EURASIP seminar day on radar signal processing.
Hot topics and new trends
September 25th 2009
Dipartimento di Ingegneria dell'Informazione
Pisa, Italy

Complex Systems: an overview

AUTHORS: A. Farina, A. Graziano, E. Spinogatti, R. Di Stefano, S. Giompapa

A. Farina, A. Graziano, Analisi Sistemi Integrati

Pisa – September 25th, 2009

SELEX Sistemi Integrati

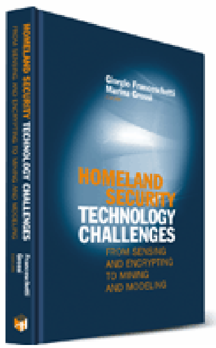
Outline

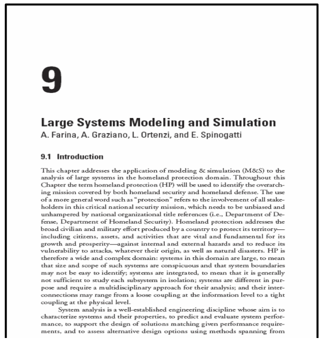
- Introduction
- Complex systems } 1st part
- New engineering approach } 2nd part
- Additional complexity challenges } 3rd part
- Managing complexity } 4th part
- Large and complex system examples:
  - The challenges of software for large systems } 5th part
- Study cases of emerging phenomena: } 6th part
  - Interdependence analysis in large critical infrastructures
  - Domino effect in a large high voltage electric distribution grid
  - The traffic congestion phenomenon in internet-like networks
- General conclusions

GdL ASI FINMECCANICA:

Border Control

2006





9

Large Systems Modeling and Simulation

A. Farina, A. Graziano, L. Ortenzi, and E. Spinogatti

9.1 Introduction

Chapter 9: Large Systems modeling and Simulation

A. Farina, A. Graziano, L. Ortenzi, and E. Spinogatti

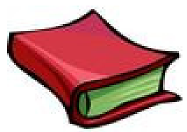
Homeland Security Technology Challenges:

From Sensing and Encrypting to Mining and Modeling

Giorgio Franceschetti and Marina Grossi, 2008

GdL ASI FINMECCANICA:

2007



Big software programs and failures:

some considerations

March 2009, ASI

Acknowledgements

- **Prof F. Zirilli** (University “La Sapienza”, Roma);
- **Prof R. Setola** (Campus Biomedico, Roma);
- **Prof S. Panzieri** (University Roma 3);
- Working Group on the **Analysis of the Integrated Systems**, Finmeccanica.

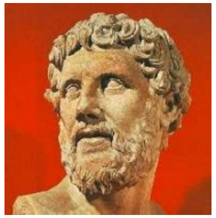
Introduction

- In the recent years the topic of **large/complex systems** has been a subject of great debate and interest;
- **Complexity** is, sometimes, an abused term; other terms would often be more appropriate: large, complicated, garbled (due to poor understanding);
- **Complexity** is often used to express our difficulty and incapacity of managing a situation, our opinion that the problem has exceeded the limit of what can be managed;
- Since 1970's **Complexity in science** has a precise meaning, even though no unique definition exists; definitions tend to differentiate according to specific domains (e.g. physics, biology, engineering, software, social sciences);
- A consensus is being gained on the main characteristics which complex systems share.

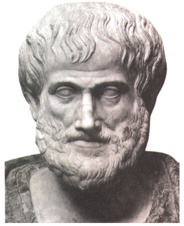
Outline

- Introduction
- Complex systems
- New engineering approach
- Additional complexity challenges
- Managing complexity
- Large and complex system examples
  - The challenges of software for large systems
- Study cases of emerging phenomena:
  - Interdependence analysis in large critical infrastructures
  - Domino effect in a large high voltage electric distribution grid
  - The traffic congestion phenomenon in internet-like networks
- General conclusions

Reductionism vs. holistic approach



Democritus

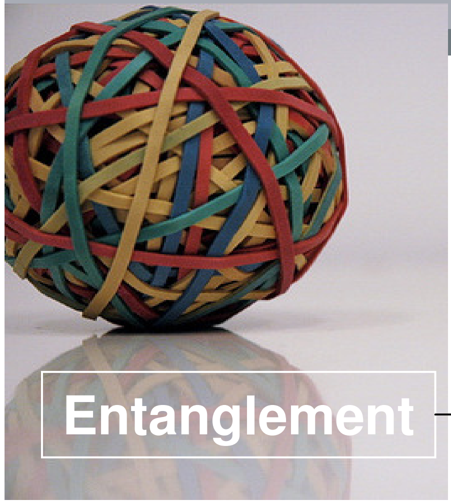


Aristotle

- “The atom is something which cannot be further decomposed”, Democritus (460 B.C. – 370 B.C.).
- “The whole is something over and above its parts, and not just the sum of them all”, Aristotle: The politics (350 B.C.).

- Atomic physics and molecular biology - the superstars of 20th century science – have both shared the searching for the constituents of the organized whole → **Reductionism approach.**
- The wonderful elementary laws (e.g.: atom spin up & spin down) of physics are **not capable of explaining emerging complexity** when a huge number of elementary systems interact together.
- In molecular biology one of the latest great achievement is the mapping of human genome; however this knowledge has given us very little insight about the causal chains that link genes to the morphological and other phenotypic traits of organisms.
- **How to use the information we capture on isolated elements to build a theory of “the whole”?**
- The “take home message” of the lessons from the history of science is that **methodological reductionism**, the analytical decomposition of structures to parts, **should be completed by searching for organizational/holistic principles**, too.

From simple elements to complexity



Greek → *plektos*

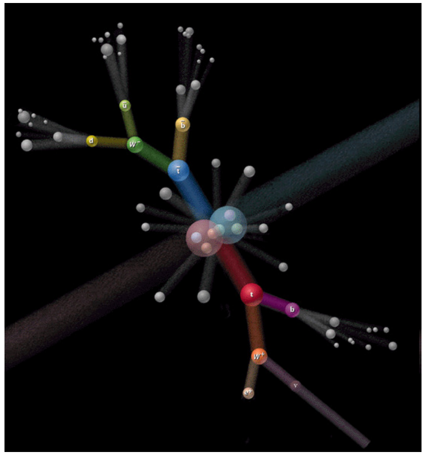
Latin → *plexus*

→ *simplex* → "once" as in "simple"

→ *complexus* → "together" as in "complex"

Entanglement → is a key feature of the way complexity arises out of simplicity

QUARK: the smallest "observed" element in nature.
All human beings and all "matter" as we know it (*) are essentially bundles of simple elements.
It is the **entanglement of the states of the particles** that is responsible for "matter".



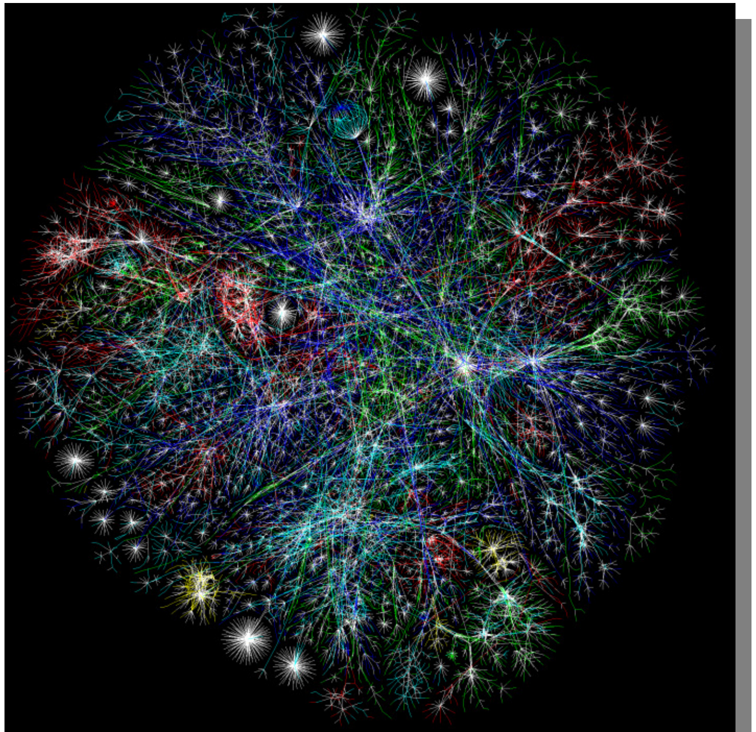
M. Gell-Mann: Nobel laureate for discovering the quark

(*) Let's call it Plectics (Complexity, Vol. 1, no. 5, © 1995/96).
Murray Gell-Mann, "The Quark and the Jaguar: Adventures in the Simple and the Complex", 1994.

Definition of “Complex System”

- Complex systems contain many constituents;
- the constituents of a complex system are interdependent and interact non-linearly;
- a complex system possesses a structure spanning several spatial scales;
- a complex system is capable of emerging behavior (i.e. a self-organizing collective behaviour difficult to anticipate from a knowledge of agents’ behaviour)

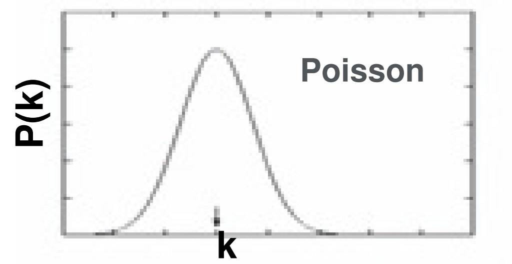
Birds adapt to neighbouring behaviour and automatically form a flock



Internet map (source “The Opte Project”, www.opte.org)

Statistical features of complex systems

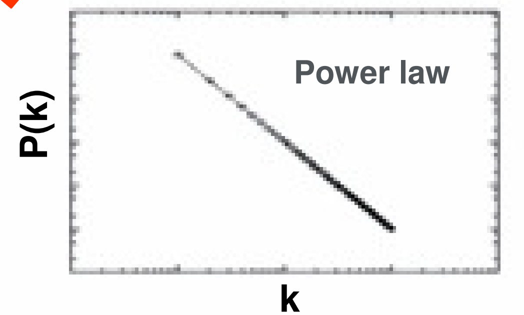
- Bell shaped distributions (e.g. Gaussian, Poisson) have been applied widely.



Poisson distribution ("democratic") applied to study the phone network (voice) in Teletraffic Theory .

Power law distribution ("aristocratic") introduced to study the new network (data) in Internet Engineering

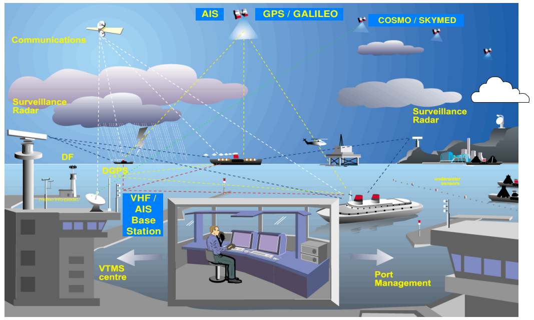
- Such kinds of phenomena, which don't really have a characteristic size (e.g.: mean), are described by an asymmetric (skew) so called power law distribution.
- Ubiquity of power law in nature and artificial made world.



Taxonomy of systems

- Simple
  - They have a **small** number of components which have well-defined roles and are governed by well understood rules;

- Complicated
  - They have a **large** number of components which have well-defined roles and are governed by well understood rules;
  - Robustness is achieved through redundancy;



VTS system

AM 346

- Complex
  - They typically have a large number of **similar** components which may act according to rules that may change over time and that may not be well understood;
  - The **connectivity** of the components may be quite plastic and roles may be fluid;
  - **Robustness** is achieved by enabling the parts to adapt to the changing environment and adopt different roles;
  - Need to distinguish between **complex system** and **complex dynamics** (complex time behaviour may arise from simple systems).

Note that in relation to the scale of observation the same system may be classified differently!

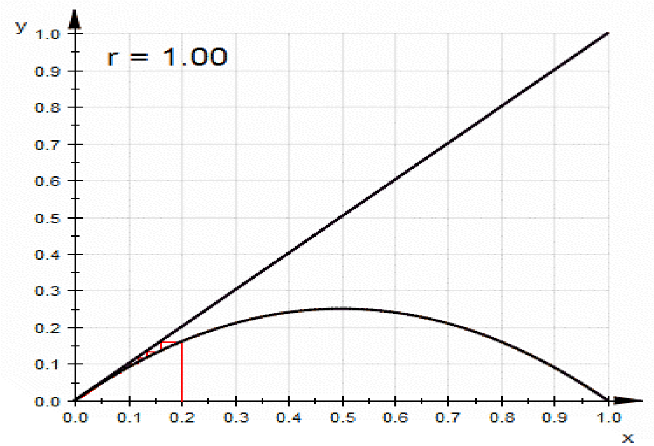
Complex dynamics in simple systems

An example of how very simple models may exhibit chaotic instability for some ranges of parameter values

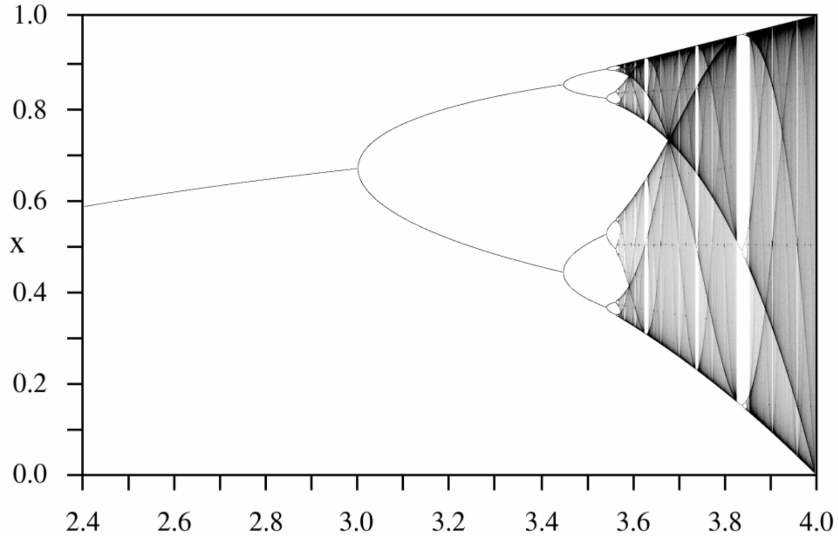
The *logistic map* is a dynamic non linear demographic model of the population size in a resource limited environment.

$$x_{n+1} = r x_n (1 - x_n)$$

where x_n is a number between zero and one, and represents the population at year n, and hence x_0 represents the initial population (at year 0), and r is a positive number, and represents a combined rate for reproduction and starvation.



A cobweb diagram of the logistic map, showing chaotic behaviour for most values of r > 3.57.



Possible long-term values of x (vertical axis) versus r (horizontal axis).

V. Volterra, an Italian mathematician and physicist, best known for his contributions to mathematical biology.

Simple models may hide unexpected behaviours!

Langton’s Ant

Another example where very simple rules produce seemingly unpredictable results→ potential unpredictable behaviour of software programs

C. Langton, one of the founders of artificial life algorithms (1987), Santa Fe Institute.

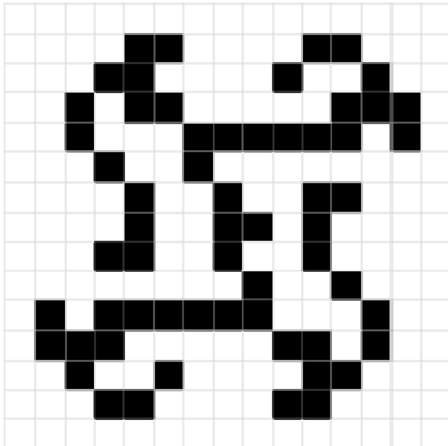
Langton's ant is a two-dimensional Turing machine with a very simple set of rules but complicated emergent behavior.

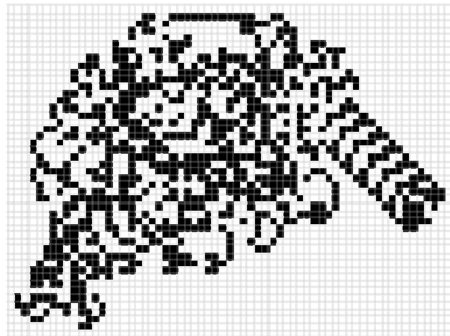
The ant starts out on a grid containing black and white cells, and then follows the following set of rules.

1. If the ant is on a black square, it turns right and moves forward one unit.

2. If the ant is on a white square, it turns left and moves forward one unit.

3. When the ant leaves a square, it inverts the color.



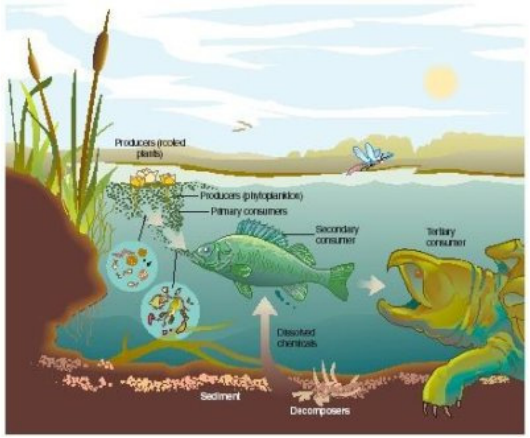


When the ant is started on an empty grid, **it eventually builds a "highway"** that is a series of 104 steps that repeat indefinitely, each time displacing the ant two pixels vertically and horizontally. The plots above show the ant starting from a completely white grid after 386 (left figure) and 10647 (right figure) steps. The fact that the ant's path is unbounded is guaranteed by the Cohen-Kung theorem. It is believed that no matter what initial pattern the ant is started on, it will eventually build a highway (although it might in principle take an extremely long time to reach this point).
This would appear to follow naturally from the fact that Langton's ant is reversible, although it remains formally unproved (Beermann and Van Foeken).

Ant Movie

Taxonomy of complex systems

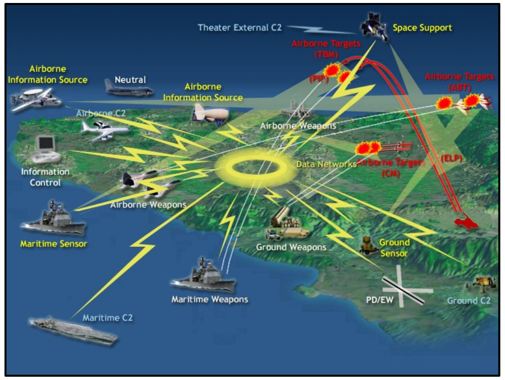
- Complex Non-adaptive Systems:
  - those that are based on models comprising the immutable equations of physics (e.g. weather forecasting) or where rules of behaviour are hard-coded into genes (e.g. micro-organisms and ecosystems);
  - behaviour may adapt to circumstances, but it does so in a predictable way, which is also encoded in their genes.
  - they offer the promise of predictability.



Ecosystem

- Complex Adaptive Systems:
  - those that are based on models and agents (e.g.: people) that can adopt new rules when circumstances change (e.g. financial markets);
  - Understanding the behaviour of a complex system necessitates a simultaneous understanding of the environment of the system;
  - conjectures are translated in phenomenological equations;
  - these need to be complemented by a model of interactions between systems.

Human brain



NCW

Woods, J. D. (2003), "Primitive equation modeling of plankton ecosystems", in: N. Pinardi, & J. D. Woods (Eds.), Chapter 18 in Ocean modeling: Conceptual basis and applications. Berlin: Springer Verlag. 472 pp.

Outline

- Introduction
- Complex systems
- New engineering approach
- Additional complexity challenges
- Managing complexity
- Large and complex system examples
  - The challenges of software for large systems
- Study cases of emerging phenomena:
  - Interdependence analysis in large critical infrastructures
  - Domino effect in a large high voltage electric distribution grid
  - The traffic congestion phenomenon in internet-like networks
- General conclusions

The new approach of engineering

- Traditional engineering tries to decouple the components of a system in order to avoid the behaviour of the complex group, i.e. to avoid the effects of the complexity (divide et impera).

- The new trend of engineering forces and encourages us to design and exploit complexity to obtain **additional performance and robustness** for the systems.
  - need to develop an adequate theory to study the intrinsic properties of complex systems (e.g.: stability, controllability, self-organization, emergent behaviour, etc.).

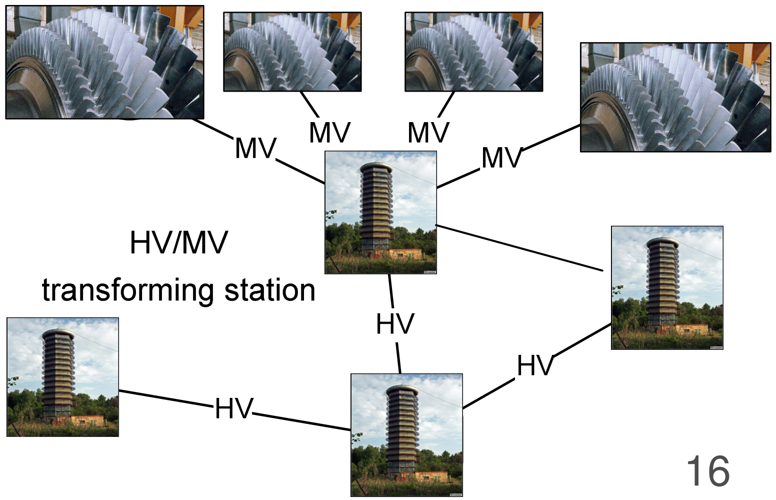
“Primitive technologies build fragile systems from precision components; advanced technologies build robust systems from many sloppy components.”

The new approach of engineering (cont)

Challenging shift: from an orchestra led by a director to an orchestra without director;

Traditional approach, an example: each electric generator in the power grid is forced to maintain synchronization on a prescribed fixed electrical phase;

New engineering: the network of power stations in a grid self-synchronizes on the same electrical phase.

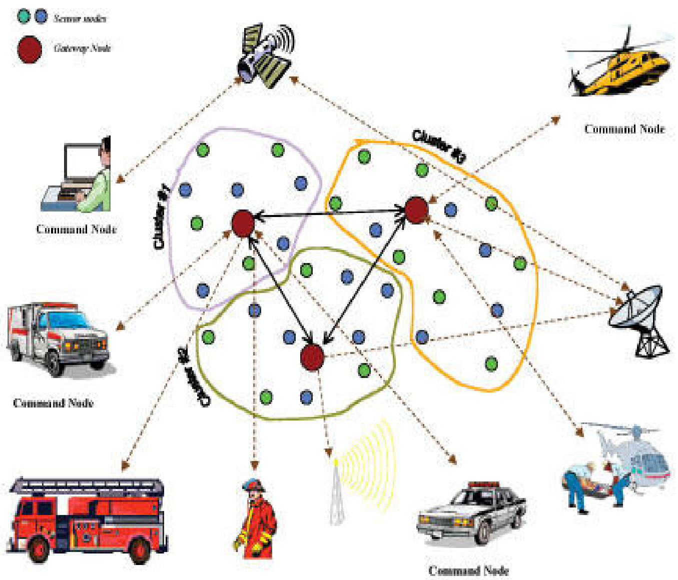


Metronomes Movie

Surveillance system via the new engineering

SELEX Sistemi Integrati

<u>System:</u> Wireless sensor network *without a global fusion centre*.
<u>Purpose:</u> to achieve a global detection/estimation/localisation through localized processing so that the final estimation/decision value is available for each node.
<u>Advantages:</u> strongly advocated approach for its robustness and ease of implementation.
<u>Broad range of potential applications:</u> homeland security, surveillance of habitat and environmental monitoring, seismic monitoring, health monitoring, structural monitoring (e.g.: bridges), contaminants, smart roads, intruders detection, battlefield.



Sensors, considered as non linear coupled oscillators, will synchronise via local exchange of measurements. Synchronisation in this case means global estimation, detection, localisation.

MOTE

Complementary to the classical surveillance with few large/costly sensors hierarchically organised.

Outline

- Introduction
- Complex systems
- New engineering approach
- Additional complexity challenges
- Managing complexity
- Large and complex system examples
  - The challenges of software for large systems
- Study cases of emerging phenomena:
  - Interdependence analysis in large critical infrastructures
  - Domino effect in a large high voltage electric distribution grid
  - The traffic congestion phenomenon in internet-like networks
- General conclusions

Additional challenges: the cardinality

- Yet we often are frustrated and defeated by apparently more conventional problems, such as the increase in cardinality (e.g. handling large numbers of system components, of subcontractors). →

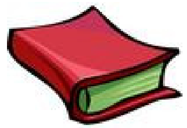
  If many companies can handle small projects, few companies can handle large projects where such numbers might become very huge.

- *A simple example:*
  - If it is only two people working on the same document, everything may go pretty smoothly;
  - If it is many of us working on the same document, the affair may quickly become a nightmare unless a **process** is set up and adequate tools supporting the process are available.
- In the systems we deliver, numbers (e.g. the number of stakeholders, of subcontractors, of systems parts) are growing all the time!

The kind of "complexity" we experience every day is apparently more conventional, yet its effects can be very nasty unless proper measures are taken.

Additional challenges: the many facets of modern large systems

- New operational paradigms, e.g. Network Centric Operations (NCO), Effects Based Operations (EBO);
- System agility: be effective across multiple scenarios;
- Systems are globally distributed;
- Modern systems are extensively software based and software has its specific criticalities;
- Cyber threats impose additional stringent requirements on software and networks;
- The merit of a system must be assessed against several measures of effectiveness (MOEs);
- Need to explore and analyze large multidimensional trade spaces;
- Predict performance across a multitude of design and technology options.



Cyber threat: a preliminary survey

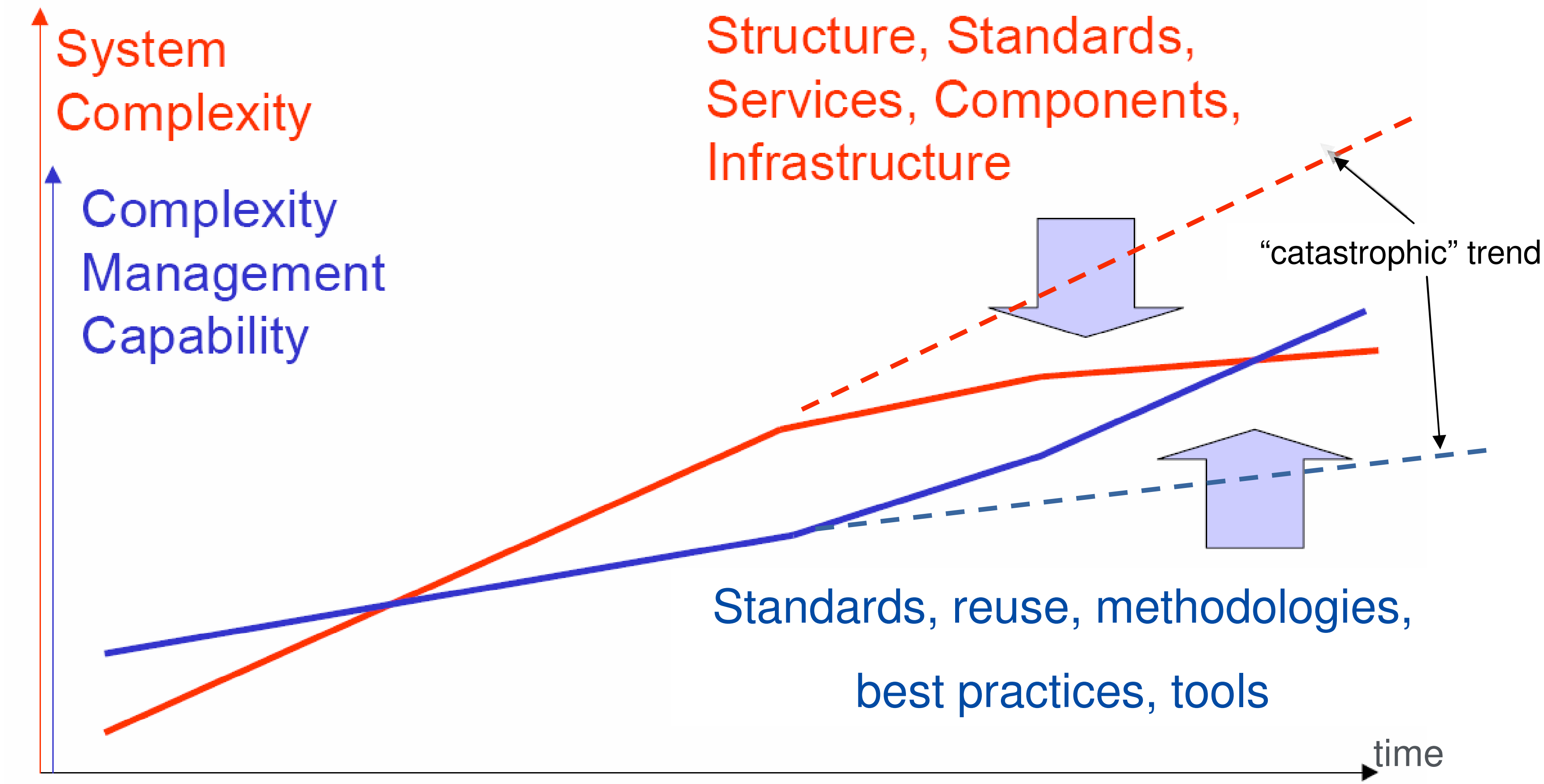
April 2009, ASI

Outline

- Introduction
- Complex systems
- New engineering approach
- Additional complexity challenges
- Managing complexity
- Large and complex system examples
  - The challenges of software for large systems
- Study cases of emerging phenomena:
  - Interdependence analysis in large critical infrastructures
  - Domino effect in a large high voltage electric distribution grid
  - The traffic congestion phenomenon in internet-like networks
- General conclusions

Complexity gap and remedies

Accepting the previous challenges means having to deal with increasing system complexity.

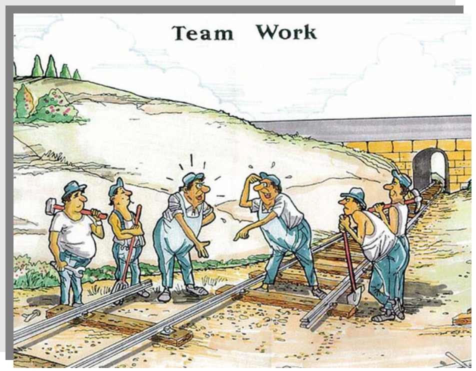


"Model driven architecture for C4I interoperability", FMV [http://www.fmv.se/]

When the gap between system complexity and complexity management capability widens, countermeasures are needed to avoid catastrophe.

Complexity management capability: main factors

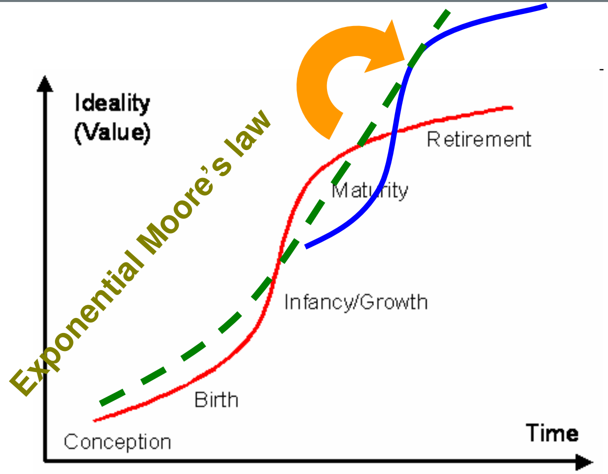
- The right people with the right skills in the right jobs;
- The importance of project and risk management: not always the risks of projects are properly perceived and hence they are not properly managed; master WBS for complex systems;
- Customer: collaboration, management and education;
- Collective professionalism & collaborative working: tools, processes and education;
- Work Group computing: messaging/chat, E-mail, multiple agenda management, task scheduling/allocation/reporting, fast file sharing, distributed database for collaborative developments, networking of thousands work stations for collaborative working models;

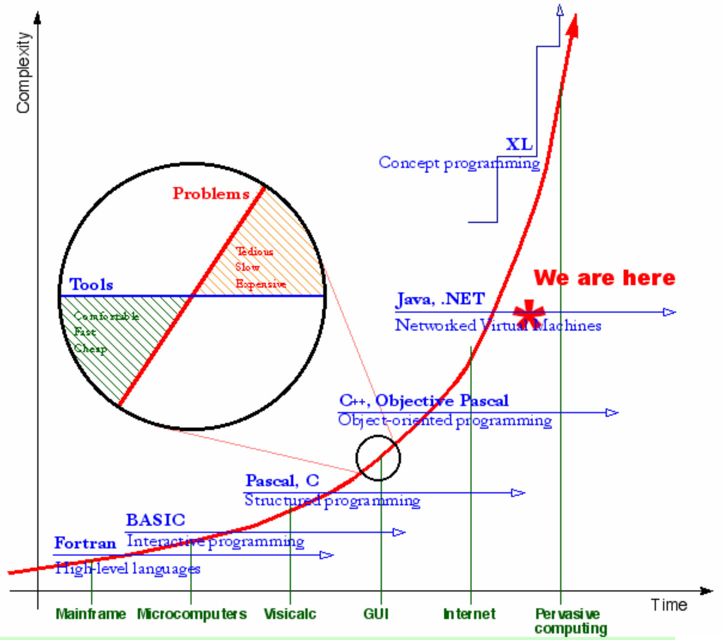


- “The System Architect in complex IT systems is the person who can look at a jewel and hit it just the right way so it falls into the right number of pieces. It is that ability to decompose in just the right way" (H. Lilleniit)”;
- Modularity: even though modularity is not a new concept in system engineering, a systematic approach to design “good” modularity systems is still missing;
- Spiral development, Integration and testing, reuse,…;
- Analysis, Modelling & Simulation: fidelity vs complexity, multiresolution and variable granularity models.

Complexity management capability: leveraging new technologies

- The traditional way to bridge a gap is to regularly invent new so-called **paradigms**, yielding repeated discontinuous **jumps** in technology, at the expense of most of the existing product base.
- Technologies develop across a contour that is the well known "**S-curve**"; this makes possible to **profitably manage the jumps** between them [1] → Moore's law.
- A possible way to manage the exponential trend is to resort to new paradigms; e.g. in the case of software new paradigms usually allows the developer to write code at a higher conceptual level (minimize translation loss).

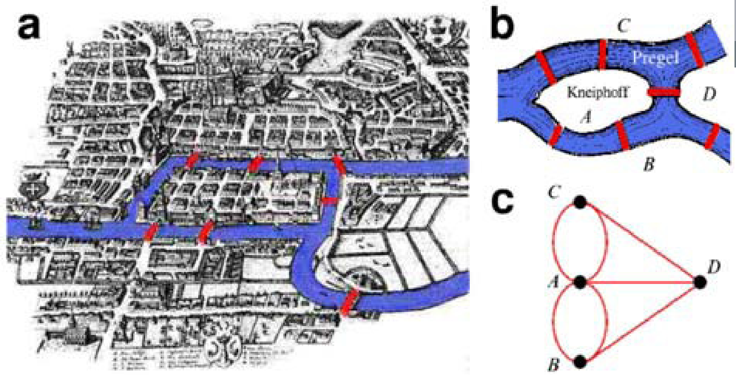




[1] Jumping the technology S-curve, IEEE Spectrum, June 1995

Successful companies are those that switch from one S-curve to the other at the right moment.

Topology of graphs as the model of interconnectivity

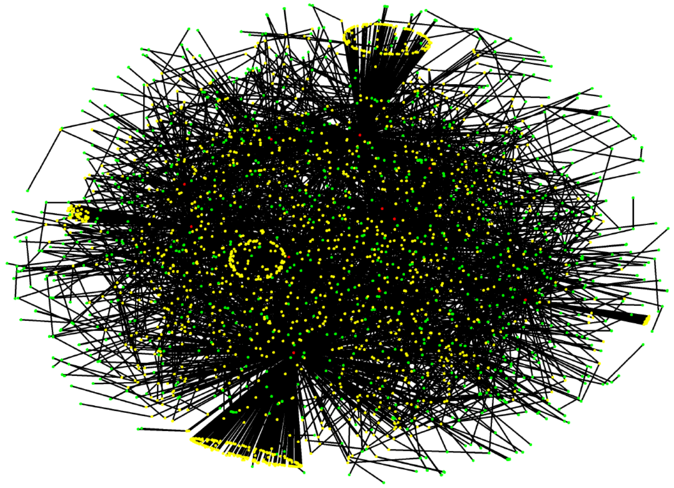


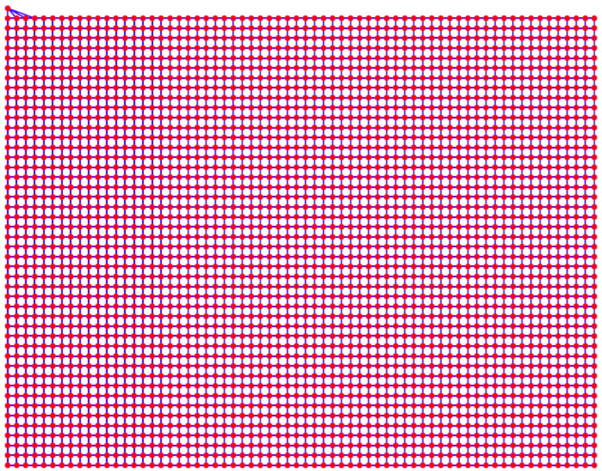
L. Euler and the conundrum of the bridges of Königsberg (1736)

- Interaction among elements is fundamental in complex systems;
- Graphs capture the pattern of interactions whose topological properties are fundamental for the emergence of collective behaviours;
- Many systems, which apparently have very little in common, share similar topologies;
- Topologies: lattice/regular, random, small world, scale free,…
- Topological parameters: node degree distribution, clustering, path lengths, diameter, …;
- The spreading dynamic on a network is conditioned by its topology;
- The new networks (small world, scale free), which are emerging by the globalisation and advanced technology (transportation, internet), speed up the spreading phenomena (e.g.: virus, data exchange);
- Difficult equation to solve: information sharing vs. information security.

An example of plecticity measure

- Comparison of plecticity between Inet 3037 (internet topology generator(*)) network and rectangular lattice.





Inet 3037

Node number: 3037

Branch number: 4788

Rectangular lattice

Node number: 3037

Branch number: 5964

The node degree distribution follows a power law

(*) S. Jamin, J. Winick: Inet-3.0: Internet topology generator, Technical Report CSETR-456-02, Electrical Engineering & Computer Science EECS Department, University of Michigan, 2002, http://topology.eecs.umich.edu/inet/inet-3.0.pdf.

An example of plecticity measure (cont'd)

- Betweenness centrality (BC) of a node v: average of the percentages of the minimal paths which link all the pairs of nodes, *s* and *d*, in the network graph, and which cross the node *v*.
- Mean betweenness centrality of the graph: betweeness centrality averaged over all the nodes *v* in the graph.

- A possible definition for the plecticity of the graph is the ratio between the maximum value of the BC (that provides the most important hub of the graph) and the mean value of BC:

$$Plecticity = \frac{maximal\ BC}{mean\ BC}$$

i.e. plecticity grows as larger hubs appear in the network.

	Inet 3037	Rectangular lattice
Plecticity	**$0.46 \cdot 10^3$**	**2**

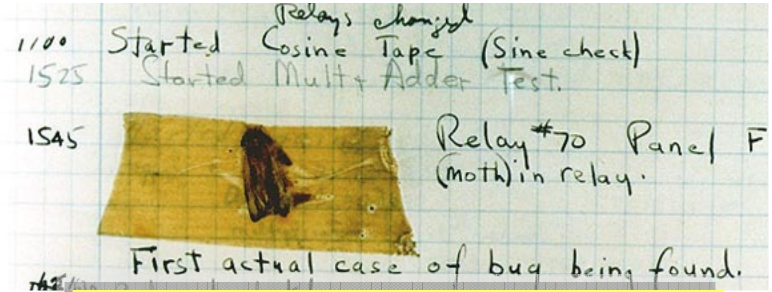
Outline

- Introduction
- Complex systems
- New engineering approach
- Additional complexity challenges
- Managing complexity
- Large and complex system examples
  - The challenges of software for large systems
- Study cases of emerging phenomena:
  - Interdependence analysis in large critical infrastructures
  - Domino effect in a large high voltage electric distribution grid
  - The traffic congestion phenomenon in internet-like networks
- General conclusions

The challenges of software for large systems

Modern systems are extensively software based and software has its specific criticalities.

- **Conventional challenges**: it is amongst the most labour-intensive and **error-prone (conceptual, maths, logic, syntax, resource, co-programming, team working, etc)** technologies in human history.
  - Known answers (CMM of SEI, testing, "divide et impera",….) to manage conventional challenges.



The first bug

- **Non conventional challenges**: ultra large scale systems, emergent behaviours
  - As the size of software modules increases (estimate of US Army's FCS software code has tripled to 95.1 MLOC from the original estimate made in 2003), bug density increases too !
  - Do we know the right answers ?

CMM: Capability Maturity Model,
FCS: Future Combat Systems ,
MLOC: Million Lines Of Code,
SEI: Software Engineering Institute

Software bugs cost to the US economy an estimated $59 billion annually, or about 0.6 percent of the gross domestic product.

Horror bugs

The Internet is full of software horror stories
http://www.cs.tau.ac.il/~nachumd/horror.html

- 1996, European Ariane 5 rocket veers off its path and is self-destructed, the cause is result of code reuse from Ariane 4 which has very different flight conditions, more than $370 million were lost !!!
- 1999, Mars Climate Orbiter crash, a NASA subcontractor used Imperial units (pound-seconds) instead of the metric system, more than 125 M$ loss! "*The problem here was not the error, it was the failure of NASA's systems engineering, and the checks and balances in our processes to detect the error.*" (Edward Weiler, NASA's Associate Administrator for Space Science);
- 2004, Los Angeles Airport, air traffic controllers suddenly loose voice contact with 400 airplanes, a countdown timer runs out of ticks (milliseconds) after 2^{32} (≈ 50 days).

One single detail can wreak havoc on the entire project !!

Software carries the scars but system engineering is often to be blamed.

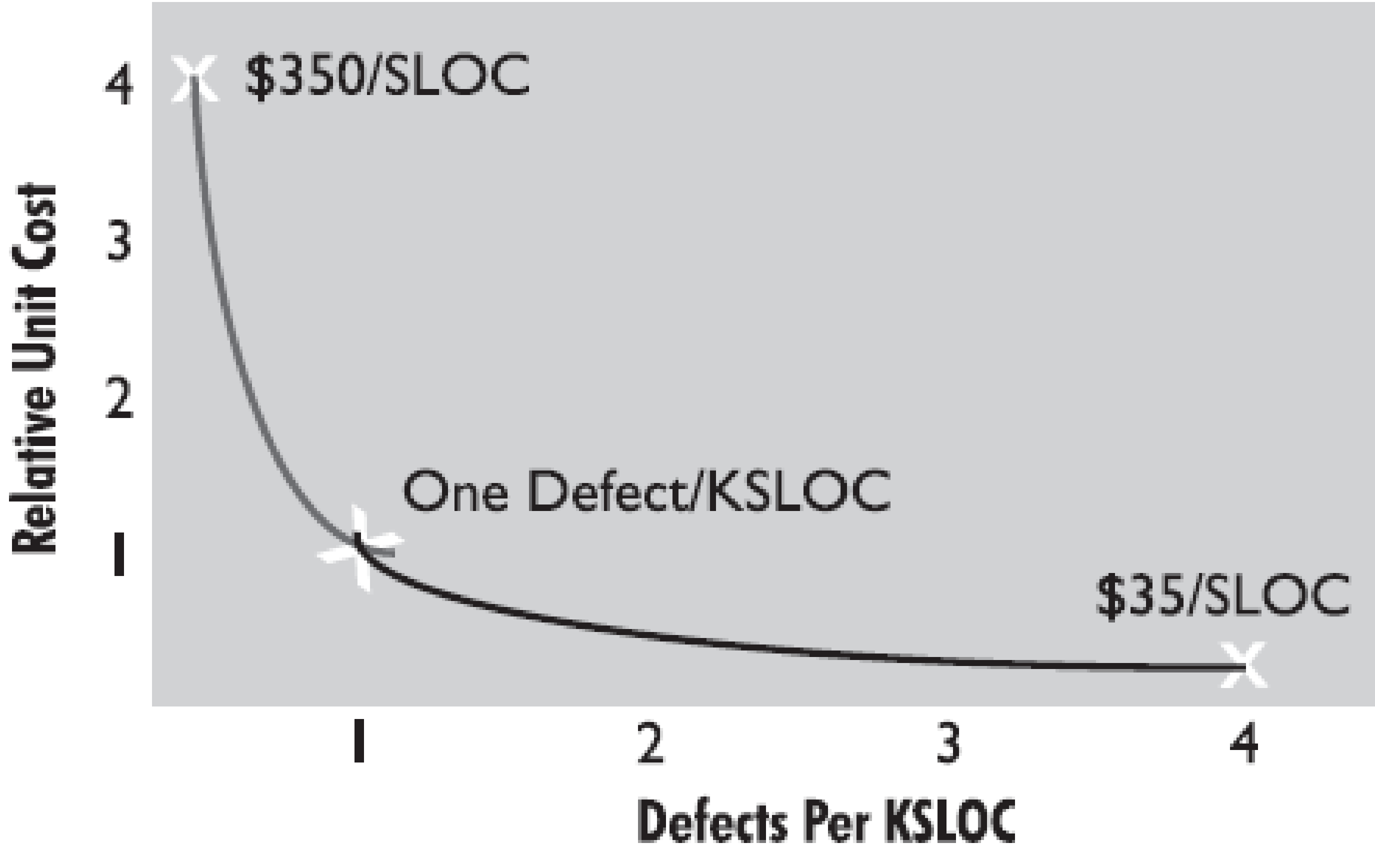
Research on “Bug free” software

- The software industry is still very immature compared to other branches of engineering,” (Dr Bengt Nordström, a computer scientist at Chalmers University in Göteborg).
  - “We want to see programming as an engineering discipline, but it’s not there yet. It’s not based on good theory and we don’t have good design methods to make sure that, at each step, we produce something that’s correct.”
- Nordström advocates a design philosophy that guarantees, from first principles, that a program will do what it says on the tin:
  - the key, he says is a mathematical approach called ‘type theory’ (funded by the EU since 1989), in which the specification for a computational task is stated as a theorem. The program that performs the computation is equivalent to the proof of the theorem – and is hence always correct.
- The TYPES* partners are releasing open source software packages that anyone can download, use and modify. These packages include several ‘proof editors’ that, in type theory, are the key to guaranteeing the correctness of programs.

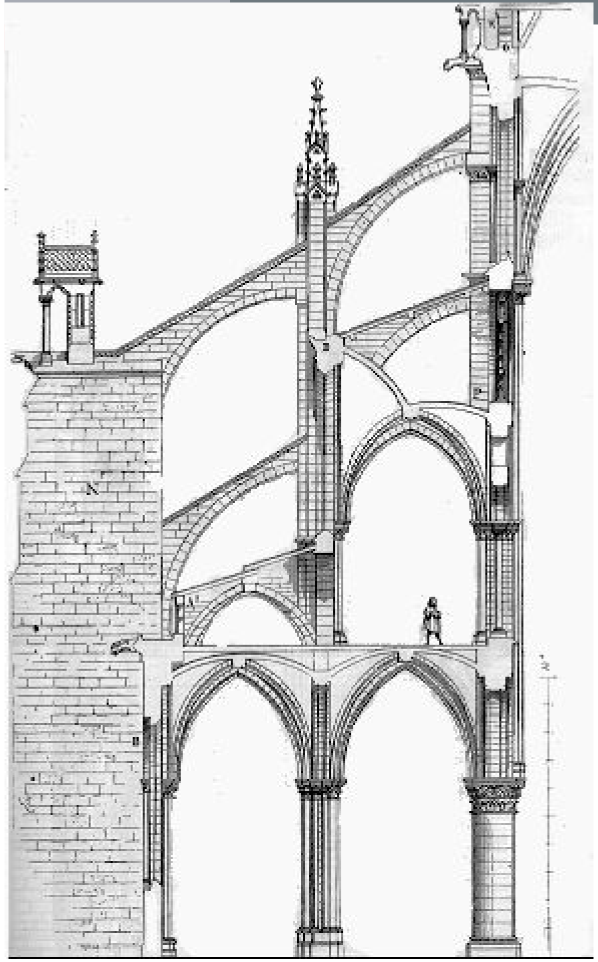
Source: Research*eu results supplement - no 7 – September 2008

Categories of software quality: state of the art

A snapshot on cost, quality, process maturity



An example of new engineering in software: the bazaar's (open source) lesson



Raymond [1] analyzes two models of software development approach: <u>the cathedral model</u>, in which source code is available with each software release, but code developed between releases is restricted to an exclusive group of software developers;

<u>the bazaar model</u>, in which the code is developed <u>collaboratively</u> over the Internet in view of the public.

Linus Torvalds Law (inventor of Linux, 1991):

"Given enough eyeballs, all bugs are shallow".

[1] The Cathedral & the Bazaar-Musings on Linux and Open Source by an Accidental Revolutionary, S. Raymond, O'Reilly 1999

Complexity in Network Centric Operation

SELEX Sistemi Integrati

Physical Domain
where effects take place and where other supporting infrastructure and information systems exist

Information Domain
where information is created, manipulated and shared

Cognitive Domain
where perceptions, awareness, beliefs, and values reside and where, as a result of sensemaking, decisions are made

Social Domain
set of interactions between and among force entities

NCO domains

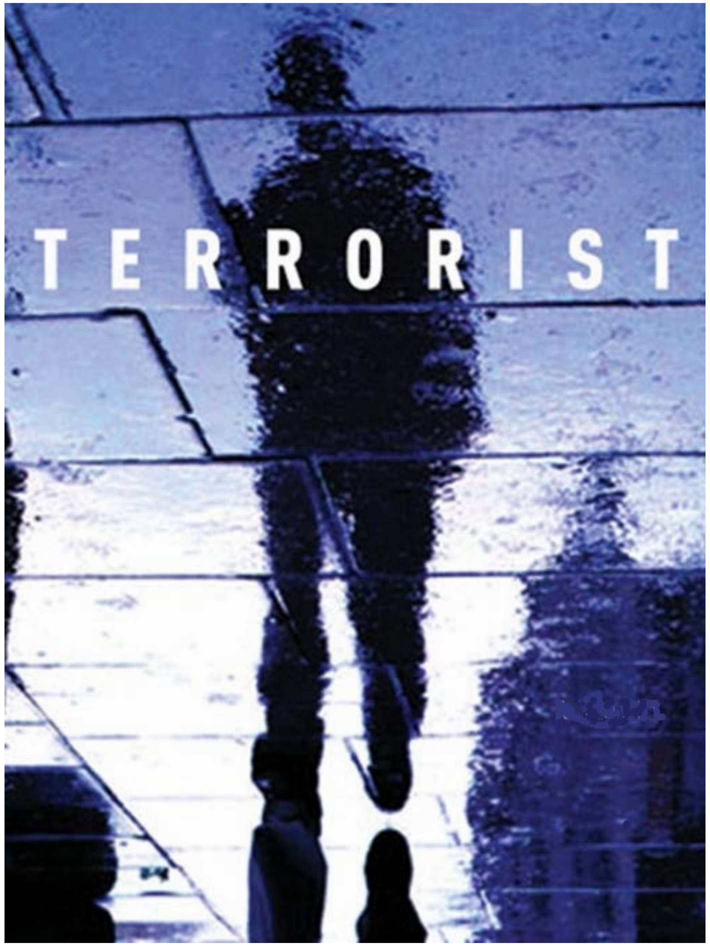
Complex Networks are recursive with social networks built upon cognitive networks which are built upon information networks. Underpinning these are physical network infrastructure.

Some factors:

- Increased technology in NCO;
- Increasing scale in NCO surveillance systems;
- Need for continuous adaptation.

Complex System and Terrorism

• Some terrorist groups have demonstrated tremendous capacity for adaptation.
• According to a report by the Homeland Security Advisory Council (USA), a terrorist group can be "proactive, innovative, well-networked, flexible, patient, young, technologically savvy, and learns and adapts continuously based upon both successful and failed operations around the globe".
• **Some terrorist groups may be best understood as complex systems**;
• Therefore, it is clear that the ability to effectively analyze complex systems has utility both for safeguarding against threats as well as mitigating or defeating potential or existing dangers.



"The best method to control something is to understand how it works."
[J. Doyne Farmer, Santa Fe Institute]

Complex System and Terrorism (cont'd)

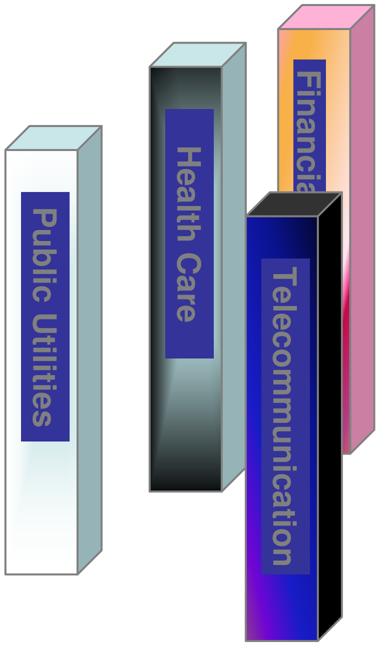
- Identical patterns of violence are currently emerging within different international war and terrorist arenas, such as Iran, Colombia, Afghanistan and non-G7 terrorism **[1]**.
  - These various examples of modern conflicts show a dynamical evolution. The respective insurgent forces are effectively becoming identical in terms of how they operate and in terms of their underlying ideologies and motivations;
  - In **[1]** a microscopic mathematical model of these forces is described. It represents the insurgent force as an evolving population of attack units whose destructive potential varies over time;
  - A power law distribution rules the probability that an event will occur with a certain number of victims.
- Such behavior can be explained using theoretical models which describe group formation, rather than having to invoke case-specific issues such as politics or geography.
- In particular, modern insurgent wars tend to be driven by the same underlying mechanism, i.e. the continual coalescence and fragmentation of attack units.

[1] N.F. Johnson, M. Spagat, J.A. Restrepo, O. Becerra, J.C. Bohorquez, N. Suarez, E.M. Restrepo, and R. Zarama, "Universal patterns underlying ongoing wars and terrorism", 2006.

Outline

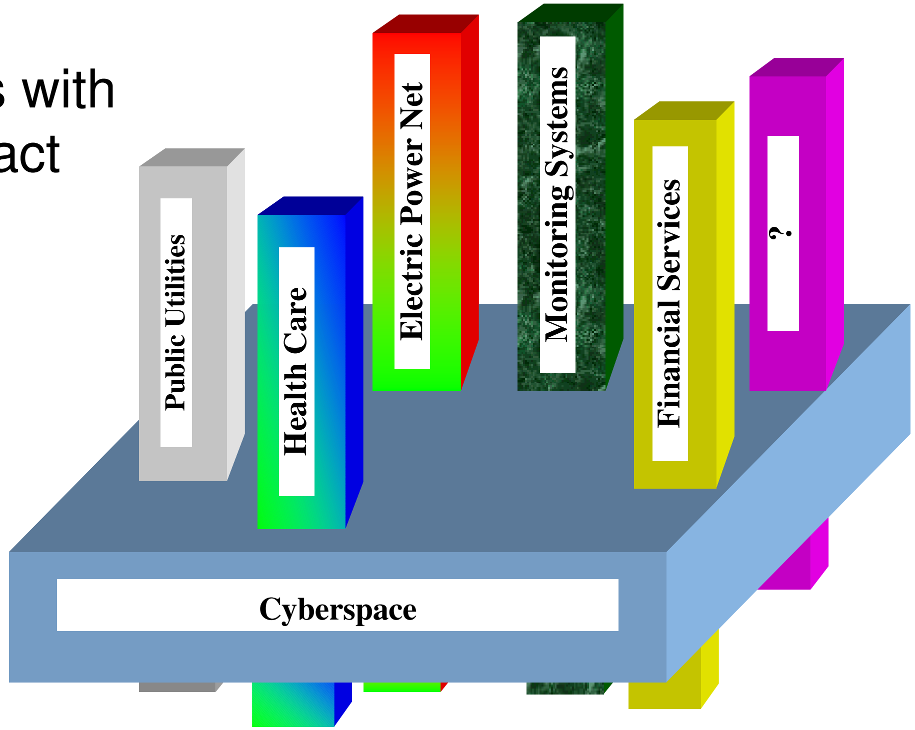
- Introduction
- Complex systems
- New engineering approach
- Additional complexity challenges
- Managing complexity
- Large and complex system examples
  - The challenges of software for large systems
- Study cases of emerging phenomena:
  - Interdependence analysis in large critical infrastructures
  - Domino effect in a large high voltage electric distribution grid
  - The traffic congestion phenomenon in internet-like networks
- General conclusions

Third Millennium Socio-techno Scenario



Before 2000

Vertically integrated infrastructures, i.e. autonomous systems with limited points of contact



After 2000

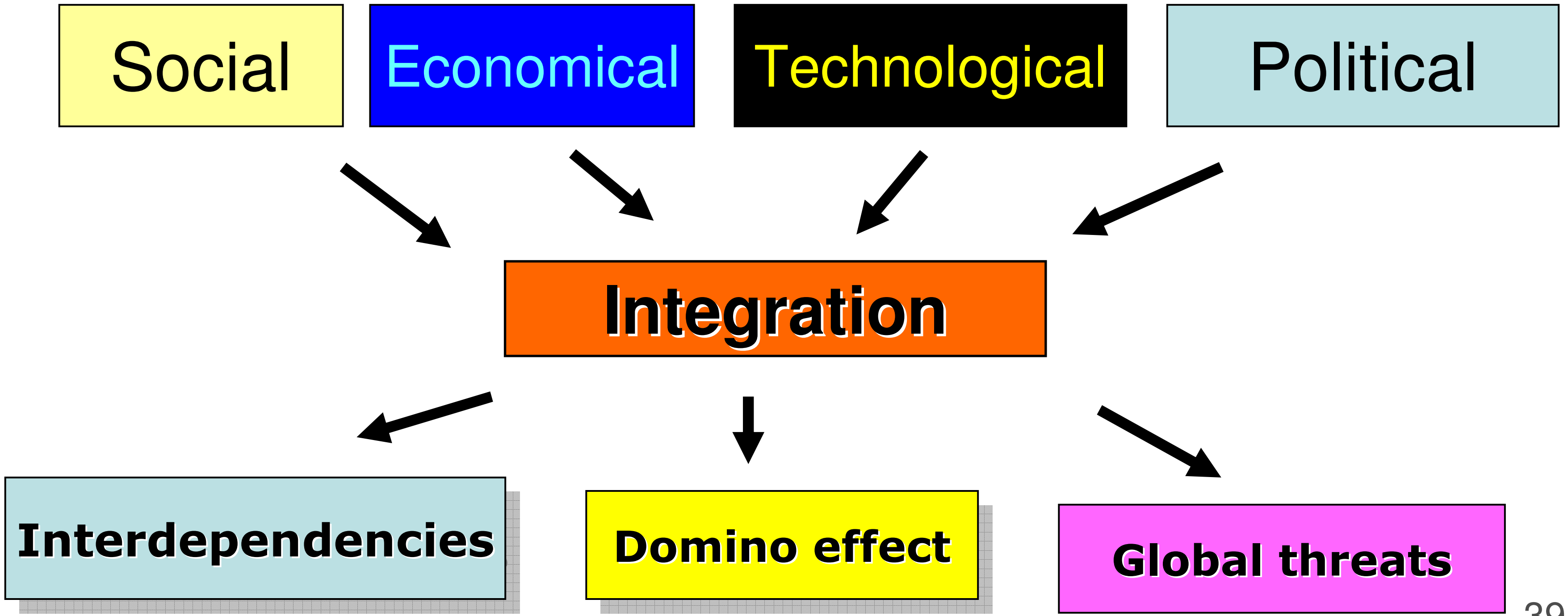
Integrated and interdependent Infrastructures sharing a common framework: the cyberspace

Integration & Vulnerability

Many years ago, the Romans learned that a good strategy to manage a complex system is

Divide et Impera

….however, *today*, for a lot of **GOOD** reasons…

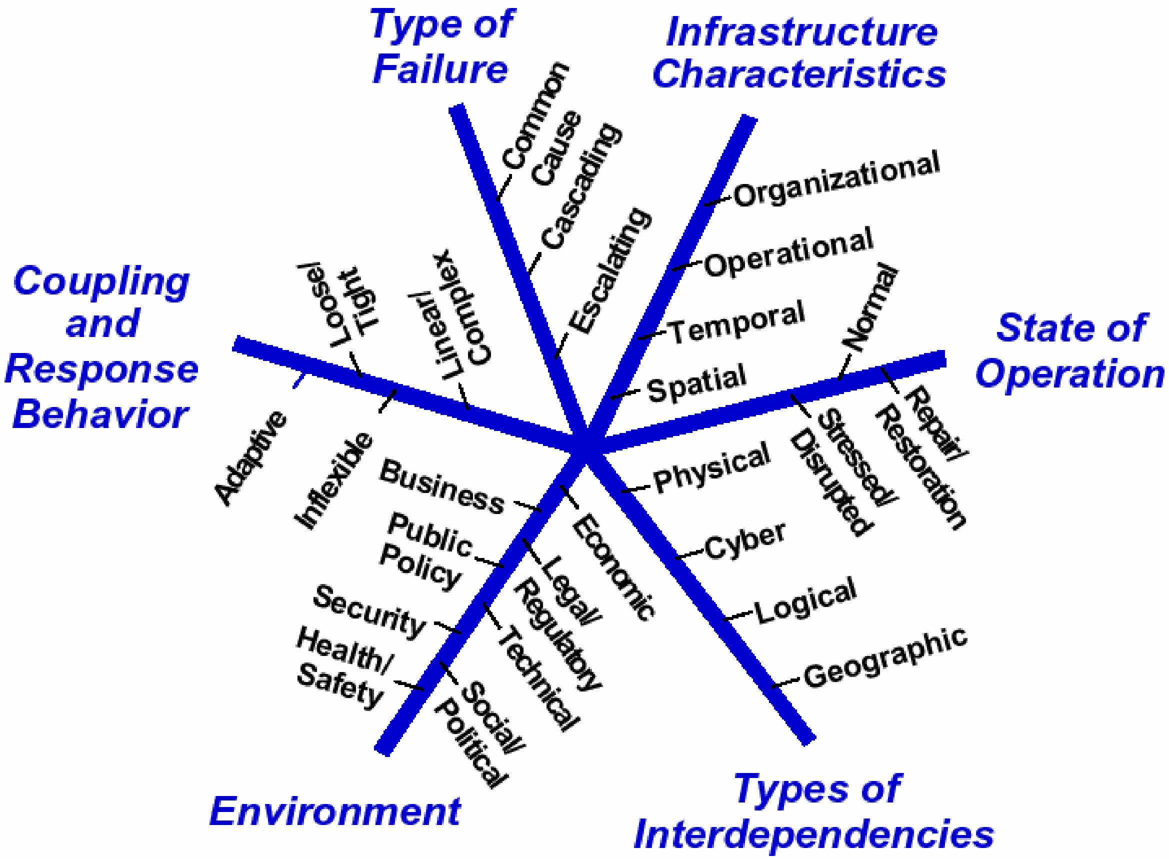


Interdependencies: the New Vulnerability

Interdependencies may be induced by a plurality of causes, many poor known or completely hidden

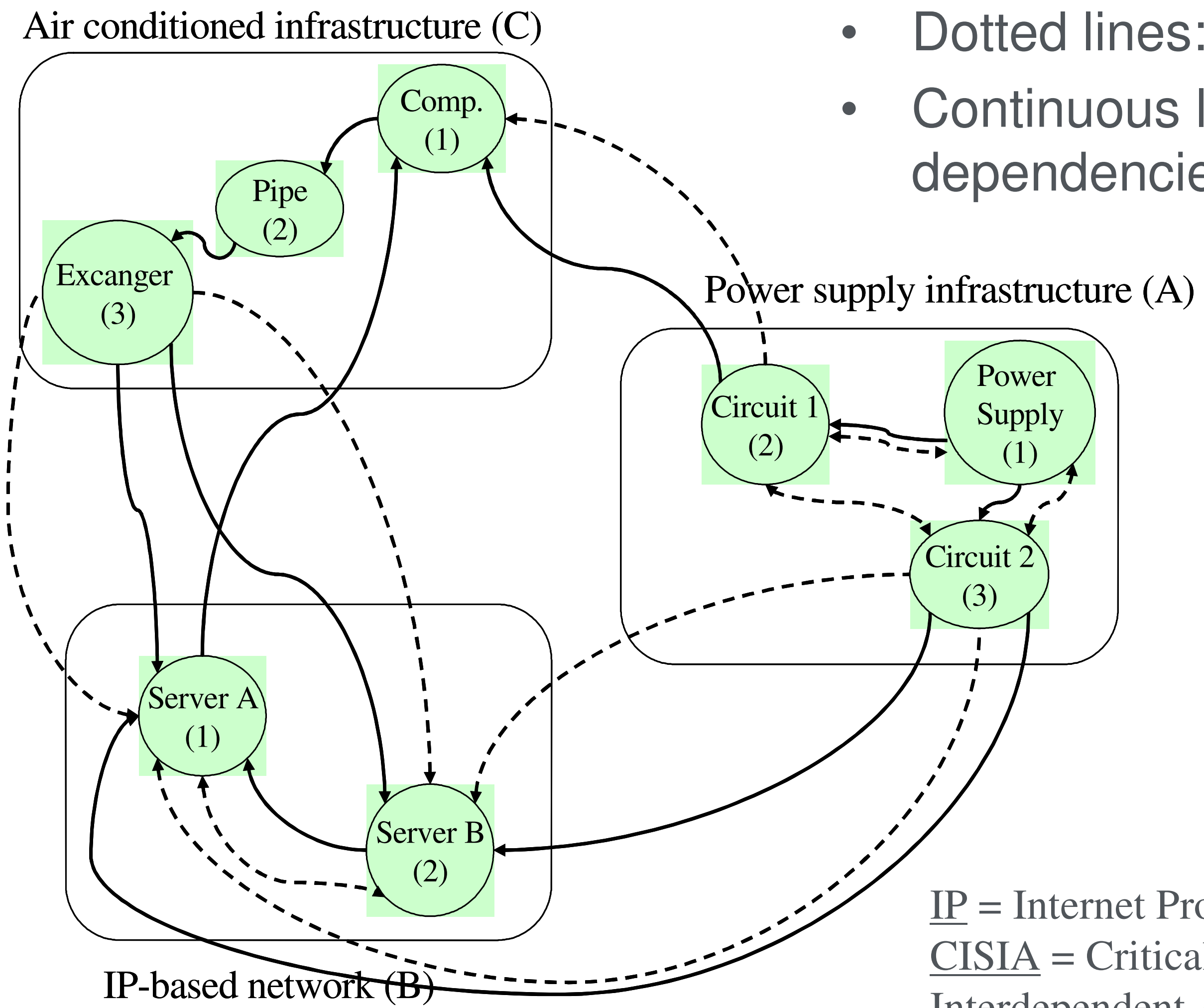
Their presence:

- augments the threat
- augments possible targets (both direct or indirect)
- amplifies the consequences of a failure
- allows simultaneous destruction of more than one service
- may induce "unforeseeable" behavior



CISIA: modeling of interconnected systems

After collaboration with University of Roma 3 and Campus Biomedico



- Dotted lines: faults propagation
- Continuous lines: functional dependencies

- Power supply infrastructure
- Information infrastructure
- Air conditioning infrastructure

IP = Internet Protocol
CISIA = Critical Infrastructure Simulation by Interdependent Agents

CISIA typical numbers & symbols

Typical numbers:

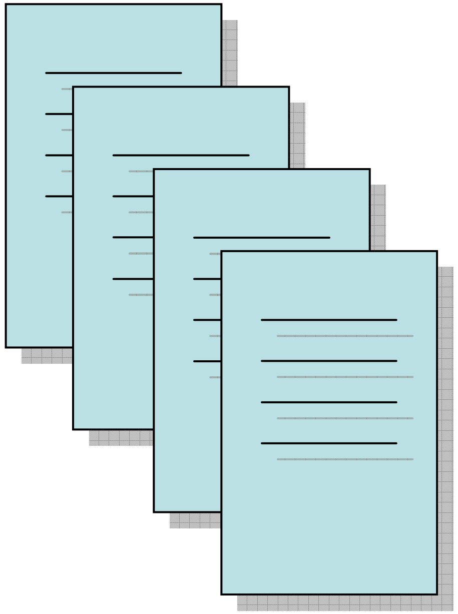
- **C++** language
- **180** classes
- Number of source files: **148**
- Code rows: **56250**
- Dimension of the source file: **3.97MB**
- Number of configuration files: **5** (xml format)
- Rows in the configuration file: **37600**
- During 36 hours of simulation, with data storing every 10 minutes, an output of 25 MB is generated, whose entries are **400.000**

Symbols:

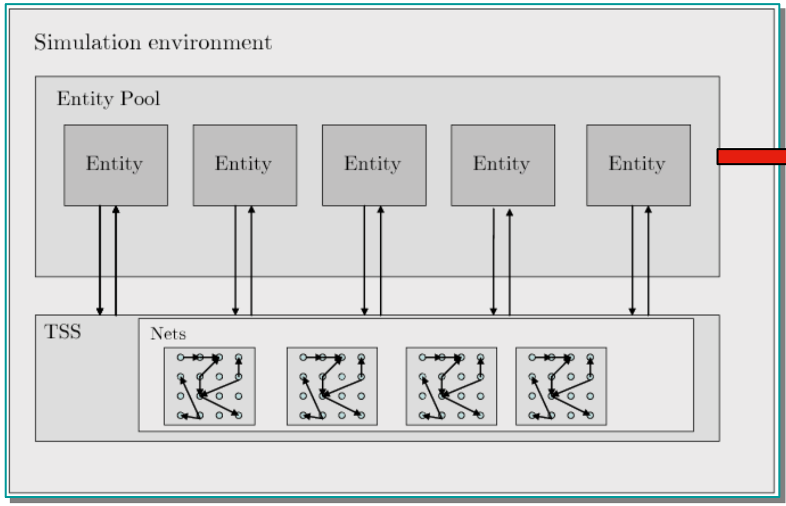
	Electrical power network			Train station
	Electrical power plant			Train track
	Gas pipe			Highway station
	Water pumping			Street
	Urban area			Highway services
	Airport			Telecommunication infrastructure
	Harbor			OCD supply

CISIA-GIS structure

Knowledge derived from the interviews

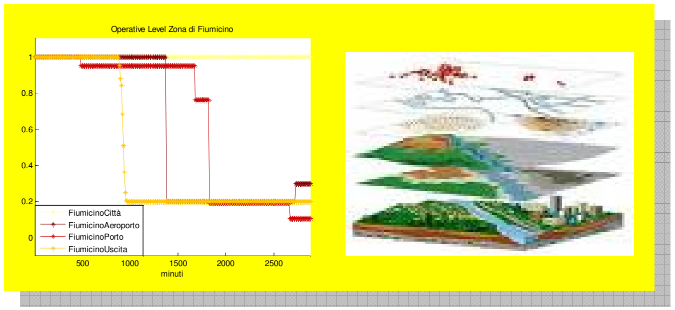


CISIA Simulator



Entity status in the time
- operative level
- failures

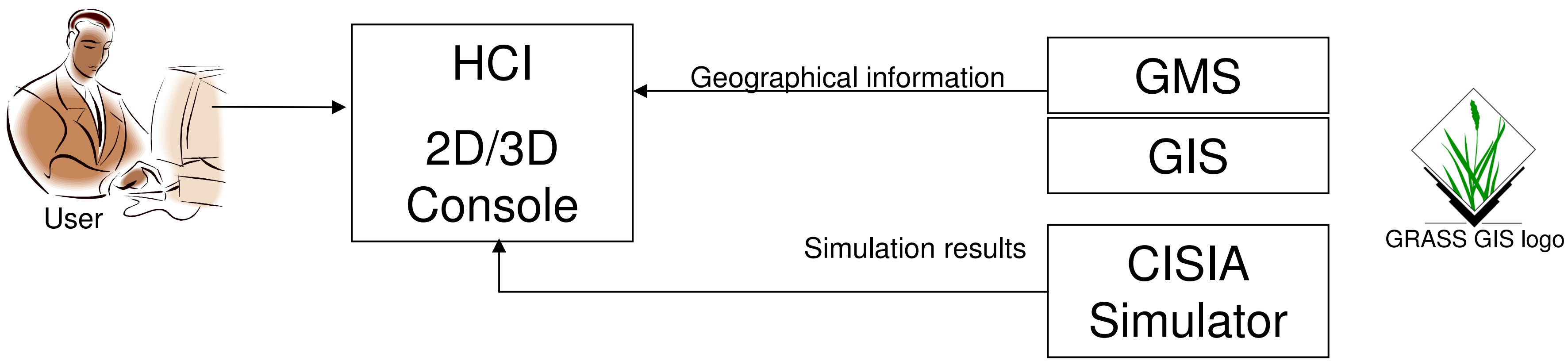
Status of the overall system



GIS

TSS: Transmission Sub System.

Architecture of the CISIA-GIS software



HCI (Human Computer Interaction): console developed by SELEX Sistemi Integrati to present synthetic data on cartographic base.

GMS (GIS Management System): product developed by SELEX Sistemi Integrati (winner of the Company Innovation award 2006) which represents the interface of the system with an external **GIS (Geographic Information System)** product, such as the GRASS (*Geographic Resources Analysis Support System*).

CISIA: simulator of the evolution events (*fault*) on the system of critical infrastructures.

Analogy between the domino effect and the phase transition (*)

Statistical mechanics	Electrical power grid
Temperature	**P_D/P_C**
Phase transition = e.g. transition of the water from the liquid state to the vapor phase observed when the pressure is 1 atmosphere and the temperature is 100°C (i.e. critical temperature).	***Domino effect*** = transition of one part of the electrical grid from an exponential density probability of the black out magnitude to a different density power (i.e. power law distributed). This transition is observed for critical values of P_D/P_C.
When the temperature increases some bubbles appear in the liquid water, i.e. in the water some "collective statuses" appear, where a lot of adjacent molecules behave in the same way (they are in the gas phase, "bubble of vapor").	When P_D/P_C increases the grid is no more able to supply the power request and the consequent disconnection of the power lines generates "bubble of black out". In the worst cases these bubble can cause a failure of the whole grid.

(*) A. Farina, A. Graziano, F. Mariani, F. Zirilli, "Probabilistic analysis of failures in power transmission networks and "phase transition": a study case of an high voltage power transmission network", Journal of Optimization Theory and Applications (JOTA), 139 (2008), pp. 171-199.

Graph of a high voltage power grid

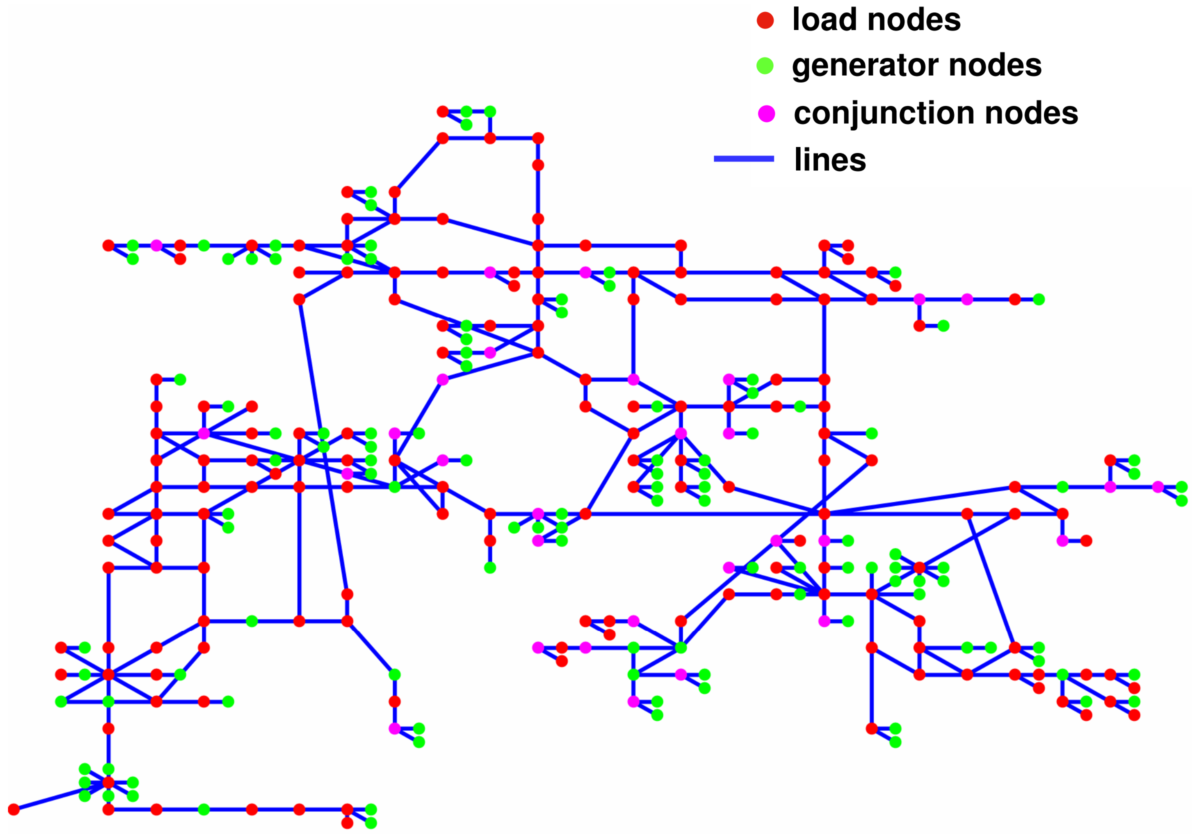
The model of DC Power Flow

The electrical power grid is described by a un-oriented graph, i.e. a set of nodes connected by branches.

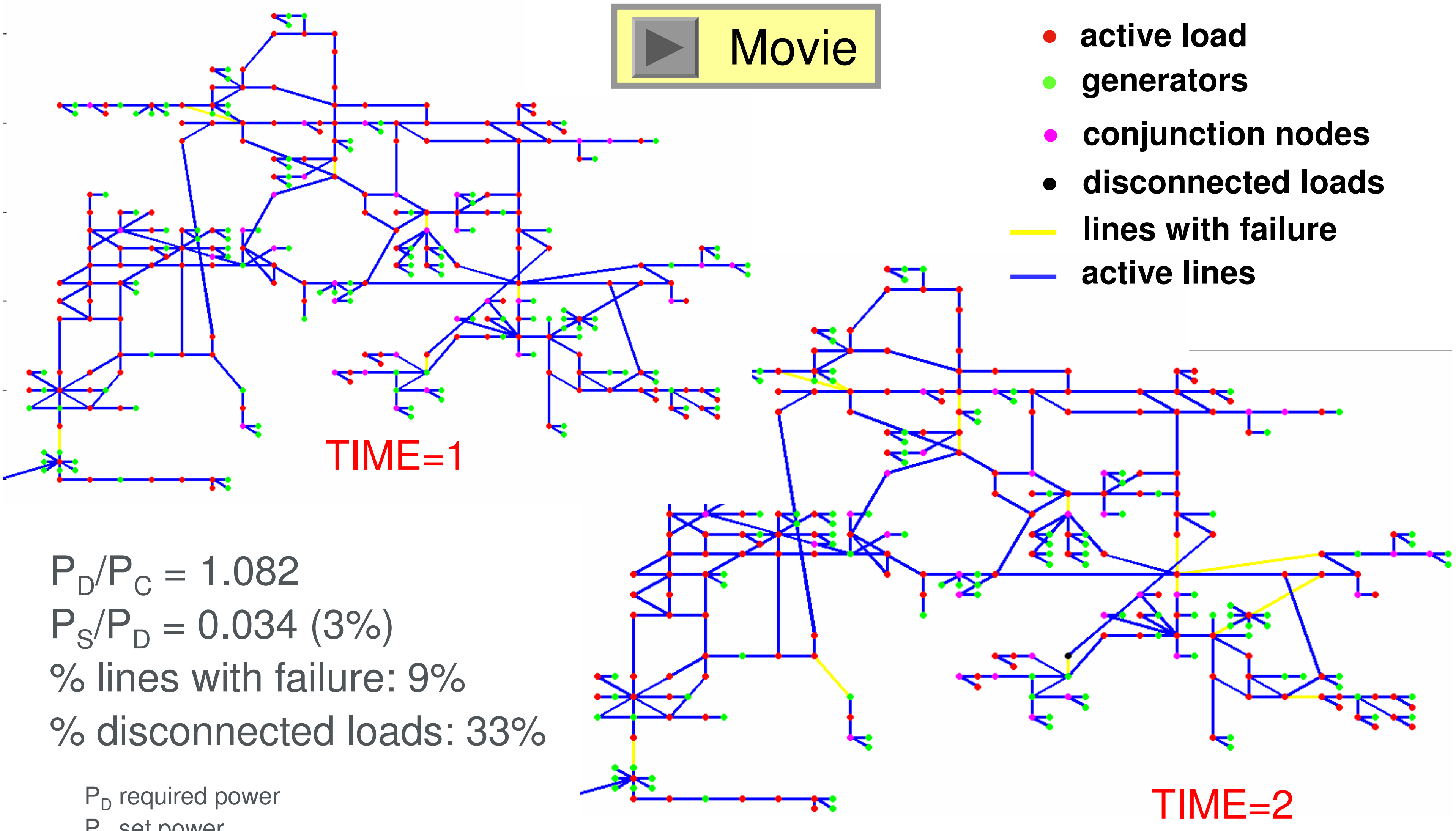
The nodes are divided into:

- generator nodes,
- load nodes,
- conjunction nodes.

Each branch (electrical line) is characterized by impedance and by a maximal power which can flow on it without generating a failure.



Blackout dynamic



$P_D/P_C = 1.082$

$P_S/P_D = 0.034$ (3%)

% lines with failure: 9%

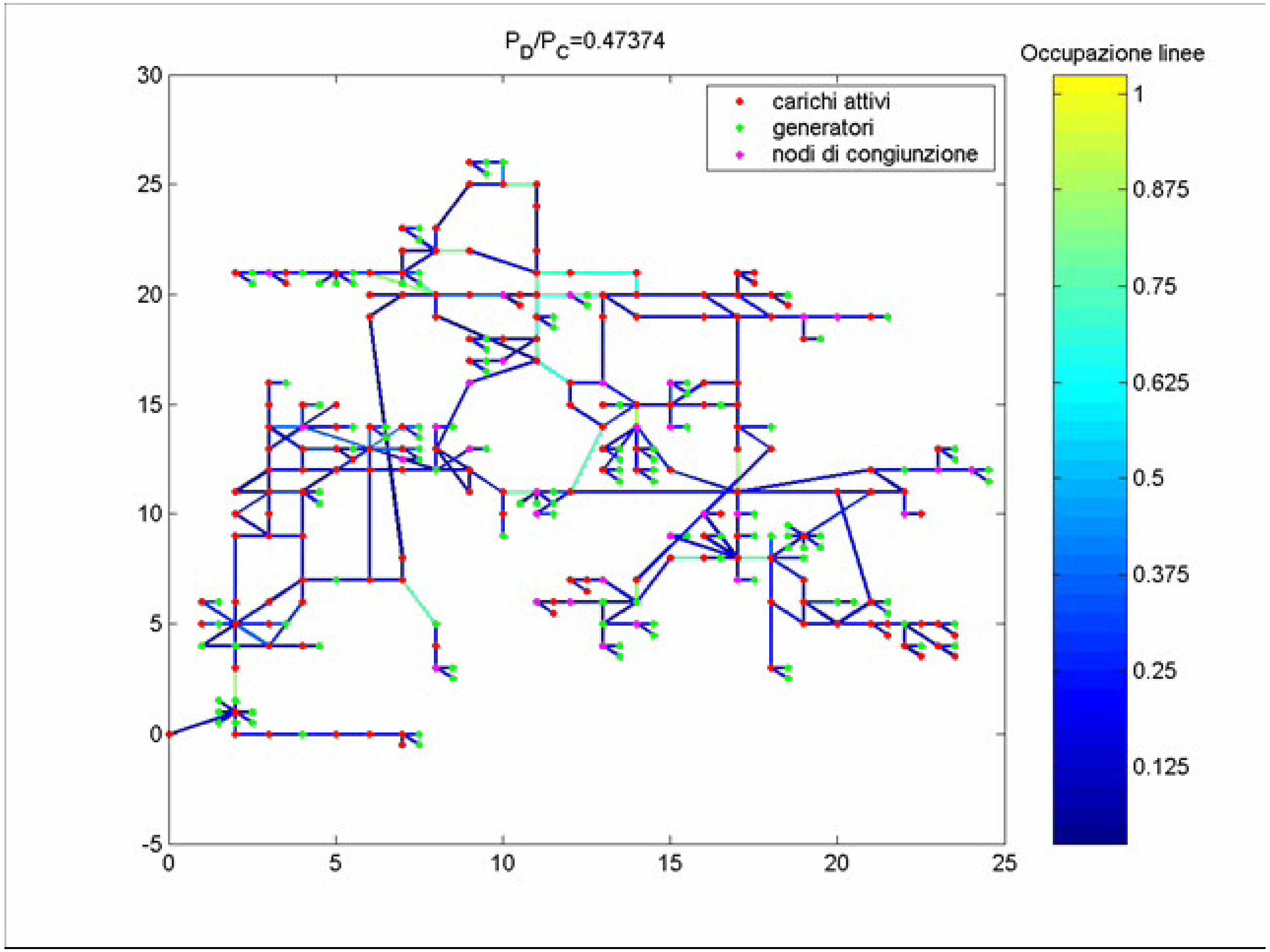
% disconnected loads: 33%

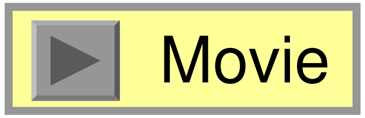
P_D required power

P_C set power

P_S obscured power

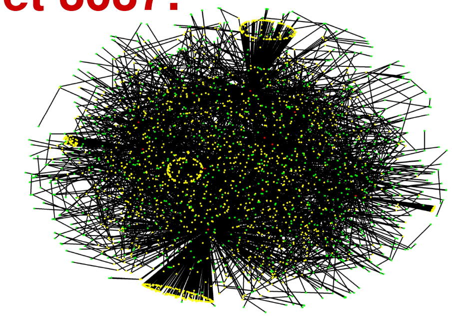
Average part of grid adopted





The traffic congestion phenomenon in internet-like networks

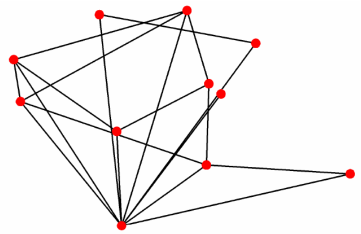
Inet 3037:



Node number: 3037

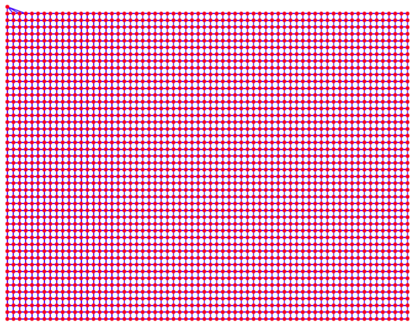
Branch number: 4788

The node degree distribution follows a power law.



The rich club of Inet 3037 (12 Hubs).

Rectangular lattice:



Node number: 3037

Branch number: 5964

The node degree is basically constant (4).

Network topological properties:

$$A^{\alpha} = \left[\left((1-\alpha) A^{I} + \alpha A^{R} \right) \right], \ \alpha \in [0,1]$$

Weighted adjacency matrix of “α-network”

Weighted adjacency matrix of Inet

Weighted adjacency matrix of modified rectangular lattice

Congestion phenomenon in homogeneous traffic case

Simulation conditions:

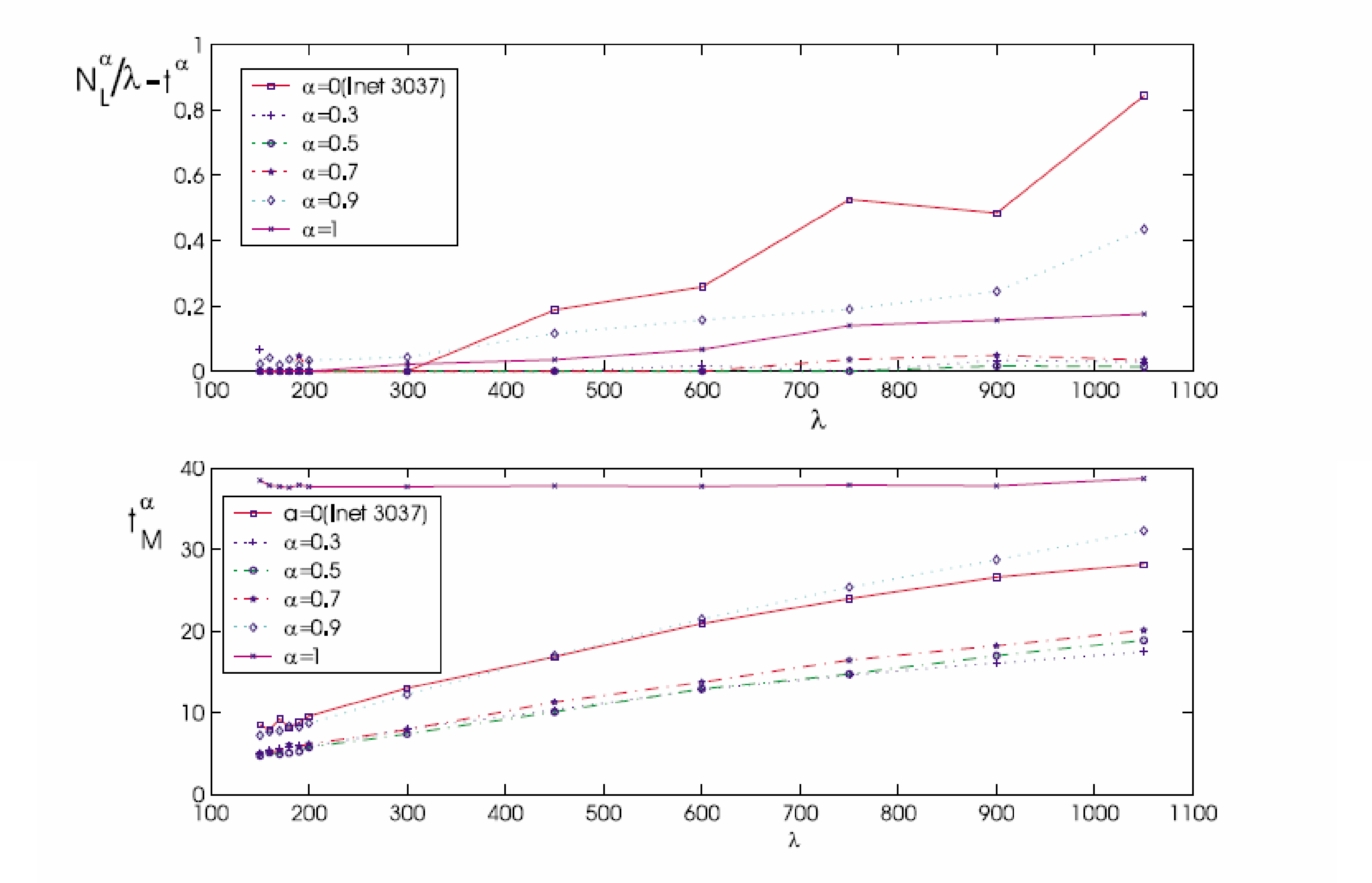
1. Choice of the number of network topologies $N_\alpha+1$ that will be considered in the simulation and of the network topologies, i.e. assign the values of α used in the simulation;
2. Packet generated as samples from a Poisson distribution;
3. Routing based on the search of the shortest weighed path connecting the source node with the destination node.
4. When the packet reaches the destination it is removed from the simulation.
5. If there is not capacity available the packet will wait for the next available time unit according to the queue management rules.

Parameter definition:

λ	mean value of the number of packets generated in the entire network in a time unit
n	number of nodes in the network
λ / n	average number of packets generated by each node per time unit
$N_L^\alpha(\lambda)$	mean number of packet that are travelling in the network and have not reached their destination
$t_M^\alpha(\lambda)$	mean travel time of the data packet
λ^*	Value of the parameter λ that separates the free flow state from the congested state

Congestion phenomenon in homogeneous traffic case (cont'd)

Performance:



Congestion phenomenon in heterogeneous traffic case

Simulation conditions:

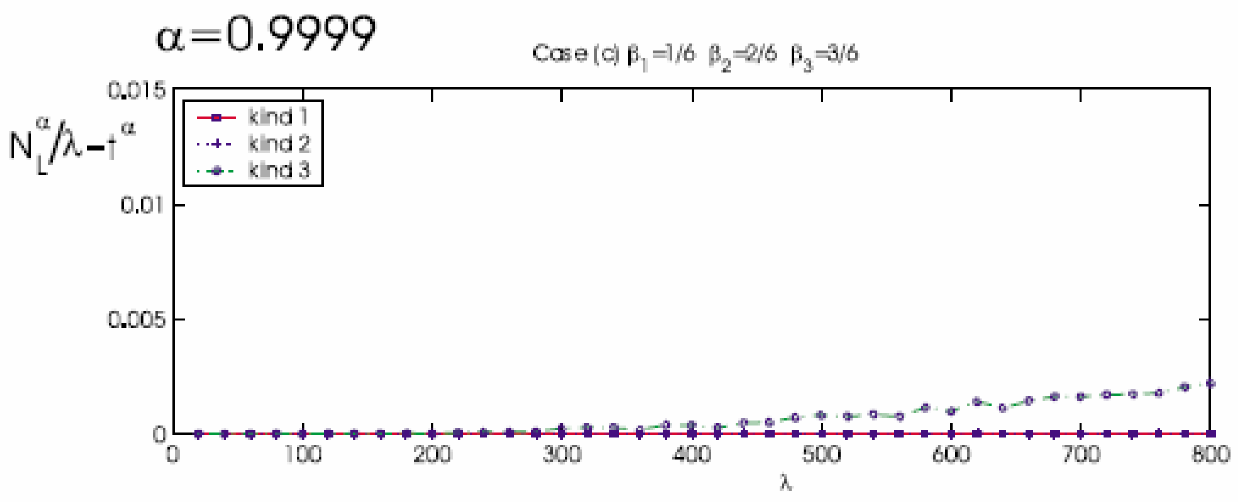
1. Choice of the number of network topologies $N_\alpha+1$ that will be considered in the simulation and of the network topologies, i.e. assign the values of α used in the simulation;
2. Packet generated as sample from a Poisson distribution;
3. Routing based on the search of the shortest weighed path connecting the source node to the destination node;
4. When the packet reaches the destination it is removed from the simulation.
5. If there is not capacity available the packet will wait for the next available time unit according to the queue management rules.

Parameter definition:

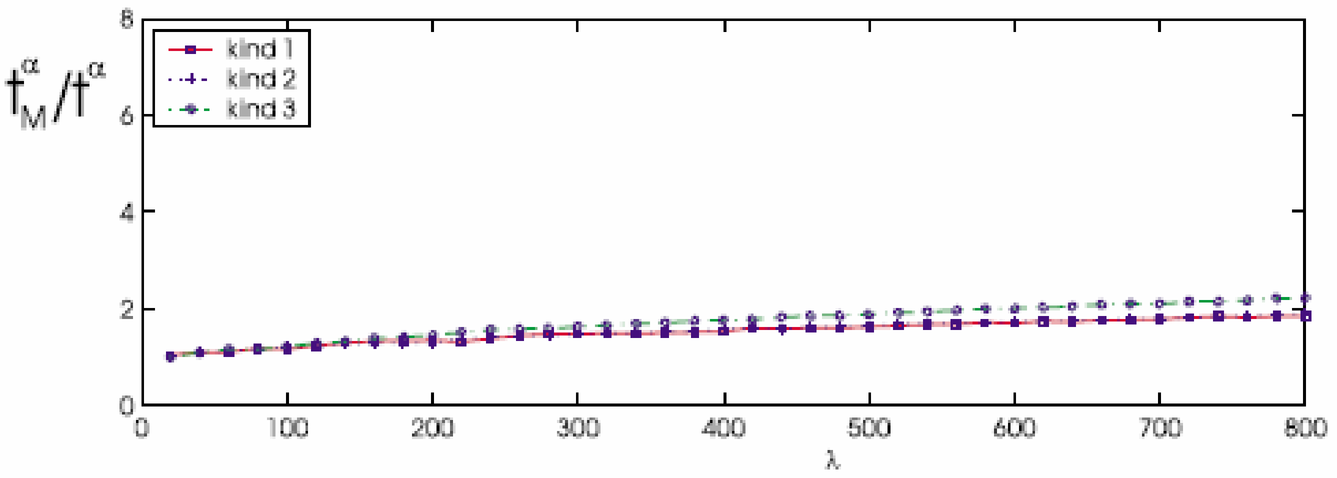
λ	mean value of the number of packets generated in the entire network in a time unit
$\beta_i\lambda$	Mean load of the packets of kind i (i=1,2,3)
n	number of nodes in the network
$N_{Li}^{\alpha}(\lambda)$	mean number of kind i packet that are travelling in the network and have not reached their destination
$t_{Mi}^{\alpha}(\lambda)$	mean travel time for kind i data packet
λ_i^*	Critical value of the load parameter λ for the α-network and the kind i data packet

Congestion phenomenon in heterogeneous traffic case (cont'd)

Performance case: $\beta_1 = 1/6 \quad \beta_2 = 2/6 \quad \beta_3 = 3/6$ **(common situation)**



α = 0.9999



Outline

- Introduction
- Complex systems
- New engineering approach
- Additional complexity challenges
- Managing complexity
- Large and complex system examples
  - The challenges of software for large systems
- Study cases of emerging phenomena:
  - Interdependence analysis in large critical infrastructures
  - Domino effect in a large high voltage electric distribution grid
  - The traffic congestion phenomenon in internet-like networks
- General conclusions

General conclusions

- We have tried to address the complex systems problem in its broadest frame - taking also advantage of the theoretical work available since 1970s from science of complexity;
- We have also tried to re-focus on the aspects of greatest interest to us;
- This is by no ways the final word on complex systems, but only an evidence of our awareness and understanding of the issues in this field.
- We have shown the importance of a **new engineering** approach that helps us to master and exploit complexity in systems;
- Some applications are now in reach;
- Complexity issues arise in all domains (e.g. business organization and management);
- **Scale is increasing in systems and scale changes everything.**

- **"I think this century will be the century of complexity."**
  Stephen Hawking (Complexity Digest 2001/10, 5 March 2001.)

Bibliography

- Gleick J., “Chaos: Making a New Science”, Penguin Books, 1987.
- Waldrop, M. M., “Complessità: uomini ed idee al confine tra ordine e caos”, Simon & Schuster, 1992.
- Gell-Mann, M., “The Quark and the Jaguar: Adventures in the Simple and the Complex”, 1994.
- Kauffman, S., “A casa nell’universo: Le leggi del caos e della complessità”, Editori Riuniti, 1995.
- Buchanan, M., “Ubiquità”, Mondadori, 2000.
- Barabasi, A. L., “Linked: how everything is connected to everything else and what it means”, 2000.
- Moffat, J., “Command and Control in the Information Age”, The Stationery Office, London, UK, 2002.
- Strogatz, S., “Sincronia (I ritmi della natura, i nostri ritmi)”, Rizzoli, 2003.
- Moffat, J., “Complexity theory and Network Centric Warfare”, CCRP, 2003.
- Cini, M., “Un paradiso perduto: dall’universo delle leggi naturali al mondo dei processi evolutivi”, Feltrinelli, 2004.
- Perry, W., Moffat, J., ”Information sharing among military headquarters: the effects on decision making”, RAND, 2004.
- Kurzweil, R., “The singularity is near”, Viking 2005.
- Carafano, J.J, Weitz, R., “Complex systems analysis- a necessary tool for homeland security”, Backgrounder, No. 2261, April 16, 2009.
- Johnson, N., “Due è facile tre è complessità”, Edizioni Dedalo, 2009.