UNIVERSITÄT PADERBORN
Die Universität der Informationsgesellschaft

Department of Computer Science
Computer Engineering Group
www.upb.de/cs/ag-platzner

Paul Kaufmann, Marco Platzner

Multi-objective Intrinsic Evolution of Embedded Systems (MOVES)

SPP OC Berlin '07

Autonomous Embedded Systems

Self-adaptation and self-optimization by:
- intrinsic evolution
- reconfiguration of hardware
- using preevolved solutions

Environmental changes:
- Change of rate is rather slow
- compensated by the evolutionary search process

Changes in the computational resources:
- are radical changes in the area and / or delay constraints
- change rate exceeds the adaptation rate of the evolutionary algorithm
- use preevolved alternatives to fulfill the constraints

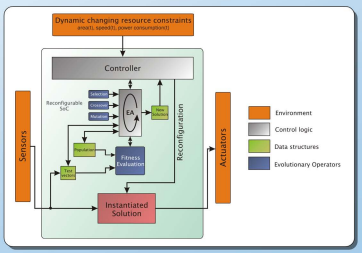
Autonomous operation mode:
- all algorithms run on the embedded system

Architecture

The fitness function changes over time (slow changes)
- the test vectors are taken from the environment
- offline mode: separate evolution and operation phases
- online mode: evolving while in operation

The controller sets the resource constraints (rapid changes):
- for both the instantiated solution and the evolution process
- utilizes problem- and system-specific knowledge

Implementation on reconfigurable system-on-chip:
- platform FPGA including CPU and logic
- self-reconfiguration

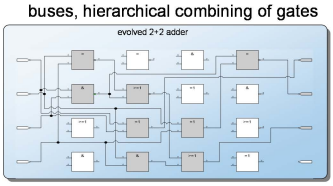


Models and Algorithms

Cartesian Genetic Programming Model (CGP)

- array of combinational blocks connected by feed-forward wires of maximal length l (levels back parameter)
- chromosome defines configuration of the array
- model similar to [Miller an Thomson '96]

Challenges:
- efficient mapping chromosomes → FPGA bitstream: restricted model
- scalability: more complex operators, buses, hierarchical combining of gates



Optimization for multiple objectives

- example circuit design: speed vs. area vs. power consumption
- often, the objectives are conflicting which leads to compromises
- evolutionary algorithms generate Pareto fronts (set of non-dominated solutions)
- Goal: diversity in the Pareto front

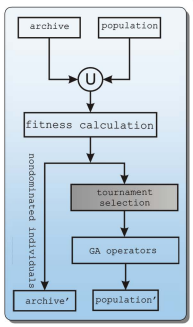
What is an objective, what is a constraint?

- some functions have a functional quality, e.g. the key distribution of a hashing function
- other functions have a correctness property, e.g. arithmetic functions
  - functional quality is a constraint rather than an objective

SPEA2 algorithm

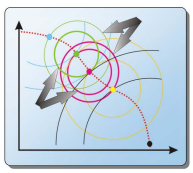
[Zitzler, Laumanns and Thiele, '01]

- Pareto dominance based selection scheme
  - dominance count + rank
  - density information
- maintains diversity by dissolving pareto front clusters with k-th nearest neighbor density estimation



TSPEA2 algorithm

- prioritize constrained objectives to speedup the convergence, but
- keep the diversity preserving mechanism of SPEA2
  - selection scheme similar to [Trefzer, Langeheine, Meier and Schemmel, '05]



Case Studies

6-parity function

Objectives:
- functional quality
- area
- speed

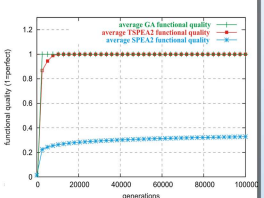
Correctness well-defined

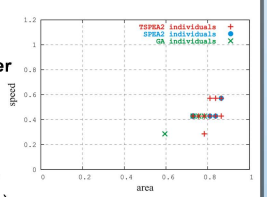
CGP model:
- 6x6 2-LUT array
- levels back parameter l=3

Results:

6-parity evolved on average after
- 36322 generations by SPEA2
- 903 generations by TSPEA2
- 63 generations by GA

SPEA2 and TSPEA2 have similar Pareto fronts (quality and diversity)

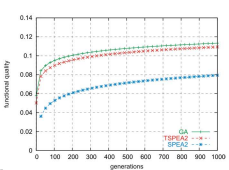


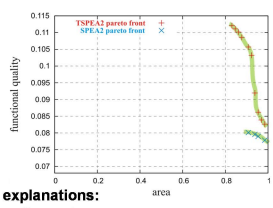


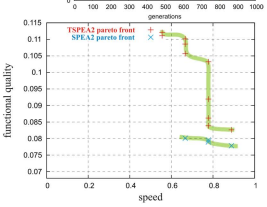
Hashing function

- find a function that maps 2^12 keys to 2^8 indices in the most uniform way possible
- no correctness property, just functional quality
- CGP model: 8x8 4-LUT, l=8

Results:
- TSPEA2 outperforms SPEA2 in quality, area and speed







Possible explanations:
- rather small examples
- there are not too many possible objective values combinations
- the objectives are not necessarily conflicting

Outlook:
- speed-up the convergence by using problem-specific objective dependencies and continuously adjustment of the prioritizing weights by a second EA