Research Methods in CS

Hans Hansson

- Research?
- How?
- Some examples

Research - What is it?

Explore!!

Engineering Research = Explore and develop

Scientific methodology - otherwise not research
Science and Engineering
(vetenskap och teknik)

- Science - new knowledge (understand the world)
- Is Computer Science about understanding computers and computing?
- Engineering - construction of (useful) products
- Engineering Science - methods for construction (design) of products

Is CS an independent discipline?

“CS has an intimate relationship with so many other subjects that it is hard to see it as a thing in itself”
Minsky

“CS differs from the known sciences so deeply that it has to be viewed as a new species among sciences”
Hartmanis
Is Computer Science a good name?

Like calling surgery “knife-science”, maybe algorithmics is a better name

Dijkstra

“Algorithmics”

Science (e.g. anal. of alg.)

Engineering (e.g. softw. eng.)

Logic-based (Complexity theory, program semantics, concurrency)

Numerical analysis

Building HardW.

Building SoftW.

Computer Literacy is not CS!

Datakunskap/Användning av datorer är inte datavetenskap

• Computer Literacy
  – Spreadsheet
  – Programming
  – Word processing

• Computer Science
  – Algorithms
  – Programming
  – Program analysis
Computer Science

• A meta-science?
  – Methods and theories that are useful
    • for other areas of science
    • for society
  – cf. Mathematics

• Young Science!
  – Methods not well developed
  – Methods not generally known
  – Methods not always properly used

Research Method

• Ad hoc homebrew not enough
• Must be based on and related to knowledge

• Different approaches:
  – Deductive methods.
  – Inductive methods.
  – Building models. Simulation.
  – Qualitative Methods - (Hermeneutics)
    – (Claes Wohlin)

Positivistic (facts)
“Science or Interpretation”

• “Science” (naturvetenskaplig approach)
  – Theory
  – Derive a hypothesis (from theory)
  – Formulate experiment to test hypothesis
  – Perform experiment (test hypothesis)
  – Evaluate results => positive/negative evidence
  – If contradiction: modify theory

• “Interpretation” (social sciences)
  – Perform investigation (collect data)
  – Develop model (theory) that explains
  – Integrated development of theory and data collection

• Social scientists are explorers!
• Scientists are “looking for oil”!
Deductive methods

- Theoretic methodology:
  - Def. - Theorem - Proof
- Theoretical Computer Science

Inductive methods

- Experimental methods:
  - Hypothesis
  - Experiment/Evaluation => +/- evidence
- “Science”
Building Models; Simulation

- Build a model
- Validate that the model faithfully captures the relevant aspects of the modelled system
- Perform simulation experiments
- Generalize the results to the real system

Qualitative (Hermeneutic) Methods

- Interpretation (social sciences)
- Observation
- Build model to explain (theory)
- Refine … Test…Discuss…Deeper understanding
  – “Hermeneutic circle” (hopefully converging;-)
Computer Science Research Approaches
(Clæs Wholin’s SE research not really covered)

• Classification:

  - Theory
    - Complexity Theory
    - Theory for Parallel Systems
  - Basic
  - Applied
    - Formal correctness proof for algorithm
    - Evaluation of the use of a specific algorithm
  - Experiment
    - Study of the dynamics of internet traffic

Research Results

• What is a result?
  - Knowledge!!! E.g. negative results (publications)
  - algorithms and methods

• Experimental results/evaluations
  - not always proper set-up (see what happened when we tested X)
  - exhaustive experiments and comparisons important
  - evaluation methodology (statistical methods)
  - reproducibility (often forgotten in CS)
Ett (varnande;-) exempel
Snilleblixten!

• A new efficient hashing function for searching large image databases (internet => $$$$$;-)
• No problems with the patent: Let’s publish a paper!

Title: Efficient hashing for large image databases
Abstract: We’ve done it again...

1. Intro
2. The alg. (pseudocode)
3. An example (it works?)
4. Conclusions (it really works?)

Some real examples (1)

• Moldelev, Gunningberg:
  “How a large ATM MTU causes deadlock in TCP data transfer” (IEEE/ACM Tr.on Netw. 3(4) Aug 95)

• Why and when a protocol stack gets the hiccups
“How a large ATM MTU causes deadlock in TCP data transfer”

- Methodology: Traditional experimental science
  - Anomaly detected
  - Experiments to understand nature of anomaly
  - Formulation of hypothesis
  - Collection of evidence => rejection
  - Gained insights => New hypothesis
  - Experiments and study of protocol => validation
  - Generalisation (and suggestion of counter-measures)

Some real examples (2)

- Presentation and evaluation of a method for end-to-end response-time analysis for ATM
Networking (example)
Applying RT-scheduling to ATM

- ATM - telecom network
  - For voice, video and multimedia
- “Off-the-shelf”-technology
  - Interesting for both soft & hard RT
- (Hard) Real-time admission control

Two solutions:

By telecom community:
Bandwidth reservation (WFQ)
- Poor utilization (especially short deadlines)
- Fast admission control
- Suited for dynamic systems

By Real-Time community:
Priority queueing
- High utilization
- Expensive admission control
- Suited for static systems

“Response-Time Guarantees in ATM Networks”

- Methodology: “Exhaustive” evaluation of specific alg./method.
  - Extension of new theory to new (more complex) area
  - Experimental setup: Topology and Traffic profiles
  - “Exhaustive” evaluation of admission tests
    • quality measure: admission probability
    • comparison with related methods
  - Simulation to evaluate precision of admission tests
Some real examples (3)

- Arne Andersson: "General Balanced Trees"  
  (J. of Algorithms)
- Keeping search trees balanced gives faster search

Balance criteria?
- diff in height of two subtrees ≤ 1
  => max h = 1.44 log |T|
- general bal. trees (gbt) - relation between height and width of T
  (e.g. h = c \log |T|)

Maintenance?
- gbt - cheaper to maintain
  (partial rebuilding \(O(\log |T|)\))

"General Balanced Trees"

- Methodology: Fundamental algorithm theory
  - Formal definition of concepts
    - trees and partial rebuilding
  - Theorem(s) expressing maintenance cost (complexity)
    - at abstract level \(O\) and more detailed level (deriving the constant)
  - Proof of Theorem(s)
  - Comparison with weight-balanced trees
  - Hint on generalisation to multidimensional trees
Some real examples (4)


• Extension of Computation Tree Logic with probabilities (and time)

• Semantics: formulas are interpreted over discrete time Markov chains

• Verification using model checking

“A Logic for reasoning about Time and Reliability”

• Methodology: Formal design method
  – Definition of syntax and semantics (using probability measures for infinite sets of paths)
  – Model checking algorithm(s) defined
  – Correctness of algorithms proved by proving that defined and calculated measures coincide
  – Example of properties and a concrete application is modelled and verified
Some real examples (4)

“Empirical Evaluation of Usage-Based Software Inspection”, Thomas Thelin, LTH, PhD-thesis, Sept’02

- Taxi system
- Spec. Design doc
- Use cases
- Usage-based Reading (inspection)
- List of faults

Efficiency – faults found/h
Effectiveness – fraction of total faults found

Some real examples (4)

“Empirical Evaluation of Usage-Based Software Inspection”

- Methodology: Interpretation, Quantitative investigation
  - 27 3rd year students divided into 2 groups
    • Prioritised use-cases
    • Randomly ordered use-cases
  - Hypotheses
    • There is a difference in efficiency and effectiveness between the groups
    • The groups find different faults
  - “Controlled experiment”
    • Instructions
    • Inspection (limited time)
      • Standardized report form
    • Statistical evaluation of results
    • (Feedback to students)
Some real examples (4)

Empirical Evaluation of Usage-Based Software Inspection

• Analysis
  – Statistical methods
  – Significant difference!

• Validity?
  – Conclusion validity
    • Robust statistical methods
    • Independent experiments?
  – Construct validity
    • “Artificial set-up risk”
    • Use-cases developed before specification!
  – External validity
    • Are students “normal”?

• Generality?
  – More (positive) evidence needed