Ensuring and demonstrating diverse quality attributes of complex systems: problems of models and cultures

Lorenzo Strigini
Strigini@csr.city.ac.uk

GAUSS2019 - 1st International Workshop on Governing Adaptive and Unplanned Systems of Systems, 2019

with thanks to ECSEL grant No 737475,
Aggregated Quality Assurance for Systems (AQUAS)

CSR Building confidence in a computerised world

www.csr.city.ac.uk

Outline

- where this talk comes from
- [unplanned, adaptive ...] systems of systems - what do we mean?
- multiple attributes: efficiency, safety, security, ...
- cultures, culture gaps and effects on modelling and insight

Background to this talk

- I have worked on dependability matters for some 40 years
- the organisers suggested I distill insights of interest for this community
- I'll address how some problems that my colleagues and I have been studying
  - how to know how well your fault tolerance, your diverse layers of defence... work
  - how to take into account human components of a system, and their changes
  - how to integrate in practice considerations of safety, security, etc
  - how to reason as to whether events that never happened and would better never happen may actually never happen

... are relevant to "systems of systems":

"roles of models and cultures in dealing with diverse quality attributes of complex systems":

CITY
UNIVERSITY OF LONDON
EST 1894

Background to this talk - 2

Serious anomalies keep happening in complex "systems of systems", even when centrally managed for a single goal ("directed")

e.g. today's New York Times headline

"This did not go well –inside PG&E's blackout control room"

https://www.nytimes.com/2019/10/12/business/pge-california-outage.html

(the utility company running managed blackouts due to the fire/weather emergency in California suffers failures in communicating essential information to users)

The inevitable "definitions" slide

- **System of Systems (SoS)** — *Set of systems or system elements that interact to provide a unique capability that none of the constituent systems can accomplish on its own.* [SO/IEC/IEEE 21839]

*I.e. **any** system is a system of systems?*

No: "system" here has an administration/procurement meaning

The inevitable "definitions" slide, improved

- System of Systems (SoS) : “a set or arrangement of systems that results when ***independent*** and useful systems are integrated into a larger system that delivers unique capabilities”
  [DoD Defense Acquisition Guidebook., Systems Engineering Guide for Systems of Systems, 2008]

The inevitable "definitions" slide, improved

- System of Systems (SoS) : “a set or arrangement of systems that results when ***independent*** and useful systems are integrated into a larger system that delivers unique capabilities”
  [DoD Defense Acquisition Guidebook., Systems Engineering Guide for Systems of Systems, 2008]

In essence: *the components systems are somewhat independent - in design, procurement and/or actual behaviour and evolution thereof*

- creating areas of uncertainty/ignorance
- examples:
  – road traffic+road infrastructure;
  – air traffic, ATM and related infrastructure
  – a warship or combat aircraft or task force
  – a hospital or hospital ward or operating theatre
  – smarts energy grids
  – E-commerce

Is "Systems of systems" engineering different from ***system engineering*?**

First answer: NO

"change in one component [...] may impact the safety of the system when that component interacts with other[s] [... calling this a "system of systems" [...] does not solve the problem. [...] more information is required [...] than [...] their external interfaces [...]   When putting two or more [...] ("systems") together, the emergent properties must be analyzed for the integrated system. Calling that [...] a "system of systems" may be misleading by implying that emergent properties can be treated differently than any other system or different system engineering techniques can be used"

N. Leveson, "The Drawbacks in using the term "system of systems", Biomedical Instrumentation and Technology, March/April 2013

"Systems of systems" engineering: just ***system engineering*!**

... ideally.

But "complex systems of systems" make it harder

The pockets of ignorance and uncertainty make it difficult to prove what the system as a whole will do

good advice is well established:

.... be concerned with "the end-to-end behaviour of the SoS", understand what is going on, "orchestrate" upgrades, model/simulate, ...
[DoD Systems Engineering Guide for Systems of Systems, 2008]

"Unplanned systems of systems" engineering?

- "safety is a system property"... like other important properties
  - e.g. overall fuel consumption across all highway users
  - probability of catastrophic blackouts in a power grid
  - financial viability of an infrastructure

- system engineering is about figuring out / controlling overall properties of the whole system

- "unplanned [adaptive] systems of systems" are systems for which this cannot be done [to the same level of accuracy/predictability/confidence to which we aspire for simpler and less changeable systems]"

"Unplanned systems of systems" engineering?

- .....systems for which analysis cannot be done to the desired level of accuracy/predictability/confidence"

- is this new?
  Not really: cities, countries, markets are such systems
- Social scientists, managers, ministers ... have dealt with such systems for a long time:
  - producing insights (some of them right), some accepted laws, some useful techniques ...

Where are the boundaries of "unplanned systems of systems" engineering?

Nowhere:

- even in tightly designed, small embedded computer-based systems meant to be immutable in time, the supplier of a chip can decide changes that undermine assumptions made by the system designer

- autonomous entities with their own separate goals will nonetheless obey (to a greater or lesser extent) laws/protocols that are designed (to a greater or lesser extent) and safeguard common objectives

CITY
UNIVERSITY OF LONDON
EST 1894

Governing vs analysis/assurance

Governing: "ruling, steering"

- i.e., feedback control: monitoring and responding to deviations
- should temper the problem of uncertainty/ignorance that prevent detailed trustworthy prediction
- and give some guarantee against bad surprises

- yet for most systems we need some assurance that these techniques deliver

hence we need to analyse the whole system ["of systems"] *including* its monitoring/governing functions

  - large role for modelling / simulation

- what is hard about it?
  - size and complexity of models? (not the main difficulty!)
  - completeness? Need for imaginativeness? Yes

Dealing with multiple quality attributes

Much investment in the last 20 years to address security of safety-critical systems, **but**...

- integration of security concerns still complex, problematic
  - different cultures within companies
  - safety & security people speak different languages, use different concepts
  - often different emphasis
    - \+ e.g. safety people want "immutable" designs verified for the long term
    - \+ vs security people desiring fast change to address new threats
  - often requiring trade-offs in design
    - \+ missing a conflict may cost expensive design rework, or worse
- uneasy evolution in practice and standards
  - some consensus that you cannot separate the two: e.g. IEC 61508 "requires malevolent and unauthorised actions to be considered during hazard and risk analysis"
  - but much resistance too: e.g. weak support in ISO 26262

Safety, security, ... are interdependent

1. one relies on the other: "if it's not secure it's not safe": adversaries can cause accidents
2. trade-offs
   - ***goals*** may conflict
     + requiring operator to prove identity before entering critical commands... to prevent malicious commands causing accidents
     + may delay emergency intervention to stop an accident
   - safety and security ***mechanisms*** may conflict:
     + e.g. extra encryption may slow down communication and violate real-time requirements
     + redundancy/diversity may increase attack surface
3. many synergies as well
   - e.g. a security-oriented precaution may improves reliability/safety
   - ignoring this may be costly

Dealing with multiple quality attributes – simplification1: separation

- some of my colleagues have been advocating *security-informed safety cases*, with some success

- but in many industrial contexts (both development and operation) we hear
  - safety is complex enough without worrying about security
  - co-ordination too difficult / not needed
  - if I build, operate, upgrade a system that is safe given certain conditions
  - then you (the security expert) only need to guarantee that the assumptions under which I have proved it safe will be and remain true

- why not? Have clear "contracts", "rely-guarantee"
- but... is it that easy?
  - can (do) experts in one culture define in advance *all* assumptions that the other cultures should keep true?

CITY
UNIVERSITY OF LONDON
EST 1894

Dealing with multiple quality attributes – simplification 2: standards

- most engineering involves following rules
- safety and security standards abound
- engineers depend on them to
  - ensure [a degree of] coverage of design concerns
  - cover their backs legally
  - protect users
- so what's there to worry about?
- the easier prescriptions to write/follow are about the easier requirements
  - focused on individual "system", not larger "system of systems"
  - focused on individual qualities, e.g. security "controls" or safety mechanisms, not their interplay
  - deterministic about precautions to take, not probabilistic about their results in a complex system

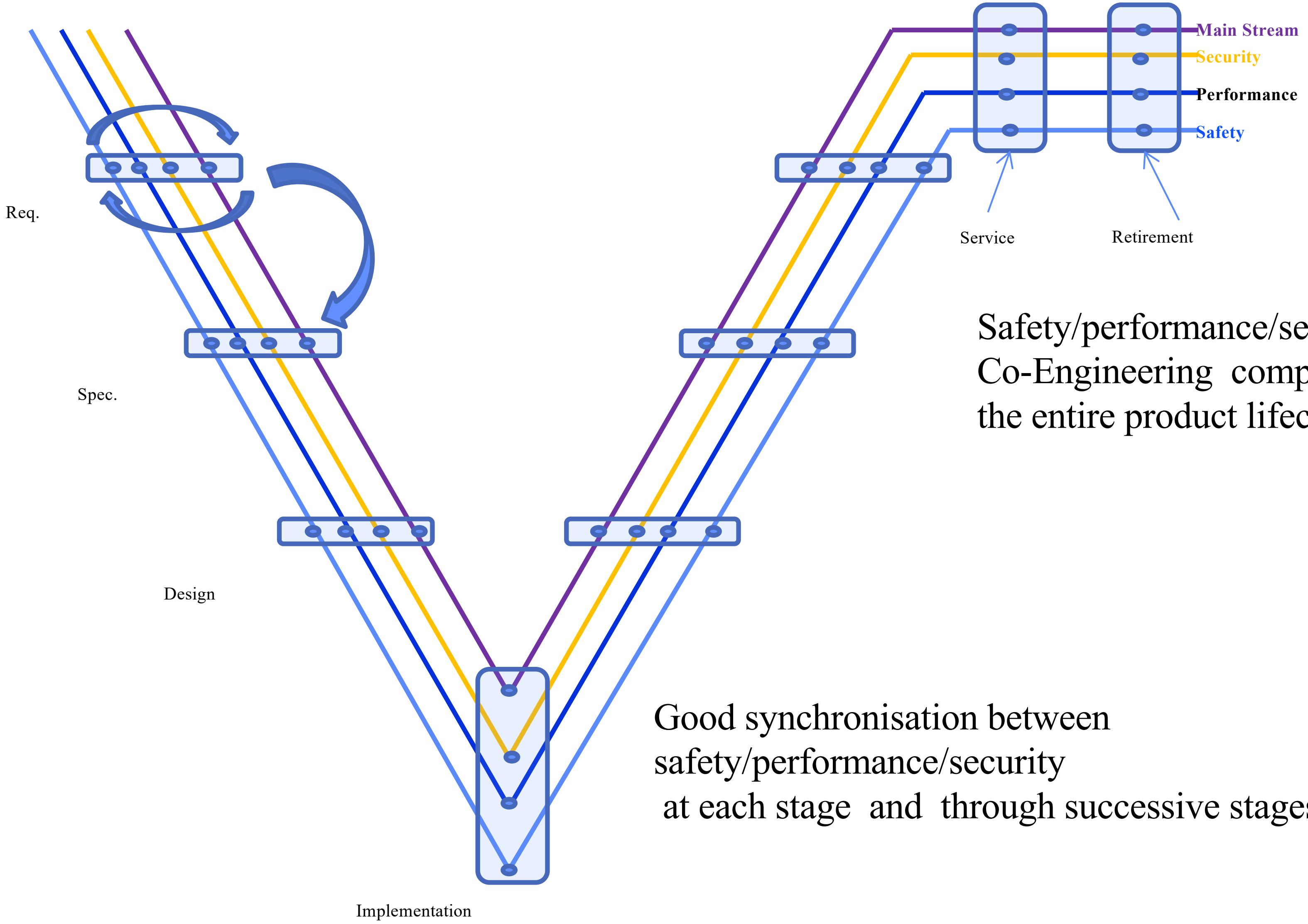
the hard parts may be de-emphasised through absence of direct prescriptions

Example: project AQUAS Aggregated Quality Assurance for Systems

investigating Co-Engineering techniques for safety, security and performance in critical and complex embedded systems

- aims at progress
  - at various levels of integration
  - across the lifecycle
  - integrated/able in current development processes
- supported by tools
- with goal to improve industrial practice and standards

The need



Safety/performance/security Co-Engineering comprises the entire product lifecycle.

Good synchronisation between safety/performance/security at each stage and through successive stages.

"Interaction points"

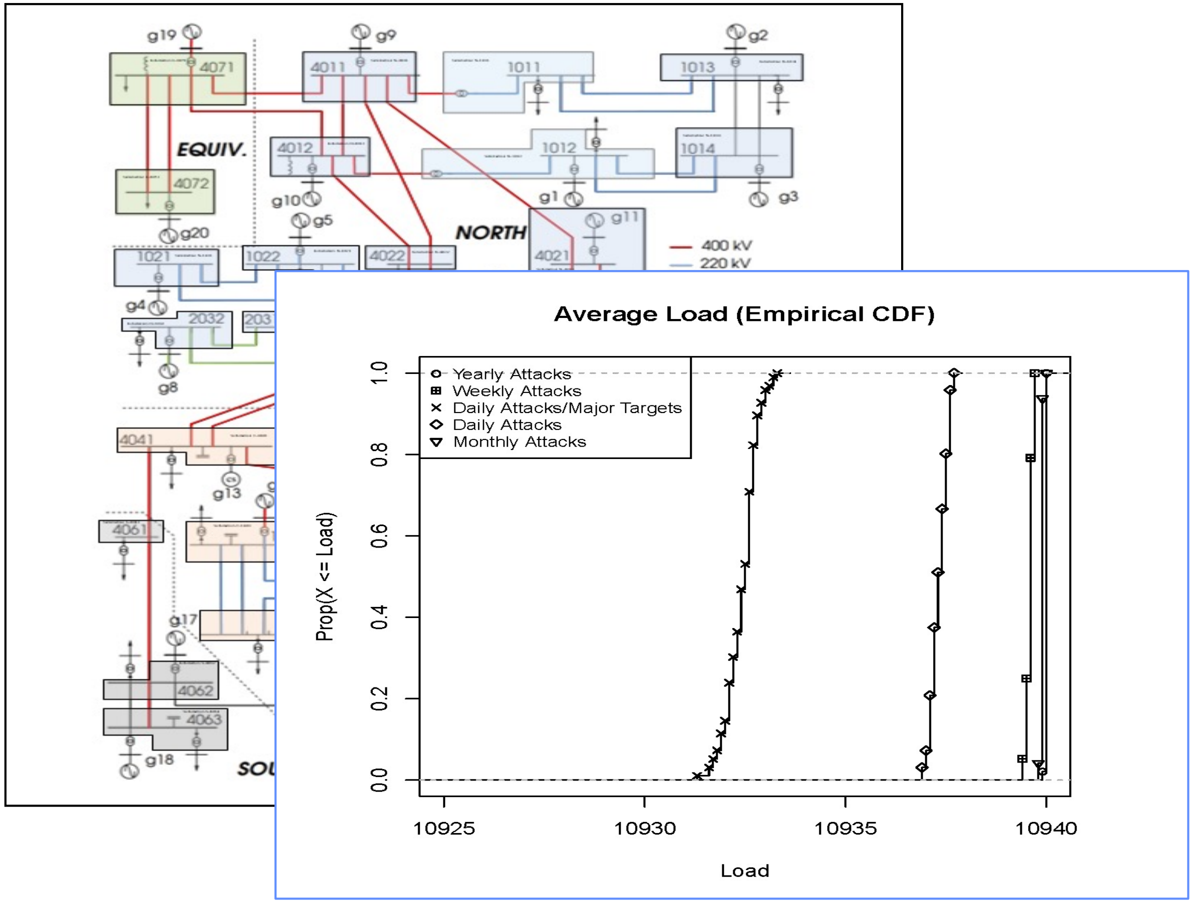
- one ideal view of how all this should be done :
  - system "design models" evolve top-down, accompanied **all along** by evolving integrated verification and certification **with** appropriate coverage of all "non-functional attributes"
- AQUAS follows another view
  - separate cultures will not integrate any time soon [or ever?]
  - "*interaction points*":
    + points in the lifecycle at which *combined* analyses (*deterministic* and *probabilistic)* support:
      * detecting the breaking of contracts agreed at earlier stages of development, or newly discovered conflicts/synergies;
      * managing trade-offs
    + frequent enough to avoid disastrous rework (or deployment)
    + starting, crucially, with early **risk analysis** stage
  - *cf* approach in automotive standard environment
    + e.g. SAE J3061 "Cybersecurity Guidebook for Cyber-Physical Vehicle Systems", new 26262 standard
    + AQUAS aims at adding practical flesh (methods, tools) on this bare-bones concept

Multi-concern modelling techniques exist: e.g. PIA- preliminary interdependency analysis

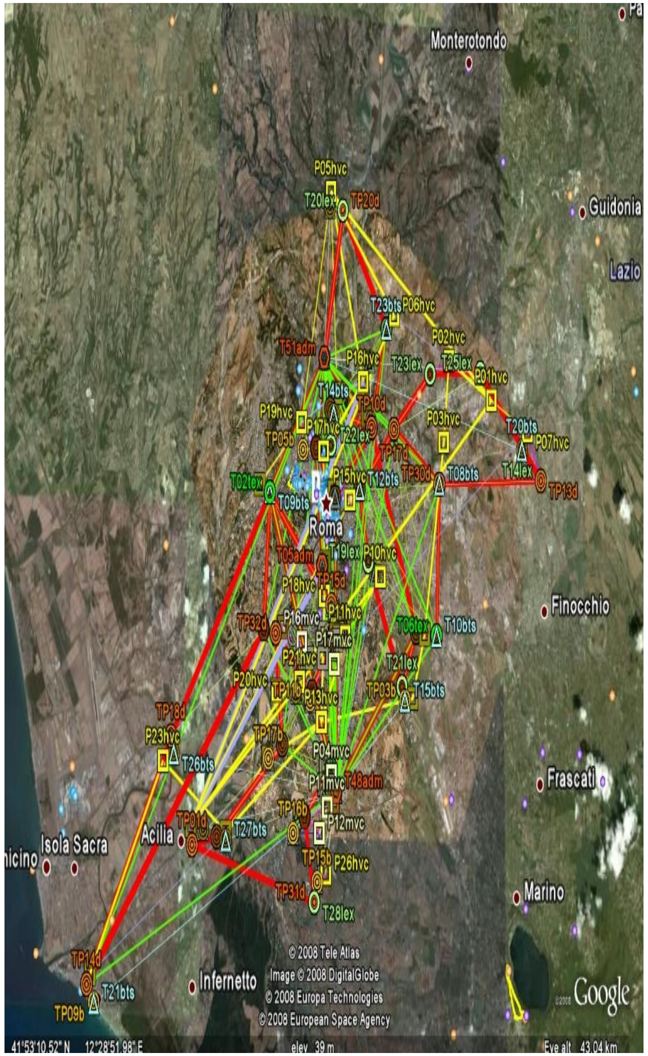
key to critical infrastructure protection - potential for cascading catastrophic failures

- method for expert analysis and translation into probabilistic assessment
- Example case studies:

Nordic 32 electrical grid control under cyber-attack



water distribution /electrical grid / telecom interdependence



see: http://openaccess.city.ac.uk/id/eprint/17456/

Now some examples...

... of culture gaps that may generate modelling gaps

... or that enlightened modelling may alleviate

Socio-technical systems, or, the human factor

- complex "SoSs" are eminently sociotechnical systems
- interspersed human and engineered components

- people
  - provide flexible, quasi-invisible fixes for glitches and design errors
  - cause some failures, e.g. through lack of global system view
  - naturally **evolve!**
    + someone needs to model/monitor for evolution, especially if *harmful*
    + e.g., safety practices degenerating into rituals, skipped when inconvenient
    + consensual violations in sub-SoSs creating dangers after mergers
    + automated aids reducing human abilities ("loss of situation awareness", "automation bias")

  - *cf* large literature on resilience, safety/security cultures etc

Loss of diversity, or creeping criticality

- partially unplanned SoSs offer redundancy
  - cf "high reliability organisations" sociology
  - people's ability to flexibly recover from glitches in their and others' performance
  - similar ideal of multiple independent systems ensuring resilience without need for centralised supervision
- but spontaneously created redundancy may spontaneously disappear
  - e.g., multiple means for navigation exist
    + but GPS offered such a convenient service that "believed to be diverse" systems might all depend on it for a time base

      cf U.S. National Timing Resilience and Security Act, 2017

(not a new phenomenon!
Cf advent of fibre backbone in telecoms; diverse software suites relying on common, successful libraries; ...)

Focus on some risks causes blind spots for others

- example Jan 2018 Hawaii "ICBM attack" false alarm
  - two options, “test missile alert”, and “missile alert” (send an alert to every mobile phone.. “seek immediate shelter, this is not a drill”
  - an accident waiting to happen, following a simple slip
  - countermanding was delayed by need for complex authorisation
  - simple HCI design mistake...
  - also an example of *focusing too much on the "main" feared event*

- an example of common blind spots: the potential for safety/security mechanisms as means for denial of service attacks
  - e.g. street kids playing the "stop the automated car" game
  - or more sinister uses, even to undermine safety

Controlling risk, or "risks"?

assessing total risk may be blind to real stakeholders' risks

- example: how safe should autonomous cars be?
  - one answer: at least as safe as the average human driver
  - if same or less number of deaths / mile driven, *who could reasonably complain?*

- well... the victims of new systematic failure modes may be right to complain:
  - these failures will "favour" certain people / circumstances
  - the switch to "safe" AVs would produce winners and losers

  - a status quo may be Pareto-optimal
  - apparent optimisation is then really a political, not just technical decision
  - modelling/measurement must deliver the measures needed for that decision

"Independent" systems

- independence between component systems is
  - a complication: how do I ensure good collective behaviour?
  - a promise: less likely to be affected by common-cause failures, hence a factor for resilience

- common fallacy:
  "they are independent, hence they will *fail independently*"
  - thus this form of resilience is overvalued:
    "if these two systems can both perform the service then the probability of service failure is the product of their two probabilities of failure"
  - nonsense: there is no reason why independent operation should bring statistically independent failures *in the same environment*!

    (and many possibilities for *causally* dependent failures, unless these SoSs were *designed to avoid them*)
  - yet independence assumptions are made in, or sneak into, many probabilistic models

Culture

We do have complex systems that work well, day in day out

- much of it is due to careful design
- much is due to *culture* (experience, shared habits, tacit knowledge, ...)

"That's all well and good in practice, but how does it work in theory? "

- That is: culture embodies evolutionary responses to what has been experienced
- it may be ill-prepared for new threats, change, rare events

- whole-system modelling, however incomplete, can support awareness of what might happen, expand views

CITY
UNIVERSITY OF LONDON
EST 1894

A possible summary?

- Yes, modelling before deployment or change is essential for insight
  - about what may happen, what should be monitored, how much good the monitoring will do
- knowledge gaps arise not only from "independence" between component systems
- ... but from gaps between cultures
  - safety and security
  - design and operation
  - social sciences and engineering
  - ICT services and users of the services
- models and measurement
  - if informed by one culture only, may be unhelpful
  - if informed by broad-minded "what may happen" analysis, they can reduce the culture gaps and improve decisions

Thank you!

Questions? Comments?

related material at:

https://openaccess.city.ac.uk/cgi/search/archive/advanced?screen=Search&dataset=archive&documents_merge=ALL&documents=&divisions=ICSWR&

http://aquas-project.eu/