Some of you came from places as far away as Australia, Japan, Korea, Hong Kong, Singapore, Argentina, Uruguay, United Arab Emirates, and others from places closer to home, Greece, UK, Turkey, Italy, Switzerland, Norway. Thank you for being here today.

I trust that you all came for reasons which are nobler than the reason, which motivated my good old friend Quique Rotstein, who courageously proclaimed:

"I could not miss the occasion when you will be officially pronounced "old".

One year ago, my first PhD student, Manfred Morari, retired from his professorship at ETH in Zurich. I felt that it was against the laws of academic nature; my academic son retiring before me? What was this world coming to? So, I decided to restore the natural order by retiring myself, and to follow his example by giving a Farewell Lecture.

Farewell lectures are a time-honored tradition in many European Universities and a few American ones. It is an opportunity to recount activities and accomplishments of a professional life; to share lessons learned; to suggest ways forward; to thank all those who have contributed in someone's life in profound ways, or simply an excuse to see long-held friends, students, and colleagues.

They are usually a pleasant blend of silliness and seriousness; expressions of relief from not having to attend more faculty meetings; and sights of trepidation about what the next phase in life has in store for them. In my case they are all of the above.

The subject of my lecture is,

"Synthesis" and "Computing" in Process Systems Engineering.

Within the framework of this subject, I would like to talk about three things:

First, what I have done as an Academic and Professional in Engineering, what were the underlying common threads of my work, and why I would follow a similar path again, if I were start anew.

Second, to give you a brief overview of the path I followed over the last 50 years, a path that shaped my views and outlook as well as the specifics of my work.

Third, extract some lessons and in the form of advice pass them on to my younger colleagues.

Let's start from the end result: I am an engineer, and very proud of it. I love to put things together. I love to solve problems. When I analyze things is for the explicit purpose of using the results of analysis in order to put things together in a better way. I am not interested in the analysis as an end by itself.

My primary mission has been that of an Educator. I have done and still do Academic Research, with the objective to uncover new knowledge that might enhance our abilities to put things together. As an advisor, consultant or manager in industry, again, my role was that of a teacher.

My role as academic advisor and mentor was not to create copies of me but to enable my students to "Write their Own History" in industry or academia; and I was very fortunate to have students who rose to the occasion.

Why did I put Education at the core of all my activities? Because from early on in my life I had subscribed to Plato's axiom:

"All (Forms of) Virtue is One Thing: Knowledge"

Throughout my lecture I will return time and time again to the issue of Knowledge. Please remember that, for me it has always been the central doctrine.

My approach to engineering has been characterized by three common threads:

- The System-View of Anything in Engineering, where the interest is on the behavior of the whole;
- Synthesis as the Core activity of Creative Engineering; and the
- Use of Computers as Information Processing Machines.

Let me say a few words for each one of them.

SYSTEM-VIEW of ANYTHING

Every problem I have been involved with has been a SYSTEM, composed of several (often quite many) interacting components (Figure 1).

- For some systems, like chemical plants, the components are processing systems and the interactions are material and energy flows.
- For other systems, like batch chemical and pharmaceutical plants, the components are Operations and the interactions are again material and energy flows.
- Again for other systems, like living cells, the components are molecules and supramolecular structures, and the interactions are covalent and non-covalent bondings.
- Then again, for other systems like molecules, the components are atoms, or functional groups, and the interactions are bonds, or non-covalent interactions.

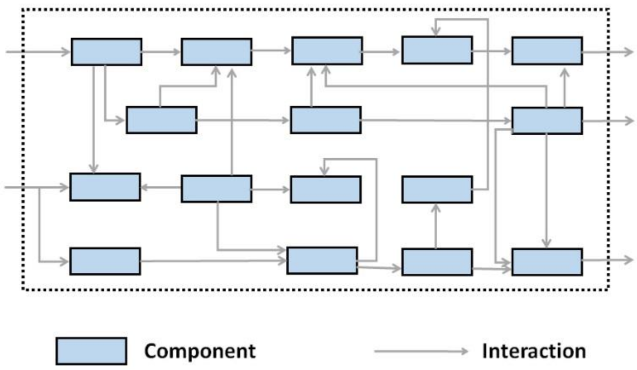


Figure 1. The structure of a system

The "Systems" Approach to Engineering that I have used, is characterized by two aspects:

- The interest has been on the behavior of the whole, while
- The focus of research work has been on studying how the components and their interactions determine the behavior of the whole.

Over the past 45 years I have worked with a very broad range of systems:

- At very large-scale systems, like Complete Industrial Sectors in National Economies such as:
  - Petrochemicals (Argentina, with Quique Rotstein), or
  - Biomass-Based Chemicals/Materials/Fuels (UAE, with Jens Schmidt)
- At meso-scale systems, such as:
  - batch and continuous chemical plants, and
  - their control systems.
- Molecular-scale systems, like
  - chemical and biological networks;
  - molecules as systems of functional groups;
  - products as systems of molecules

I have also worked with systems, which included as components both,

- engineered artifacts and
- humans,

such as those encountered in

- Process Safety, and
- Management of Process Operations.

Depending on the type of system, I have worked on the following engineering tasks:

- How to synthesize the structure of feasible systems,
- How to optimize their design,
- How to operate and control them,
- How to monitor their operational trends, detect abnormal operations, and diagnose the causes of abnormal operations

By studying such a broad variety of systems, I have learn a number of things, which I am struggling to put into a concise and comprehensive textbook on "Process Systems Engineering".

Today, I believe that every interesting engineered artifact is a system. For example: A pharmaceutical is a system composed of an active ingredient, a mixture of excipients, and a delivery mechanism. The LED screen of your modern smart telephone is a system of 8 interacting material-films and processes. If you focus on the individual components, you may never have what you wanted to achieve.

With time the engineered systems have become more complex for three reasons:

- The scope of the systems has been broadened, to include besides function and economics, safety, ecological and other considerations.
- The number of components of the system has increased significantly, and
- The individual components have become more complex in their own behavior.

So, today, the time of low-hanging fruits, i.e. simple systems, is largely gone from advanced economies. The system view is not simply desirable, but absolutely necessary.

There are two types of systems; the Complicated ones, or as the mathematicians call them, the <u>Reducibly Complex</u>, and the <u>Irreducibly Complex</u>. My late friend Giancarlo Rota, a mathematician at MIT, an expert in Combinatorics, taught me the distinction as it is understood by mathematicians: The behavior of the Complicated can be described by large but finite amounts of information. For the Irreducibly Complex, we cannot have all the information we need to describe the system's behavior, which is therefore "Unkownable".

Mathematicians try to find the Simple in the Complex, and the Finite in the Infinite, and thus make descriptions Reducibly Complex. However, the jungle of combinatorial particularities in the complex systems of modern engineering, has put fundamental constraints on the advances that mathematics can achieve. This is where Computers excel. By blazing trails into the zone of combinatorics, and complexity, they can identify "theorems", "Rules", which govern their behavior, and render, seemingly Irreducibly Complex systems to Practically Reducibly Complex.

That is when I appreciated the logic behind Kronecker's famous statement:

> "God created the integers: everything else is man-made".

Most of the systems in chemical engineering are Complicated, and most of my work has dealt with such systems, especially during the period 1965-1985. However, my later work, from 1985 to the present, has dealt with several Irreducibly Complex systems, such as: Inductive reasoning for process hazards identification; Process safety; Modeling Languages, and a Living Cell.

SYNTHESIS OF SYSTEMS

Of all the engineering tasks that I have worked on, SYNTHESIS has been the one that I find to be the most exciting. It is, in my view, at the heart of all creative engineering. Synthesis is the activity that puts together the System. It requires the selection of the components that you will use for the engineered artifact and determination of how they interact with each other.

So, what is Synthesis?

You are given certain inputs e.g.: materials and energy; information, knowledge, which is pertinent to the system you try to synthesize. You specify the desired characteristics of the system (and its outputs) you want to synthesize. You mix all of these in a box, you bless it with your magic wand, and voila you have the engineered artifact you want to produce.

How does Synthesis work?

- You start with the "Problem Statement" (Figure 2). From this you construct an Interim System (Interim Solution); this is a Synthetic activity.
- You proceed to analyze it and identify its strengths and weaknesses. This is the Analysis step.
- From the results of the analysis two things may happen: you synthesize a new Interim Solution, or you reformulate the problem.

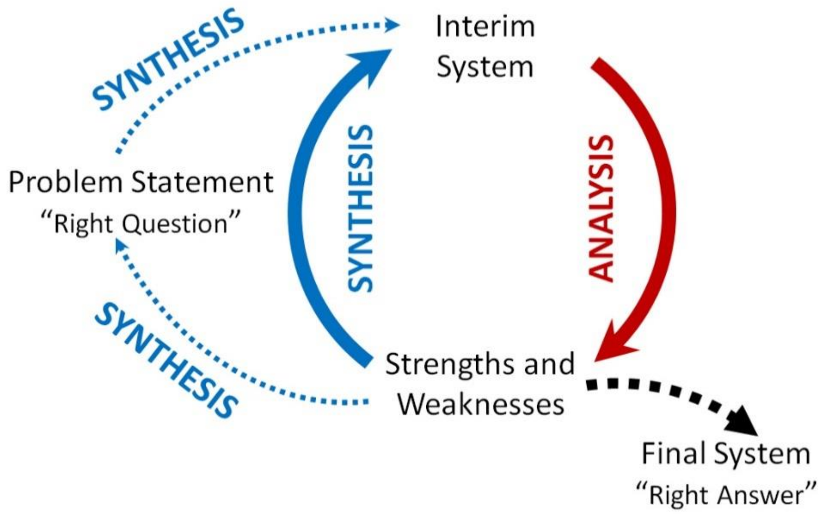


Figure 2. The interplay of Synthesis and Analysis in engineering activities

When does this cycle end? It ends when the cost of running the next cycle exceeds the benefits from the improvements, or simply when you get tired. Or, as my late friend Reuel Shinnar put it:

"Every Engineering Design Activity is a Trade-Off
Between Information and Cost"

Synthesis affects two things; the evolutionary definition of the "Problem Statement", and the "Construction of Interim Solutions". Over the last 40 years I have seen time and time again academic researchers paying very little attention to the Problem Statement. They borrow the Problem Statement from another academic and focus their attention on providing a "Better Solution" to a Potentially Wrong, or at least Weak, Problem Statement. This is not the right approach: to quote a professor from Wharton School,

"the wronger answer to the righter question,
is better than
the righter answer to the wronger question".

Let's consider the Synthesis-Analysis-Synthesis cycle again. Analysis is a Deductive activity. It is constrained by your experimental apparatus or the capabilities of your computer. Synthesis is an Inductive activity. Computers, by their very nature, are incapable of induction. Can we at least formalize a practical approximation of it? Let me paraphrase one of the greats in chemical engineering, a friend and academic mentor, Rutherford Aris of Minnesota:

"What goes on in the designer's head is not purely formalizable,
either in abstract terms.., or in taxonomic views....
It has structure, it has technique that can be taught and learned, but
involves also a personal touch, not only in trivialities but
in deeper considerations of skill and suitability ..."

Understanding this process became my professional life's obsession. I worked in many areas of Synthesis and we learned a lot. They include synthesis of

- Chemical processes; continuous, or batch
- Control structures for complete chemical plants
- Molecules with desired properties
- Closed-Cycle reaction Networks
- Biochemical networks to produce desired chemicals
- Petrochemical sectors for national economies
- Optimal control structures for traffic networks
- Monitoring and Diagnostic systems
- Operating Procedures

Later on I will describe some of our synthesis work in more detail.

COMPUTERS

Computers and programming have been at the center of my academic work for the last 50 years, but my views of the computer and what it can do for process systems engineering have evolved drastically over this period.

As an undergraduate I learned that the computer was a superfast number cruncher. It helped me solve the Navier-Stokes equation with radiative heat transfer and a set of combustion reactions in an air flow field, for my undergraduate diploma research thesis. In the mid-80s, when I returned back to the US and joined MIT, I found myself in the midst of an Artificial Intelligence renaissance. I came to appreciate the computer as a fast and efficient <u>Information Processing</u> machine. It brings together vast amounts of information and processes it very quickly and very efficiently. That changed my views completely.

Why? Let's look again in explicit terms at how an engineer might approach a problem (Figure 3):

- The Engineer defines the boundaries of the problem and sets the objectives, by making assumptions and simplifications.
- Then he/she proceeds to the Problem Formulation. In doing so the engineer requests information from the Knowledge Base of Chemical Engineering Science. At this point, the computer does not simply return facts, but an intelligent discourse takes place between the Computer and the Engineer. For example, the Computer informs the Engineer that there exist; conflicting assumptions or objectives, or missing phenomena for the complete description of the System
- The Engineer makes the adjustments and proceeds with a well-formulated problem for the Analysis.

- At the Analysis step, The Engineer requests facts within the context of the formulated problem. This step involves a certain level of intelligence from the Computer, which "knows" the framework of analysis, as set by the boundaries of problem, the desired objectives, the assumptions, and the simplifications made.
- The Computer supplies the desired facts.
- The Engineer proceeds to the Implementation. To solve the formulated problem, the Engineer requests from the Computer a set of tools, which are appropriate for the task. The Computer makes these tools available and uses them to find the Solution.
- An Iteration back to the beginning may take place, before the final solution is reached. The computer guides the formulation of the iterations.

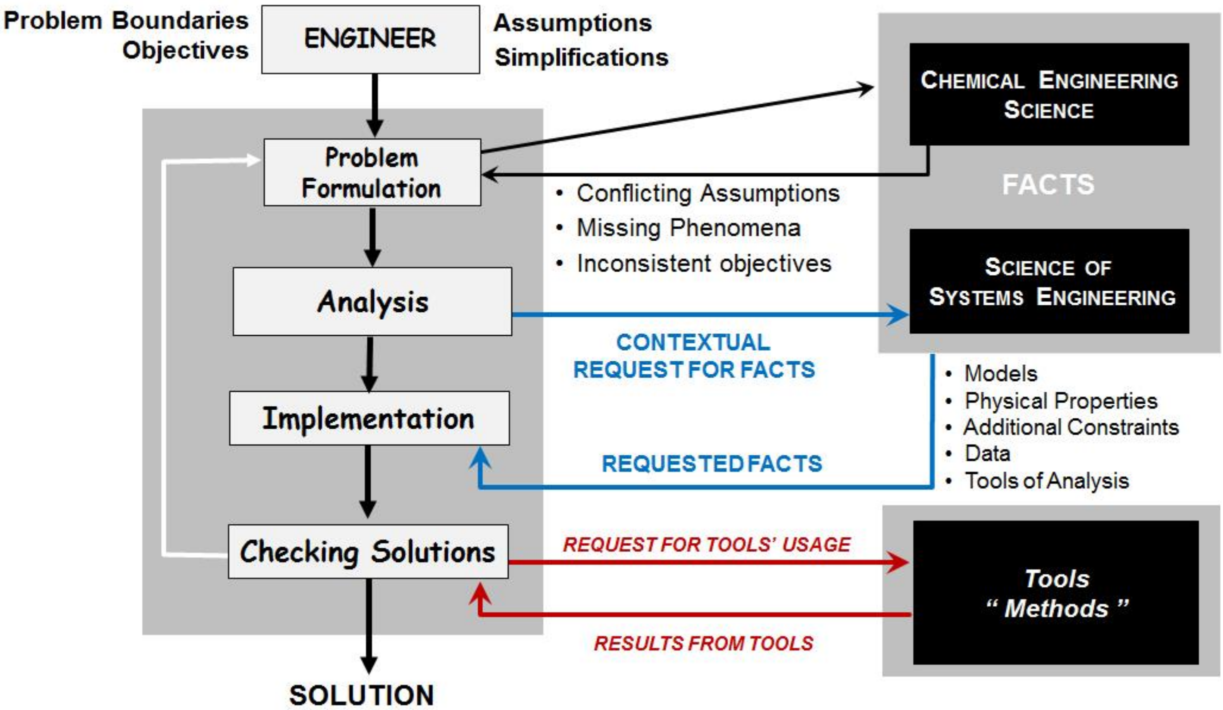


Figure 3. Computers as Information Processors and Servers in Engineering Design

It is clear from my description that to establish such an intelligent discourse between the Engineer and the Computer, one needs a Computer Program with the following features:

- Interacts with the Engineer through human-like channels; drawings, text, speech.
- Possesses rudimentary knowledge of Chemical Engineering Science, akin to that of a BS chemical engineer.
- Can search efficiently through vast amounts of information, and of course
- Carry out complex numerical tasks fast.

Technology to deploy all of these features exists. In fact it existed when we started working on Intelligent Systems at MIT in 1984.

At this point let's remember that our biological/physiological well-being depends on large numbers of microbial populations, and primitive living forms, living within our body, e.g. the microbial populations in our gastro-intestinal track. Let's call the Protozoa (although several of them are higher organisms).

Then, let's note our increasing dependence for higher-level activities, e.g. information acquisition, communications, decision-making, etc., on small digital devices, which have started populating our hands (Apple watch), our pockets (your smart phone), our clothes (wearable digital devices). In analogy with the Protozoa, let's adopt Schwartz's suggestion ("Discrete Thoughts", Katz, Rota, Scwartz) and call these devices, Crystalozoa. As the population of Crystalozoa expands, they create colonies with interacting abilities, and start resembling well-organized and well-coordinated extensions of human intelligence. In my view this is the anticipated trend, and sometime in the near future, the crystalozoa will form complex interacting systems, functioning as "Extensions of Human Intelligence" in solving every-day or more complex engineering problems.

MY EDUCATIONAL AND PROFESSIONAL PATH

Let me now try to describe to you the path I followed and which shaped the views that govern my thinking as an academic and engineer.

The 9-year period, 1965-1974, is the Growing-Up Period, during which I was educated in two general areas of knowledge: The Chemical Engineering Science (Chemistry, Transport, Thermodynamics, Kinetics, etc.), and the Science of Engineering Systems (Modeling and Simulation, Optimization, Dynamics and Control, Queueing Theory, Graph Theory, Convex Analysis, etc.)

National Technical University of Athens (1965-1970)

I entered Chemical Engineering at the National Technical University of Athens, without knowing what exactly the discipline was all about. I am thankful to my "protector angel" for steering me in the right, for me, direction.

A cultural inheritance from the German School of Applied Chemistry, the program was focused on technologies (memorization exercises in explosives, polymers, food, textiles, organic commodity chemicals, inorganic commodity chemicals, metals, ceramics,…).

I was very fortunate to have two teachers, Professors Nicholas Koumoutsos and John Marangozis, who had recently returned to Greece, and introduced me to the beauty of mathematical analysis that was hidden in Transport Phenomena, and Reaction Engineering. Professor Koumoutsos was particularly influential in my plans for the future. He supervised my Diploma Research Thesis, and strongly encouraged me to pursue my dream for graduate studies abroad.

Nicholas Koumoutsos
(1922-2000)

John Marangozis
(1929-2014)

My Diploma Thesis was on modeling and analysis of the combustion of fuel droplets, with the objective to find the operating conditions that optimize performance of internal combustion engines. It was a thesis that involved experiments and simulations. I built the experimental apparatus; a vertical pipe with two circular glass windows where I placed a movie camera, FASTAX, taking 400-500 frames per second. With an electric spark I initiated combustion of a fuel droplet suspended in the air stream and took photographs of its evolving reduction in diameter. I wrote the program that solved the Navier-Stokes equation with chemical kinetics, and convection and radiation energy transport. This is when I discovered that computers were not just number crashers. They could represent "knowledge". To solve the equations, I did not use finite differences, but expressed the unknown solutions as functions, and computed their coefficients. This simple idea, which in retrospect was not so novel, for me it was an eye-opener. The functions were not just numbers, but composite representations of an entity. I will not come back to this idea until 15 years later, when I arrived at MIT and got indoctrinated to the modern concepts of computing, and appreciated the role of computers as information processing machines. The results of my diploma thesis were reported in my first paper: "The Effect of Size and Velocity on the Burning Conditions of Fuel Droplets," N., Koumoutsos and G. Stephanopoulos, *Technica Chronica*, **11**, p. 681 (1970).

This was my first and last foray into experimental research. I also came for the first time into contact with the high performance computer; a CDC 3300, which was housed in its own room, I could not enter. The CDC 3300 had 24-bit architecture, relocation capabilities, and floating point arithmetic. It was designed for scientific computing. It carried 92K instructions per second. Compare it to 350,000 K instructions per second of today's computers. A factor of x3,500 faster.

McMaster University (1970-1971)

At McMaster University I was thrown into a candy store. My M.Eng research thesis advisor, Cam Crowe, was an analyst par excellence. He introduced me into the beautiful world of Pontryagin's Minimum Principle, and Abe Johnson helped me to get a thorough introduction to Large-Scale Process Simulation and Optimization. Both are responsible for introducing me to the beautiful world of Process Systems Engineering, and for discovering Art Westerberg of Chemical Engineering at the University of Florida.

Cam Crowe

Art Westerberg

University of Florida (1971-1974)

At the time, there were several active academic groups in PSE, and in universities with significantly higher ranking than the University of Florida. But, there was something very attractive about Westerberg's work: It dealt with large-scale systems; it captured simulation at an equation-level; it integrated simulation with optimization for large-scale systems in a very natural way (it solved expanded sets of equations); and all his work had mathematical rigor. I became truly hooked and the only thing I wanted to do for my PhD, go to Florida and work with him.

At that time, the department of chemical engineering at the University of Florida, was in an explosive phase of intellectual growth. It had received an NSF Center of Excellence Grant, had attracted a number of first-rate young academics, and was buzzing with excitement.

Art Westerberg had a number of active research areas. Process Synthesis is what attracted me most, especially because we were going to approach the problem as an optimization problem. The Lagrangian 2-Level approach was going to be our approach, because it offered two important features: Allowed explicit decomposability of the structured system, under development, and offered natural upper and lower bounds of the optimal solution: the "primal" and "dual" bounds.

Unfortunately, for certain classes of non-convex problems we had a "dual" gap at the solution. So, there was no saddle point for the Lagrangian and we could not find the solution through the 2-Level Lagrangian approach. This was an opportunity to make a contribution to the optimization theory itself. The

JOTA paper ["The Use of Hestenes' Method of Multipliers to Resolve Dual Gaps in Engineering Systems Optimization," G. Stephanopoulos and A.W. Westerberg, *J. of Optimization Theory and Applications*, **15** (3) p. 285 (1975)] attracted a lot of attention after it was published, but then nothing, until a few years ago, when everybody was interested in the optimization of systems with independent agents. In the last two years alone the number of citations has spiked.

With the Process Synthesis papers we established the "branch-and-bound" strategy as a viable option ["The Use of Hestenes' Method of Multipliers to Resolve Dual Gaps in Engineering Systems Optimization," G. Stephanopoulos and A.W. Westerberg, *J. of Optimization Theory and Applications*, **15** (3) p. 285 (1975)]. This idea was pursued by many researchers later on, and with different methodologies for the computation of the upper and lower bounds. Today, it has become a dominant approach for process synthesis problems.

40 years later with my last research associate Dr. Ana Torres, we would close the cycle by linking this early work to a "Game Theoretical Approach" for the optimal design of multi-actor, distributed processing systems ["Design of Multi-Actor Distributed Processing Systems: A Game-Theoretical Approach", *AIChE Journal,* 62: 3369–3391, 2016].

Questioning the Premises: Recasting Old and Formulating New Problems University of Minnesota (1974-1983)

Joining Minnesota as a faculty member was a very intimidating undertaking. There they were, next door to my office, the demigods of modern chemical engineering: The legendary Neal Amundson, Rutherford Aris, Skip Scriven, Arnie Fredrickson, and many others.

Aris became my role model. Scriven the continuous supplier of unbounded enthusiasm, always prodding for expansive views, and higher challenges. Fredrickson taught me how to teach.

Rutherford Aris
(1929-2005)

Skip Scriven
(1931-2007)

Arnie Fredrickson

This was a department characterized by: Superb scholarship; Healthy balance of engineering and science; Wonderful colleagues with non-stop intellectual stimulation; Unparalleled mentorship.

As I was trying to make sense of Process Design, Control, and Optimization, a couple of major themes stood out:

- The prevailing control theories could not adequately address the reality of process control problems.
- Poor process designs were creating difficult process control problems.

With my first 2 PhD students, Manfred Morari and Yaman Arkun, we started questioning old premises and this reassessment led to recasting old problems and formulating a series of new ones, which would remain in the active research agenda of PSE for a long time, such as:

- The Interaction of Process Design and Control.
- The Synthesis of Control Structures for Complete Chemical Plants.

At Minnesota, in addition to Morari and Arkun, I was very fortunate to have very bright PhD students, like Manfred Morari, Yaman Arkun, Jose Romagnoli, Spyros Svoronos, Tasos Sophos, Tom Bejger, Henry Lau, Jesus Alvarez, and wonderful academic collaborators like Quique Rotstein from Argentina, and Panos Michalopoulos from Civil Engineering. At Minnesota I started collaborating with my brother Greg, who was doing his PhD with Aris and Fredrickson. This collaboration would continue in subsequent years, reach its apex at MIT 15-20 years later, and produce more than 20 co-authored papers.

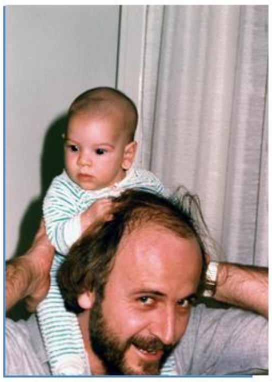
We labored on many interesting problems, and we produced interesting work:

- Synthesis and Analysis of the Argentinian Petrochemical Industry (with Quique Rotstein). This was the first work to address both objectives; economic and environmental impact.
- Synthesis of Reaction Pathways (with Quique Rotstein).
- Analysis and Control of Nonlinear Systems through Bilinear Approximations (with Spyros Svoronos)
- Control of Poymerization Reactors (with Tom Bejger)
- Data Reconciliation (with Jose Romagnoli)
- Variable Measurement Structures (with Jesus Alvarez), etc.
- Optimal structures and policies for traffic networks (with Panos Michalopoulos).

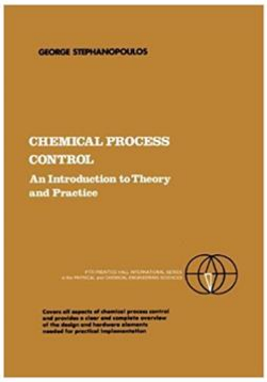
An Interlude: Back Home Enriching my Personal World
National Technical University of Athens 1980-1984

In 1980 I returned as Professor to the place I started where from, the National Technical University of Athens. My tenure at the Polytechnic was short-lived, but very productive in many ways: I met Eleni as soon as I returned and we were married in 1981. Nikos was born in 1982, and along with him came the Colburn Award. My Chemical Process Control book, written largely in Athens, was published in 1983. With my students we carried out an analysis and synthesis of the Greek Petrochemical Sector, producing a book, which helped the country avoid a very costly mistake.

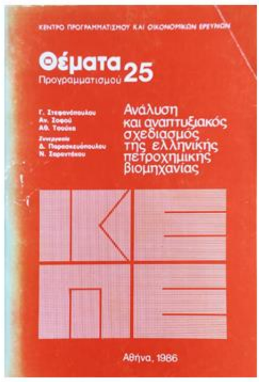
Eleni (1981)



Nikos (1982)



Process Control Book (1983)



Analysis & Synthesis of Greek Petrochemical Sector

At NTUA I was blessed with a group of incredibly talented students. Nearly 20 of the students I taught those years are today in academic positions. Several of them are with us today in this room. We carried out research in (a) the synthesis of reaction networks, (b) design of molecules with desired properties, and (c) design of controllers for structured systems. Many of the ideas generated at NTUA would blossom and expand later on at MIT.

" … The Wonderful Becomes Familiar and the Familiar Fills You With Wonder…"

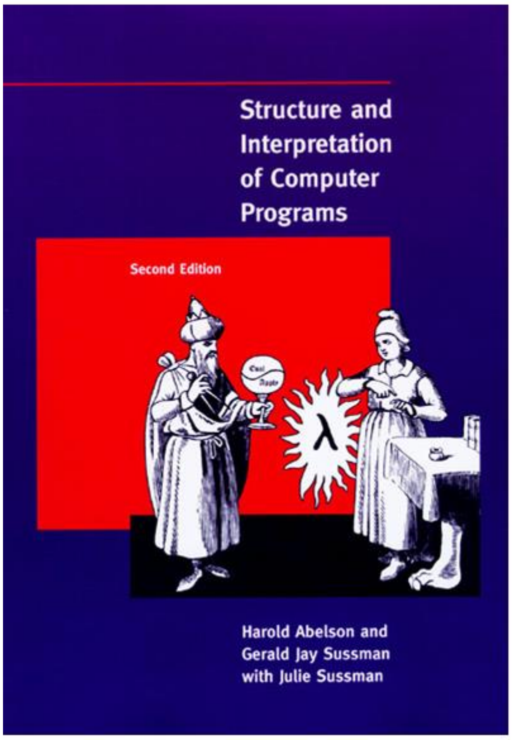
MIT 1984-2000

In January 1984 I arrived at MIT, the birthplace of chemical engineering, and a few months later Elvie was born. The department under the guidance of Jim Wei, it was in the midst of an explosive Renaissance, to recover its premier position among the academics in chemical engineering. Everyone was on a mission. The place was bustling with enthusiasm, intellectual excitement, ambition and determination to be at the top of the world.

The first thing I did was to become a student again. I took the famous 6.001 course (Structure and Interpretation of Computer Programs) taught by Abelson and Sussman, spring 1984. Their course and the accompanying textbook (colloquially referred to as, The Wizard Book), changed completely my views about computers and computing.

It introduced me to the fundamental principles of computer programming, such as: the indistinction between data and procedures, abstraction in programming, modular and object-oriented programming, and other.

Then, I learned about the Lisp Computers, which could do exactly what I was learning in 6.001, and I realized that all the constraints that I had encountered in my previous life to convert my ideas to computer-aided implementations of engineering methodologies, were suddenly disappearing.



The "Wizard Book"

Symbolics 3640

The first commercially available "Workstation"

Two words about the Lisp Computers are in order. Remember, this is 1985, 32 years ago. Symbolics, was bringing to the market the Lisp Computer, which had been developed at MIT's AI Lab, with features that could revolutionize Process Systems Engineering. Indeed, the Symbolics computers, were dedicated personal computers and consoles, not time-shared stations. Hey offered interactive, high-resolution, bit-mapped graphics, and an object-oriented programming environment.

Important features that influenced materially our work:

- Flavors: an object-oriented extension of Lisp, with message-passing among objects, patterned after Xerox's Smalltalk, but with multiple inheritance.
- Object-Oriented database, STATICE.
- The Symbolics Document Examiner, a hypertext system used for on-line manuals was enormously influential in our work.

Immediately we went to work:

- Created LISPE, the "Laboratory for Intelligent Systems in Process Engineering".
- Brought together a fairly large group of brilliant graduate students and postdocs (Table 1).
- Formed a consortium of industrial companies from around the world, and established semiannual symposia and short courses for people from industry (Table 2).

PhD Students		Postdoctoral Associates
Michael Mavrovouniotis '88	Thomas Meadowcroft, '93	Gabriela Henning, 1986-89
John Calandranis, '88	Pedro Saraiva, '93	Horatio Leone, 1986-89
Charles Siletti, '88	Chonghun Han, '94	Andreas Linninger, 1994-97
Kevin Joback, '88	John Carrier, '94	Enrique Salomone, 1994-96
Rama Lakshmanan, '89	Alexandros Koulouris, '95	Jae Hyung Cho, 1997-99
Theodore Kritikos, '91	Christine Ng, '97	Manuel Rodriguez, 1998-99
James Johnston, '91	Shahin Ali, '99	Ajay Modi
Christopher Nagel, '91	John Paul Aumond, '99	
Matthew Realff, '92	Jerry Bieszczad, '99	
Bhavik Bakshi, '92	Matthew Dyer, '00	
Jarvis Cheung, '92	Orhan Karsligil, '00	

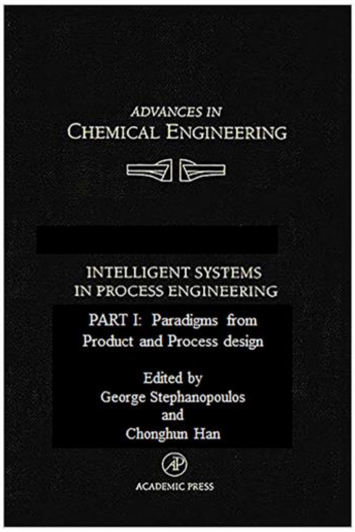
Table 1. PhD Students and Postdoctoral Associates in LISPE (1984-2000)

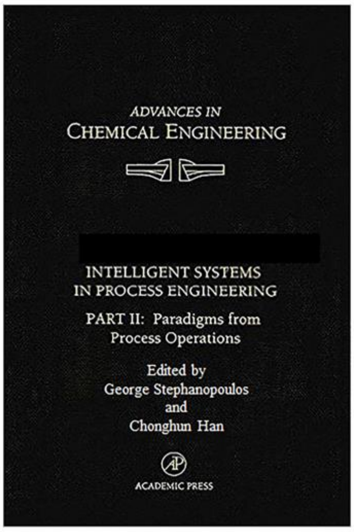
DuPont, US	Honeywell Inc., US	Neste-Oy, Finland
Mitsubishi Chemical, Japan	Amoco Oil Co., US	EXXON, US
Air Products & Chemicals, US	Mobil Res. & Develop., US	Honeywell
ICI Ltd, UK	Shell Development, US	The Foxboro Co, US
Rhone Poulence, France	Texaco, US	DEC, Digital Equip. Co., US
Hoechst, Germany	Japan Energy, Japan	Ryoka Systems Inc., Japan
Union Carbide Corp., US	Koa Oil Co., Japan	Duke Power Co., US
Dow Chemical, Canada	Badger Engineers, US	
Merck, US	Combustion Engineering, US	

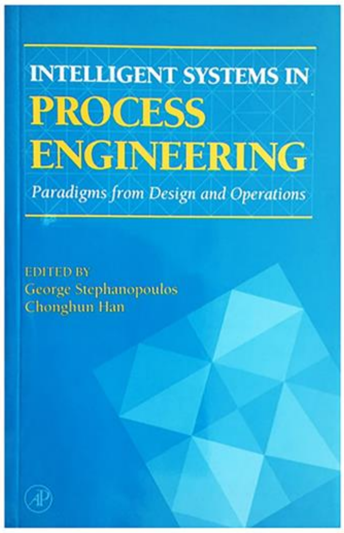
Table 2. The first wave of the LISPE-Industry Consortium companies (1986-90)

We explored many ideas and methodologies from Artificial Intelligence. Did we want to "emulate human intelligence through computers"? Not really, but studying what the researchers in AI were doing we started learning of new ways on how to use the computers for Process Systems Engineering.

Many aspects of our work during this period were put together in two volumes on "Intelligent Systems for Process Engineering".







They provide paradigms on how to use AI concepts and methodologies in addressing various Process Systems Engineering problems.

Part-1: **Paradigms for Product and Process Design**

Modeling Languages:
Declarative and Imperative Descriptions of Chemical Reactions and Processing Systems
Automation in Design:
Conceptual Synthesis of Processing Systems
Symbolic and Quantitative Reasoning:
Synthesis of Reaction Pathways.
Inductive and Deductive Reasoning:
Identifying Potential Hazards in Chemical Processes

Part-2: **Paradigms for Process Operations and Control**

Nonmonotonic Reasoning:
Synthesis Operating Procedures.
Inductive and Analogical Reasoning:
Data-Driven Improvements of Process Operations
Empirical Learning through Neural Neworks: The Wave-Net Solution.
Reasoning in Time:
Modeling, Analysis, and Pattern Recognition of Temporal Process Trends.
Intelligence in Numerical Computing:
Improving Batch Scheduling Algorithms through Explanation-Based Learning.

We also developed a series of computer-aided systems with imbedded logic, some of which are still unparalleled even today, after 20-30 years (Table 3).

Design-Kit	An Object-Oriented Environment for Process Engineering
Model.la	A Modeling Language for Process Engineering
BioSep-Designer	Recovery and Purification of Proteins
Concept-Designer	Synthesis of Conceptual Processing Systems
Molecular-Designer	Design of Molecules with Desired Properties

BatchDesign-Kit	Integrated Environment for the Design of Pharmaceutical Processes
Wave-Net	Multi-Resolution, Hierarchical Neural Network for Localized Learning
Ops-Planner	Non-monotonic Planning of Operating Procedures
Wave-Net	Multi-Resolution, Hierarchical Neural Network for Localized Learning
DataCompressor	Data Compression and Multi-Scale Feature Extraction with Wavelets
Diad-Kit	On-Line Monitoring, Assessment, and Diagnosis of Integrated Boiler Systems

Table 3. Computer-Aided Systems developed in LISPE during the period (1985-95)

Let me give you some examples, which illustrate how research in Artificial Intelligence inspired our work on Intelligent Systems in Process Engineering:

Modeling Languages

Can you have an Engineer interact with a computer, through a high-level language, the way you converse with Siri in your iPhone, and have the computer carry out the tasks the Engineer wants? We developed three languages to address three distinct tasks: **MODEL.LA.**, **LCN**, and the language in **BatchDesign_Kit.** They do not perform just a mechanistic translation. They possess grammar, syntax, and the rich semantics endowed the words with meaning, and thus high-level "understanding" of what is described.

For example in **MODEL.LA.** (designed and implemented by Gabriela Henning and Horatio Leone, and later expanded and enriched by Jerry Bieszcad) the human uses graphical and textual input, to describe a process, and the computer composes the equations of the mathematical model. It is very versatile for a broad range of Process Systems Engineering applications.

LCR (Language for Chemical Reasoning), designed by Chris Nagel, is a Language for the construction of chemical reactions and modeling and reasoning with them. It was used to identify *Inductively* the potential hazards in a chemical plant.

The **BatchDesign_Kit** , developed by Enrique Salomone, Andreas Linninger, Shahin Ali, Kiko Aumond, and Eleni Stephanopoulos, is a comprehensive environment for the development of pharmaceutical processes. It was endowed with a language, which converts the recipes of chemists into batch processing systems with all the ancillary functions; M&E energy balances, costing, wastes treatment, selection of solvents, etc. For example, the following recipe was interpreted and converted automatically into a batch process diagram:

Step-1	CHARGE	ST-100 with 120.5 kg of acetic-acid, with condenser outlet temperature 20 degrees C
Step-2	CHARGE	ST-10l with 204 kg of tetra-hydro-furan, with condenser outlet temperature 20 degrees C
Step-3	CHARGE	ST-101 with 13 kg of potassium-butoxide, with condenser outlet temperature 20 degrees C
Step-4	CHARGE	ST-101 with 23 kg of hydroxamine IV, fh)m Drum-lOl, with condenser outlet

temperature 20 degrees C

Step-5 AGE ST-101 for 1O minutes , with condenser outlet temp of 20 degrees C

Step-6 REACT in ST-100, for 120 minutes, while adding 100 % of ST-101, via reaction RING-CLOSURE

Step-7 CHARGE ST-101 with 37 kg of tetra-hydro-furan, with condenser outlet temperature 20 degrees C

Step-8 CHARGE ST-101 with 37 kg of acetic-acid, with condenser outlet temperature 20 degrees C

Step-9 TRANSFER 100% contents of ST-101 to ST-100, condenser outlet temp 20 degrees C

Step-10 AGE ST-100 for 30 minutes , with condenser outlet temp of 20 degrees C

Step-11 FILTER batch from ST-100, in FI-100, separating solids [100% , potassium-butoxide][100% , NaCl] [100.0wt% , potassium-acetate] as SOUD, lod of cake 30 %, sending mother liquor to ST-102, giving the name Mother-Liquor, operating tirne 240 minutes, with outlet temperature 20

Step-12 WASH CAKE in FI-100 with 10 gallons tetra-hydro-fiiran, sending wash to ST-102, name it Spent-wash, lod of cake 20%, number of wash 1, operating tirne 90 hours per wash

Learning from Experience

Let me give you another example of our work, inspired from computer-generated poetry. Here are 4 stanzas of poetry. Two were written by William Carlos Williams and the other two by a program, the "Kurtzweil Cybernetic Poet".

I am lonely, lonely,
I slap an answer myself
she hides deep within her
yet plays -
Milkless

Was this written by Williams or the computer? What about the following stanza?

Pink confused with white
flowers and flowers reversed
Take and spill the shaded flame
darting it back
Into the lamp's horn

Can you distinguish which were written by the computer? What about the following two?

The days locked in each other's arms
seem still
so that squirrels and colored birds
go about at ease over
the branches and through the air

and

Is a steady burning
The road the battle's fury -

Clouds and ash and waning
Sending out
Young people

Raymond Kurtzweil sampled many humans and found out that about 55% of grownup adults got it right. With children the success rate is 48%. How about you?

How does the program create stanzas of poetry? The program is given an input file with poems written by a human author or authors. Then, it analyzes these poems and creates a word-sequence model based on the poems it has just read. It then writes original stanzas of poetry using the model it has just learned. The "Kurzweil Cybernetic Poet" has created some original word-sequence models from the combination of its experience with poems of T.S. Elliott, William Carlos Williams, and Percy Shelley. Matthew Realff, now professor at Georgia Tech developed for his PhD thesis a program, which monitors and analyses the behavior of branch-and-bound algorithms, while it solves a combinatorial problem on batch scheduling: "Explanation-Based, Machine-Learning Techniques for the Improvement of Branch-and-Bound Algorithms", Realff, M. and Geo. Stephanopoulos, *INFORMS Journal on Computing*, **10**, p. 56-71 (1998). From this experience the program deduces generic rules, which, when used at a subsequent problem, improve the efficiency of the branch-and-bound algorithm.

It also offered a model for integrating "intelligence" into numerical computing (Figure 4). With my brother Gregory and our jointly supervised students, Daehee Hwang (the most prolific student I have supervised), Bill Schmidt and Jatin Misra, we used Pattern Recognition and Machine Learning algorithms, to address a series of questions in Biological Systems, such as these:

- Diagnosis of tissues
    - Clinical testing and diagnosis of pathologies
- Labeling of tissues
    - Identification of "housekeeping genes" for healthy tissues
- Identification of co-regulated or anti-regulated genes
    - Identification of common promoters, transcription factors

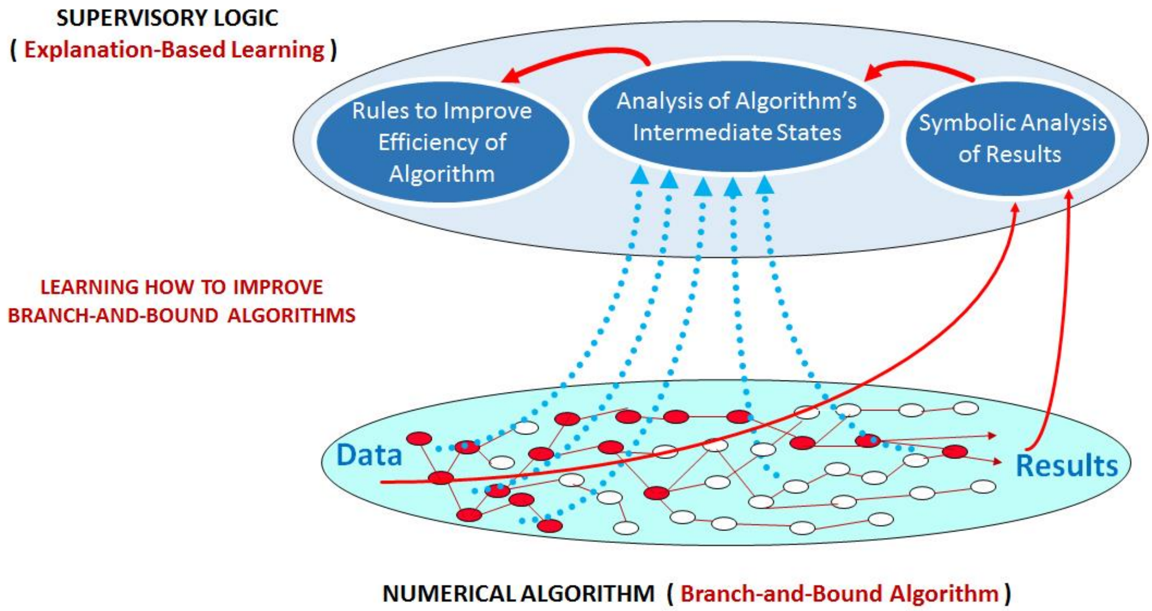


Figure 4. Machine Learning in Scientific Computing

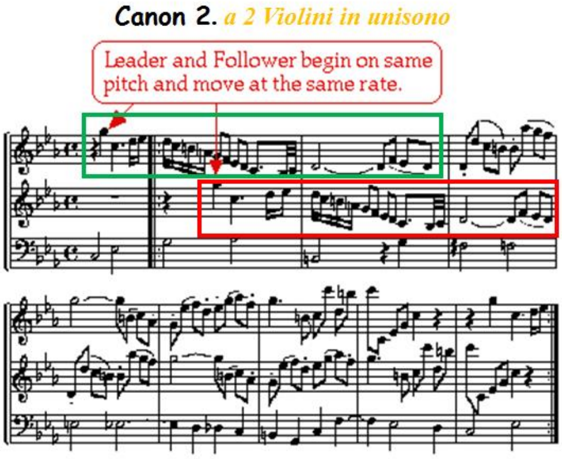
Now let's see an example from computer-composed music. Bach's "Musikalisches Opfer", i.e. Musical Offering, is considered a musical piece of marvelous technical complexity. It is a 6-part fugue, which Bach called it "*Ricercar (to seek) a 6*". https://www.youtube.com/watch?v=3i6MorFy3YE

Bach composed it, using the following elements:

- The "Royale Theme", supplied by Frederic the Great, King of Prussia.

Royal Theme

- 10 canons (rules) for harmonization of the 6 parts (voices). These rules were known and used extensively. Here are two of the Canons (see also Figure 5):
  - Canon-2: Leader and Follower begin on the same pitch and move with the same rate.
  - Canon-8: The Follower plays the mirror image of the Leader.
- Bach also used several rules to evaluate the aesthetics of the harmomonizations sang by several voices at the same time. These rules were also known and taught by Bach and his contemporaries.



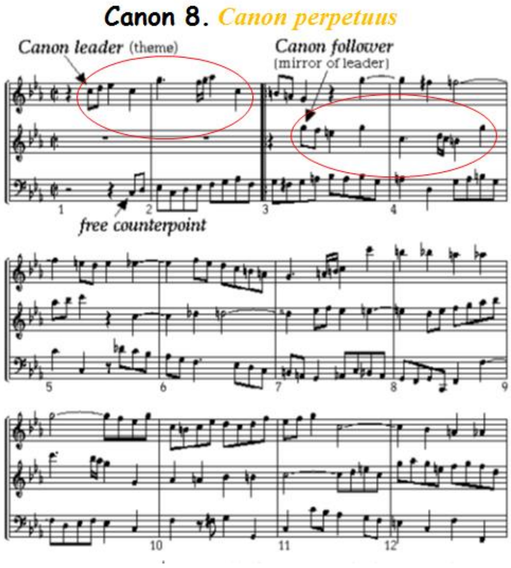


Figure 5. Two Canons used by Bach in *Ricercar a 6*

Could a computer have composed Bach's Musical Offering? Yes, if the computer had, the Royale Theme, the 10 canons for harmonization, and the rules to evaluate the aesthetics of harmonization. Indeed, a program was written to do exactly that. In fact, Kemal Ebcioglu published in 1984 a program for harmonizing chorales in the style of Johann Sebastian Bach. This program managed on several occasions to produce Bach's solutions exactly.

This is remarkable because the lasting value of Bach's particular work is based on his unbelievable ability to synthesize existing knowledge into forms that defied the abilities of his contemporaries.

So, using Canons, i.e. Rules, in specific domains, we started synthesizing systems, which carried out fairly sophisticated Synthesis of original processing systems.

Here are some examples from our work.

- Michael Mavrovouniotis synthesized novel biochemical pathways
- Charlie Siletti created original sequences for the recovery and purification of specific proteins.
- Shahin Ali created batch processing schemes, which were superior to those produced by synthetic chemists at a pharmaceutical company

Now that you have an idea of how we used the computer to emulate aspects of human intelligence and activity, you may not be very impressed, and you may still be skeptical on the emerging capabilities of the computer. That reminds me of how Sherlock Holmes reacted, when a man questioned the brilliance of his deductive reasoning in solving one of his cases:

> "I began to think, Watson," said Holmes, "that I made a mistake in explaining.

' Omne ignatum pro magnifico'

"everything unknown seems magnificent"

Where to Next?

Where do these trends towards smarter and more intelligent software systems lead us? A recent study revealed the following “alarming” prospects:

- 45 percent of work activities could be automated using already demonstrated technology.
- With technologies that process and “understand” natural language an additional 13 percent of work activities could be automated.

So, we are on truck to realize the forecast made in 1853 in United States Review Magazine.

“Machinery will perform all work
automata will direct all activities
and
the only tasks of the human race will be
to make love, study and be happy.

The Most Fascinating Voyage of my Life
Mitsubishi Chemical Corporation
2000-2005

In July 2000 I was invited to assume the position of Chief Technology Officer for the group of companies of Mitsubishi Chemical Corporation (MCC), a company that I had known and interacted with for the previous 10 years. It was a fascinating challenge that I could not pass.

Mr. Eiji Tanaka in his lecture, yesterday, described the structural and cultural transformation we undertook under difficult economic circumstances; the reverberations of the 1997 Asian Financial Crisis were still fresh and fairly strong. So, I will not spend any more time on them. However, I would like to say a few things about the core reason that drove all our work at MCC.

The Reformation of R&D was based on a simple observation, which explained why R&D was not producing new business opportunities:

The Character of the Chemical Industry was changing
from a Process-Centered to a Product-Centered Industry.

In a Process-Centered, like MCC in 2000, the goal of a process is the production of a chemical (well-known and well-characterized) from given and well characterized raw materials. In such case, all the degrees of freedom are in the process itself, i.e. select catalyst, design of unit operations, selection of operating conditions. MCC had excellent, world-class catalysis group with many successes in its history. It also had first-class Process Systems Engineering group. Unfortunately, these tremendous capabilities were not what the company needed, as the market was shifting to a Product-Centered Industry.

As the industry was becoming Product-Centered, the need to anticipate the market needs and the desired characteristics of the products became very pronounced. Close collaboration with the "downstream customer" in defining the characteristics of the desired product, as well as close interaction with the "upstream supplier" were essential in deploying the right R&D Technology Development projects. Therefore, everything that was introduced in the R&D organization in the 5-year period 2000-2005, aimed at transforming the R&D from a process-centered organization to a product-centered one.

Today, MCC is a very strong product-oriented company, and has an R&D organization, which is extrovert, sensitive to the market conditions, and has developed very smooth interactions with suppliers and customers.

Before closing this section, I would like to share with you two important lessons that I learned, which are relevant to both the academic and industrial worlds:

1. <u>The real-world problems are far more exciting than the academic ones in engineering research</u>. Why? When you define your own problem, it invariably ends up being an easier problem than the one someone else has defined for you. Indeed, when engineering problems are defined by industry, they are more representative of the realities on the ground, i.e. include more objectives, their scope is more complex, and usually demand synthesis of many technological components, rather than one. This is a more interesting problem, compared to the, usually, unidimensional problems addressed in academia.
2. <u>Academia-Industry Alliances are Essential for the Creation of Strategic, High-Value Business.</u> Two reasons justify this statement: First, strategic high-value new business, require synthesis of several new ideas, and as such carry with them a higher risk. Universities are a place of much lower cost for reducing the risk, by reducing the extent of the "unknown" factors. Second, the universities are a far richer enterprise in generating, maintaining and distributing knowledge. However, one important note is in order: When I speak of an academia-industry alliance, I refer to a comprehensive collaboration between a company and a university. This is an agreement between two institutions, which marshals the resources of both places towards a specific goal. Such collaboration has nothing in common with the usual single-project interactions between individual professors and companies.

Starting Again

2005-2017

MIT

Returning to MIT was not easy. But, I resolved to spend the rest of my academic career doing research in something that I did not know, in principle, how to do.

I discovered this opportunity in the proposition:

Conceptually Design, Fabricate, and Operate a Molecular Factory

Two potential areas of application were: Artificial Cells, and Nano-scale Factories.

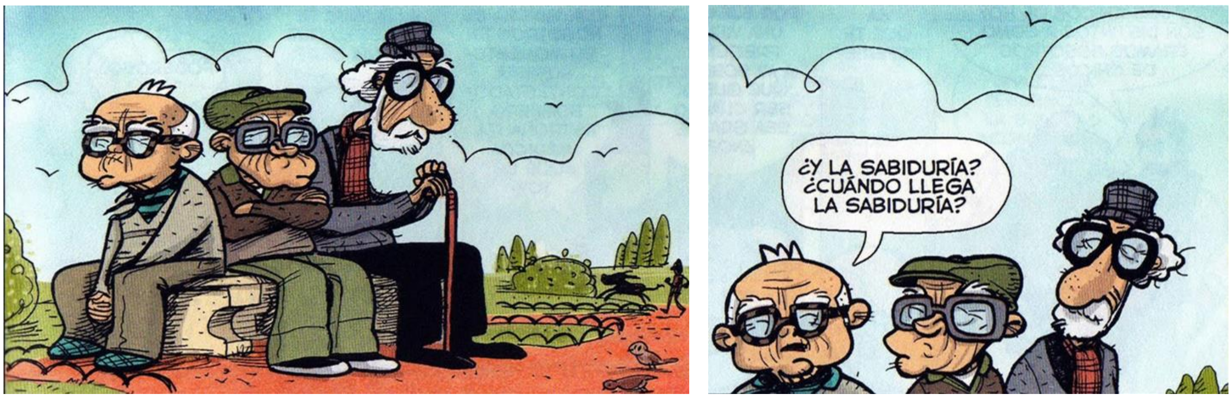
I was fortunate to be guided in this area by my son Nikos, an expert in organic synthesis and nanotechnology, and assisted by the daring and hardwork of two PhD students, Earl Solis and Siva Ramaswami, and a postdoctoral fellow, Richard Lakerveld, presently a professor at Hong Kong University of Science and Technology. The methodologies we had developed for systems at higher-scales, are not applicable at the nano- and molecular-scale (Table 4). A new recasting of the problem was required, and we made significant progress. However, we are still away from a comprehensive solution framework.

	Macro-	Micro-	Nano-
Scale	m	mm-µm	nm
Spatial Considerations	Cost-Safety	Space Utilization Process Efficiency	Overall System Functionality
Materials Properties	Bulk (space averaged)	Bulk (space averaged)	Atomic Molecular
Area/Volume Ratio	Small	Large	Effectively Infinite
Fabrication Philosophy	Top down (man-machine controlled)	Top down (man-machine controlled)	Bottom up Self-Organization
Control Strategy	Centralized Coordinated decentralized	Centralized Coordinated decentralized	Integrated into Unit Design Independent Self-Regulated units

Table 4. Features of Processing Systems at Various Scales

… and the wisdom, when will the wisdom arrive?

Life's trajectory is not linear. It is thankfully quite nonlinear, full of ups and downs, full of new experiences and learning moments. So, now as I reflect on what has transpired in my life over these 45+ years, like the characters in Quino's cartoon, I ask the same question: … and the wisdom, when will the wisdom arrive?



The answer of course has been with me all along. IT HAS BEEN ON THE ROAD I HAVE TRAVELLED ALL THESE YEARS.

What have I learned?

- Continuously learn new things.
- Continuously share these new things.
- Keep asking; you do not really know much.
- Do not fear to undertake new things, if they are interesting and worthy.

And as a teacher and mentor, what did I learn?

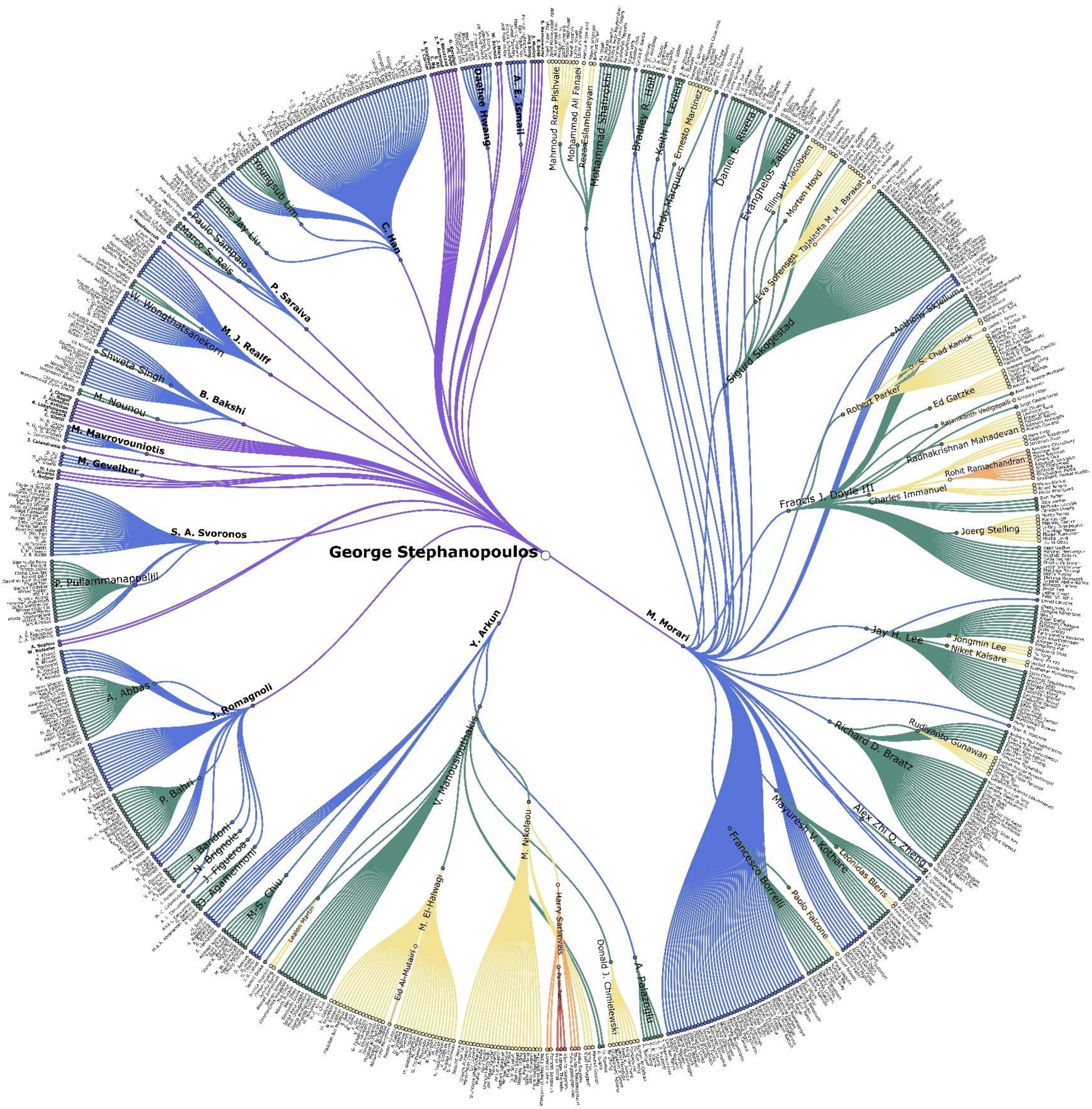
- Your PhD students are your Legacy; Enable them to write their own history.
- Give your PhD student an area to explore NOT a pre-fabricated project.
- Teach Enthusiasm and Challenge your students in Class: They are active participants, not passive audience.
- Resist the Tendency to Render Easy that which cannot become easy without being distorted.

EXPRESSION OF GRATITUDE

I consider myself blessed by having met and worked with so many people who extended their friendship to me.

- My teachers: Koumoutsos, Crowe, Westerberg: I owe them the "good life"
- My mentors: Westerberg, Aris, Scriven, Shinnar, Wei, Brenner. They opened roads for me to travel.
- My colleagues at MIT, who gave me a fantastic place full of intellectual stimulation, support and friendship.
- My academic collaborators: Greg Stephanopoulos, Enrique Rotstein, Panos Michalopoulos, Greg. Rutledge, Kris. Prather, Brad. Olsen, Yuriy Roman, Jens Schmidt, Coleman Brosilow, George Papatheodorou, Alkis Payatakes. My brother Greg has been my closest collaborator with more than 20 joint publications.
- Presidents and Chairs of the Board at Mitsubishi Chemical, who entrusted me with the responsibility of a CTO and Board Member: Miura-san, Shono-san, Tomizawa-san, and Y. Kobayashi-san.
- The members of my leadership team at MCC: H. Kobayashi-san, Nojiri-san, Imanari-san, Shojo-san, Eiji Tanaka-san, Oohta-san, Matsuda-san, Mitsuka-san, and Ihara-san.
- The wonderful leaders of STO (Science and Technology Office) and Corporate STRC (Science and Technology Research Center) at MCC.

- My **Academic Clan**, more than 850-member strong. What can I say? Thank you. Without you nothing in my professional life would have been worth the effort.



- My **Family Clan**: What a sense of security through love they have given me. And as we move to the next generation, I have a tremendous sense of pride for the people they are and the good they spread around them.

- My Parents: for giving me life and what I have become.

Nicholas Stephanopoulos

Elizabeth Stephanopoulos

- My wife and my children. They gave me so much happiness that it is impossible to return.