Research roadmap for self-adaptive systems

”Modelling view”

Jesper Andersson, Kurt Geihs, Gabor Karzai, Sam Malek, Massimo Tivoli, *Danny Weyns*, Rogerio de Lemos, et al.

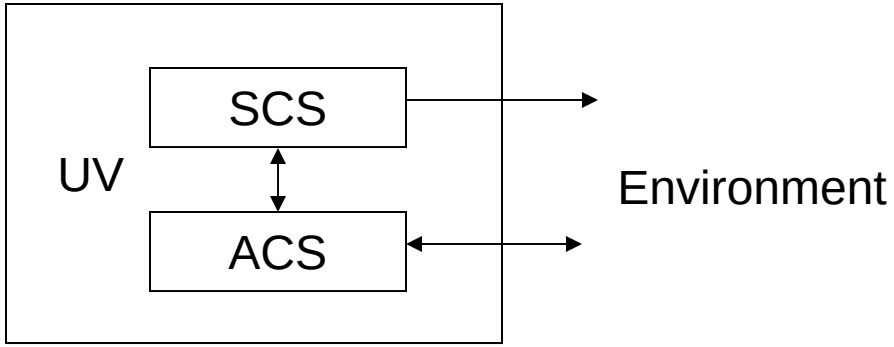
Motivation for a modelling framework

"*The need to identify and classify modelling dimensions for obtaining precise models to specify the self-adaptive problem and support runtime decision making for achieving self-adaptability*"

Approach

- Brainstorm starting from two case studies:
  - IT change management
  - Embedded application
- We identified a set of modelling dimensions
  - Each dimension describes an aspect of the system that is relevant for self-adaptation
  - We defined a domain for each dimension
  - Dimensions are grouped (adaptation activities, timing, dependability)
- We identified some challenges ahead

Illustrative case
Darpa Urban Challenge



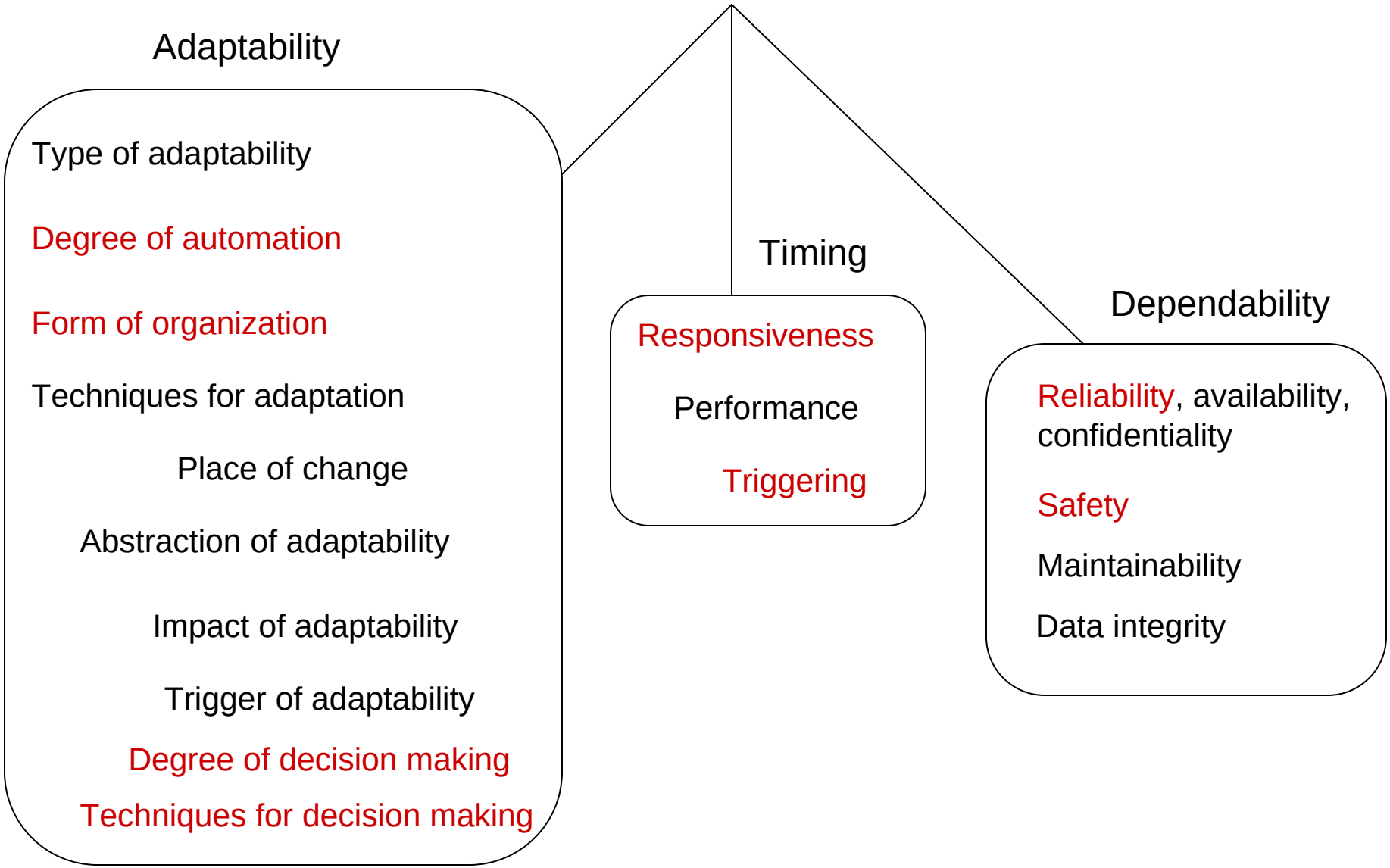
System/subsystem

UV = Unmanned vehicle

ACS = autonomous control system

SCS = self-adaptive control system

Modelling framework



Group Adaptation (excerpt)

- Degree of automation
  - Domain: autonomous …. human-based
  - UVs have to avoid collisions without human intervention
- Form of organization
  - Domain: weak/centralized … strong/decentralized
    - Strong organization
      - no global model of environment
      - components change their structure/behavior to self-adapt
    - Weak organization
      - global system model incorporating a feedback loop
      - subsystem monitors and adapts the base system
  - SCS UV seem to fit naturally with weak organization

Group Adaptation (excerpt)

- Degree of decision making
  - Domain: static/predefined …. dynamic/runtime
    - Static/predefined
      - self-adaptation scenarios are exhaustively defined before deployment
    - Dynamic
      - Decision of self-adaptation will be made during execution
  - SCS monitors environment and decides at runtime when to take over control to avoid collisions
- Techniques for decision making
  - Domain: utility functions, case-based reasoning etc.
  - SCS will likely use a reasoning-like approach to determining whether the vehicle is in collision range of an obstacle

Timing (excerpt)

- Responsiveness (answering or replying of the self-adaptation)
  - Domain: guaranteed …. best-effort
  - SCS must guarantee that the UV reacts effectively to avoid collisions
- Triggering (refers to initiation of self-adaptation process)
  - Domain: event … time
    - Event
      - Self-adaptation is triggered whenever there is a significant change in the state
    - Time
      - Process is initiated at predetermined points in time
  - Obstacles in the UV case can appear unexpectedly, thus self-adaptation is event-based

Dependability (excerpt)

- Reliability
  - Domain: low …. high
  - Reliability of the SCS avoiding collisions is expected to be high
- Safety (consequences caused by absence of self-adaptation on user and env.)
  - Domain: critical … non-critical
  - Safety in UV case catastrophic in case of failure

Some challenges ahead

- Definition of models for self-adaptability
  - Precise enough to support runtime analysis and decision making
  - Simple enough to make synthesis feasible
- Supporting decision making
  - Techniques for defining suitable utility functions that take into account runtime changes (e.g. user reqs)
  - Practical techniques for automatic generation and efficient evaluation of such utility functions

Reflection

- First shot to define suitable modelling framework for self-adaptability
- Modelling cross-cuts all phases of life-cycle
- Modelling framework can guide requirements engineers, architects, developers, etc. to model important aspects of self-adaptable system