Marching Pixels

seit 1548

Friedrich-Schiller-Universität Jena
Institut für Informatik
Technische Informatik II

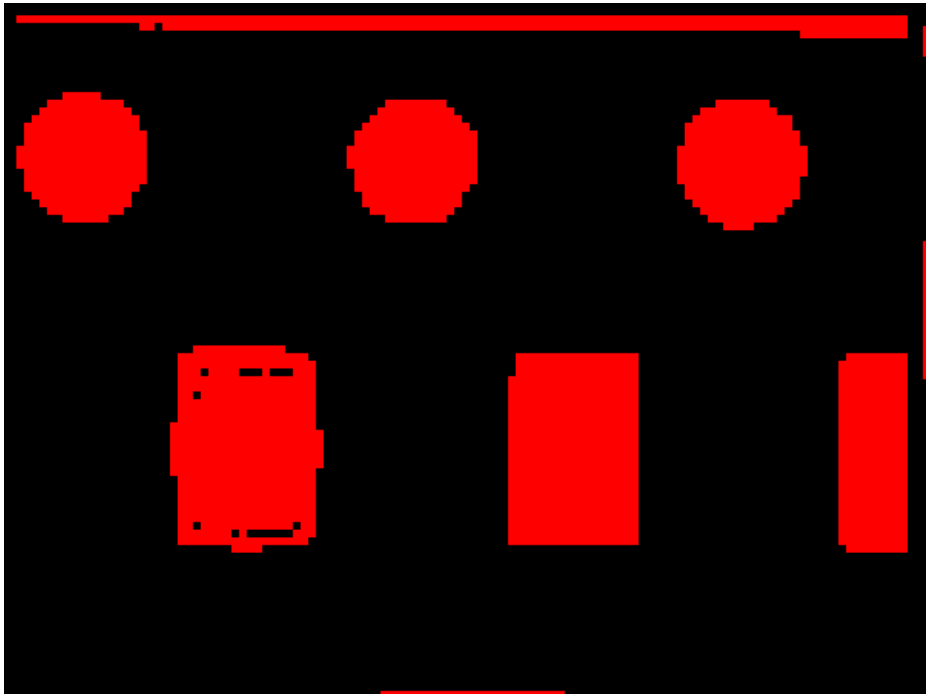
Marcus Komann

Agenda

- OC Toolbox for Marching Pixels as basis for the second phase
- Prototype chip
- Generic hardware for emergent algorithms
  - Idea
  - Pipeline implementation - Multiplexing
  - Reliabilty and robustness
- Evolution of emergent programs
  - Motivation
  - Results
  - Evolutionary framework

Marching Pixels Toolbox
Basis for the second phase

- Centroids, Area, Rotation
- Gaphop, Flooding, Opposite Flooding
- Emergent running schemes
- With different capabilities
- Moment calculation
- Synthesised and tested

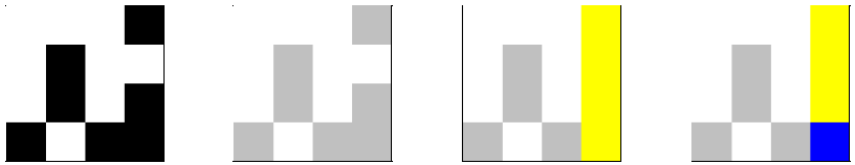


M. Komann

Universität Jena –
Inst. f. Informatik

Toolbox

- No agents but waves
- All positions work simultaniously
- Result emerges after time



ms		sum_x
z_y	pix	m_y
n_y		s_y

		1			2			3			3
0	1	1	0	1	2	0	1	3	0		3
0		0	0		0	0		0	0		0
—		1	—		1	—		2	—		3
0	1	2	0		3	0	1	5	0	1	6
9		1	9		2	9		3	9		3
		1			1			2			2
8	1	3	8		4	8	1	7	8		8
9		3	9		5	9		8	9		9
		0			1			1			1
17		3	17	1	5	17		8	17		9
9		6	9		9	9		15	9		17

Toolbox

- Virtex-II Pro FPGA

	Flooding	Opposite-Flooding
parallel	182,930 MHz	195,478 MHz
seriell	143,798 MHz	157.587 MHz

- Parallel Flooding Pics per second (150 MHz):

PFW x PFH	MCPL	PPS
8 x 16	48	3.125.000
1024 x 1024	4096	36.621
2048 x 1536	7168	20.926

- ASIC: todo

Toolbox – Path planning

M. Komann

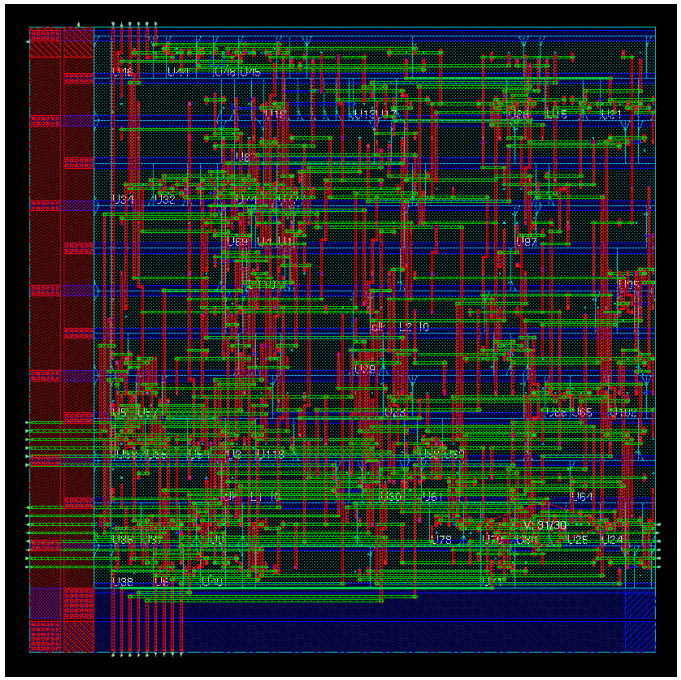
Universität Jena –
Inst. f. Informatik

Status Colloquium

Zürich
September 2008

Prototype chip

- Flooding algorithm
- Specified in VHDL
- Tested and validated on FPGA
- Observer processor is a Microblaze on a FPGA
- Real chip in next 10 months
- Size: todo



Generic hardware - Idea

- Generalising Marching Pixels idea
- Generic array of generic processing elements
- For emergent algorithms
- No central control
- High potential for future

Generic hardware - Realisation

- VHDL
- Parameterized array:
  - Size
  - Number of PEs
  - Neighbourhood/Connectivity
  - Data width
- Parameterized PEs
  - State machine (propagation)
  - Registers, Operations
- Programmable PEs
  - Microprogram
  - Conditions

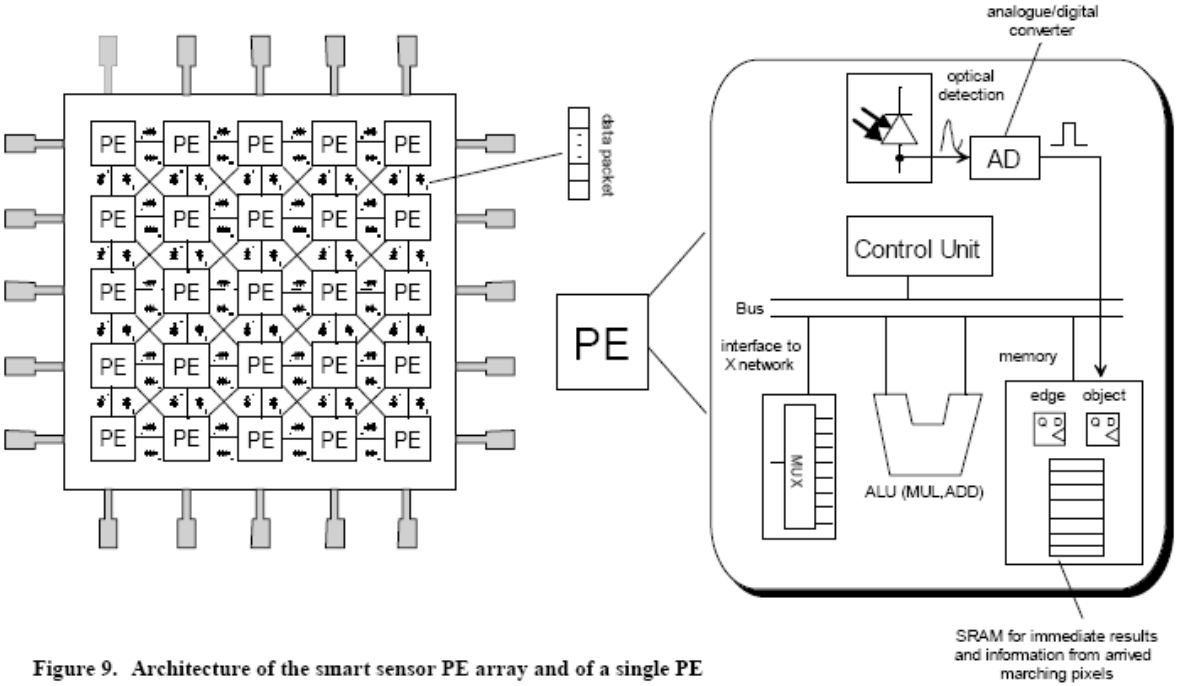


Figure 9. Architecture of the smart sensor PE array and of a single PE

M. Komann

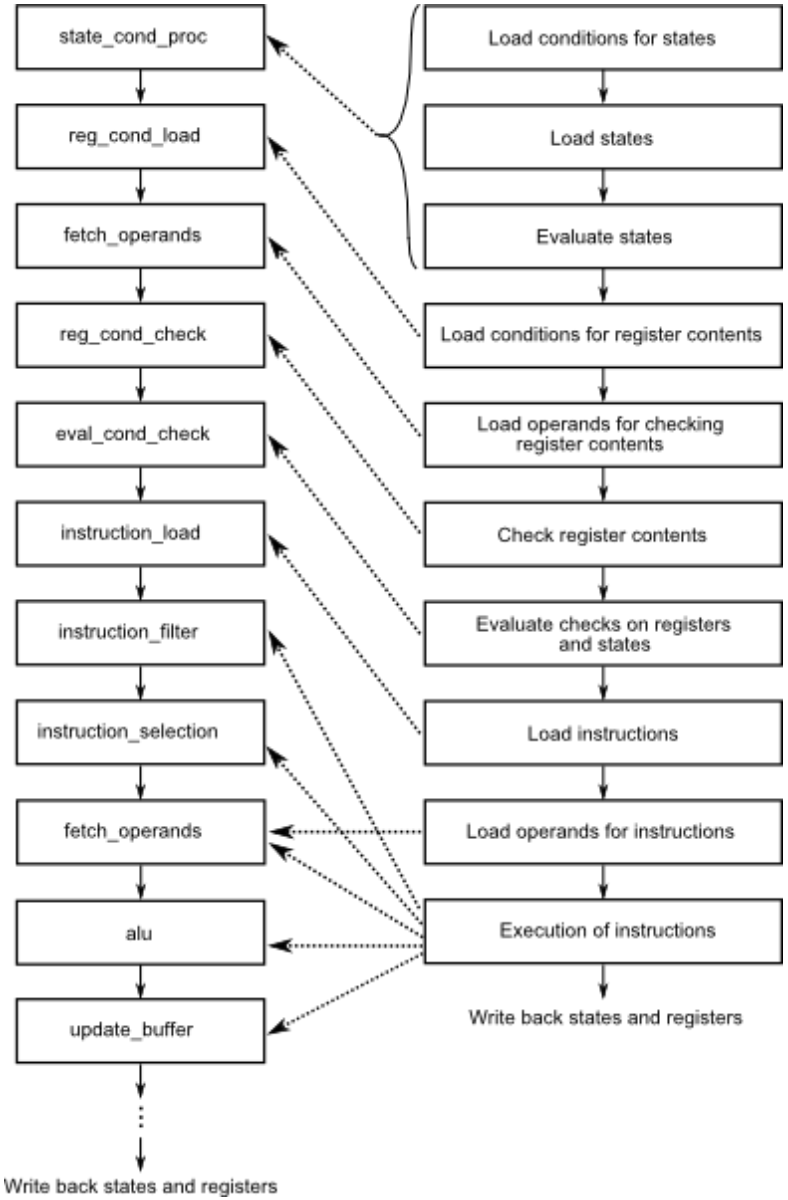
Universität Jena – Inst. f. Informatik

Status Colloquium

Zürich
September 2008

Generic hardware - Pipelines

- Massive parallelism grants enough compute power
- Trade time for chip area -> time multiplexing
- 90 nm UMC process:
  - 500 MHz
  - 0.66 mm^2



Generic hardware - Reliability

- Agents deliver some degree of robustness
- Question: Cause transient errors system breakdown?
- Idea in cooperation with Tübingen –> Shadow registers / redundancy
- Answer: Depends on algorithm and stage
- Neuralgic points identified

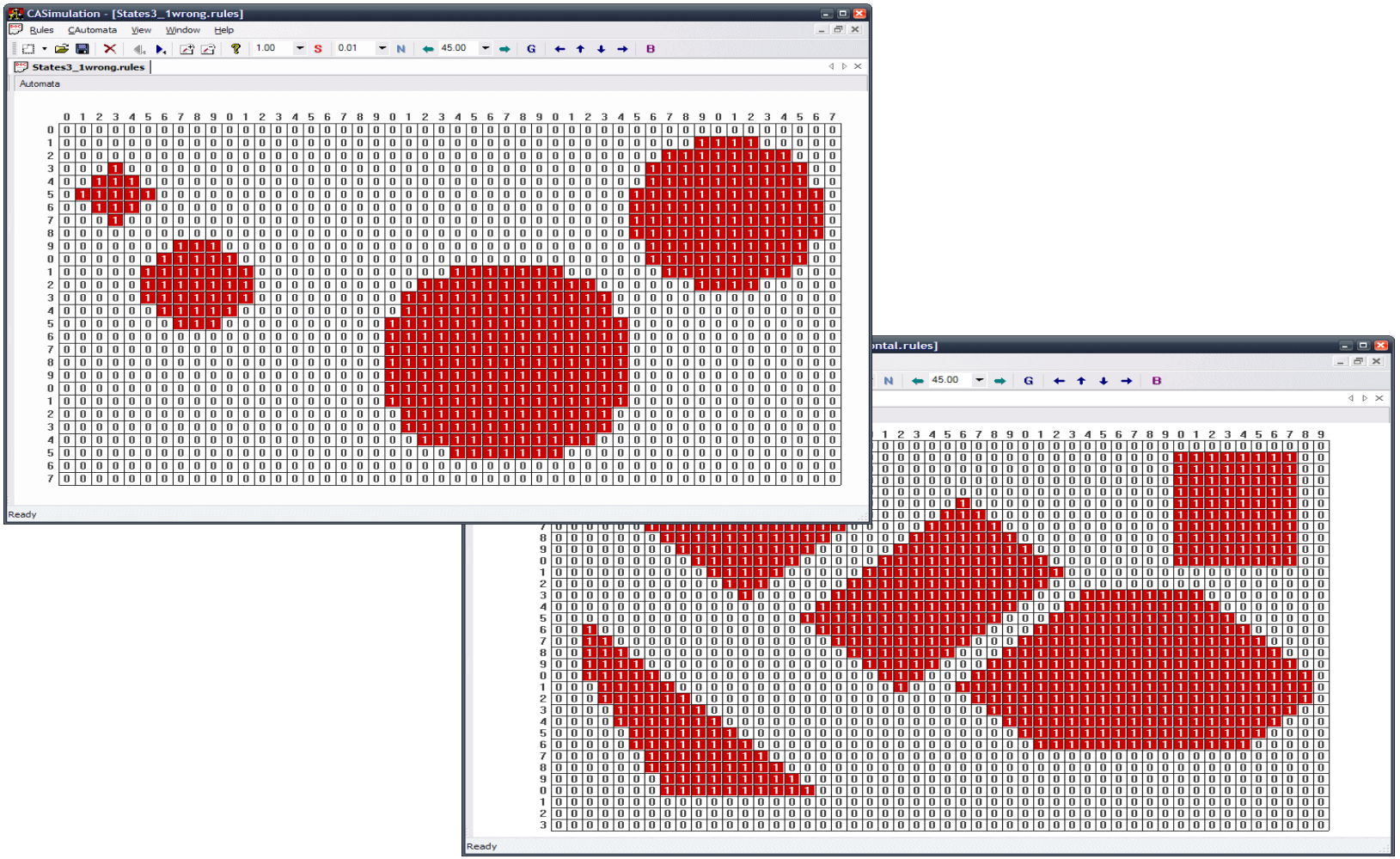
Reliability - Results

- 90nm UMC process - 14407 transient errors injected

	Errors			Costs	
Stages protected	State	Register	Sum	relative	absolute
none	1189	1178	2367	515285	100,00%
eval_cond_check	554	545	1099	516056	100,15%
instruction_selection	237	263	500	535731	103,97%
state_cond_proc	209	142	351	537927	104,39%
reg_cond_check	149	136	285	549712	106,68%
fetch_operands	33	78	111	574228	111,44%
Alu	0	14	14	589974	114,49%
instruction_filter	7	0	7		
instruction_load	0	0	0		
reg_cond_load	0	0	0		
all	0	0	0	606875	117,77%

Evolving emergence - Motivation

- Genetic algorithms have problems
  => Genetic Programming



M. Komann

Universität Jena –
Inst. f. Informatik

Status Colloquium

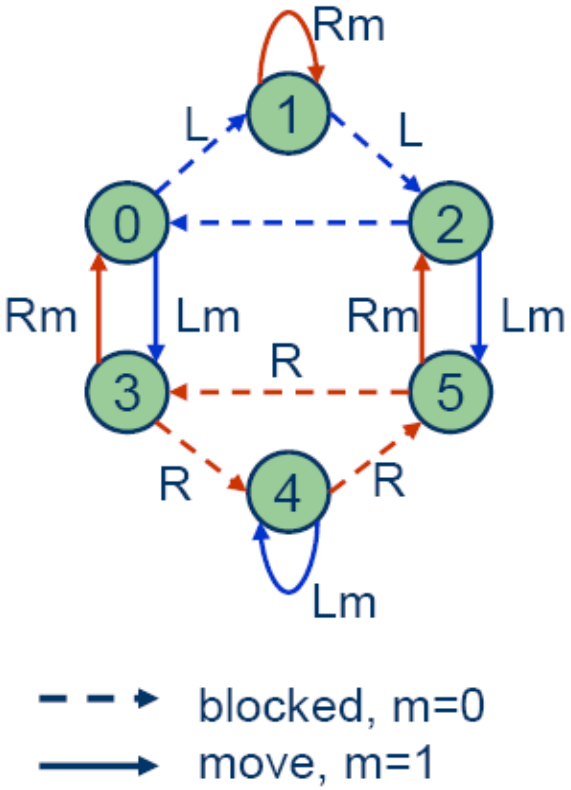
Zürich
September 2008

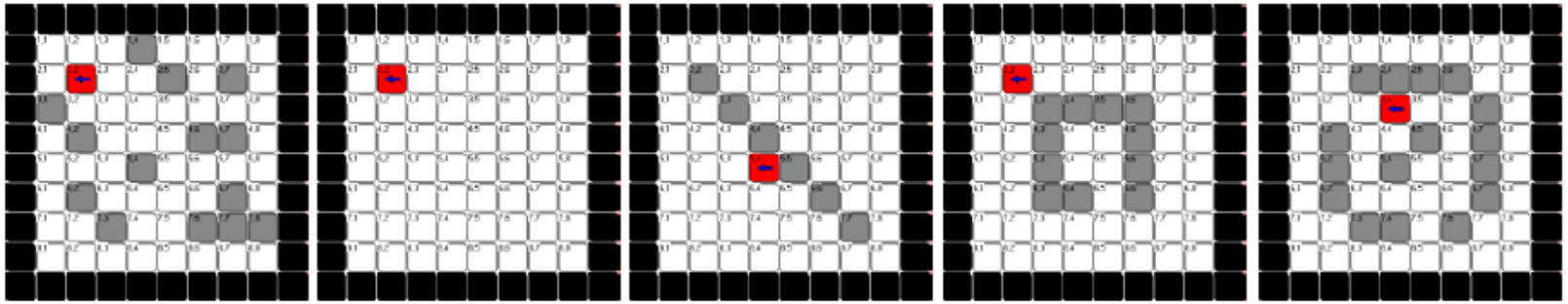
Evolving emergence - Motivation

- Generalising Marching Pixels idea
- Creatures‘ exploration problem
- Question: Best 6-state-automaton?
- 8.92 x 10^{12} possible automata

-> simulation of all expensive

- Simulating more states impossible
- Evolution helpful?





M. Komann

Universität Jena –
Inst. f. Informatik

Status Colloquium

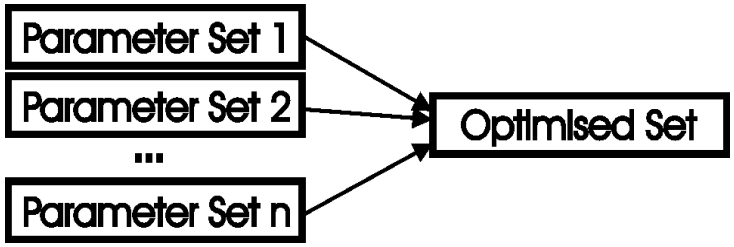
Zürich
September 2008

Evolving emergence - Results

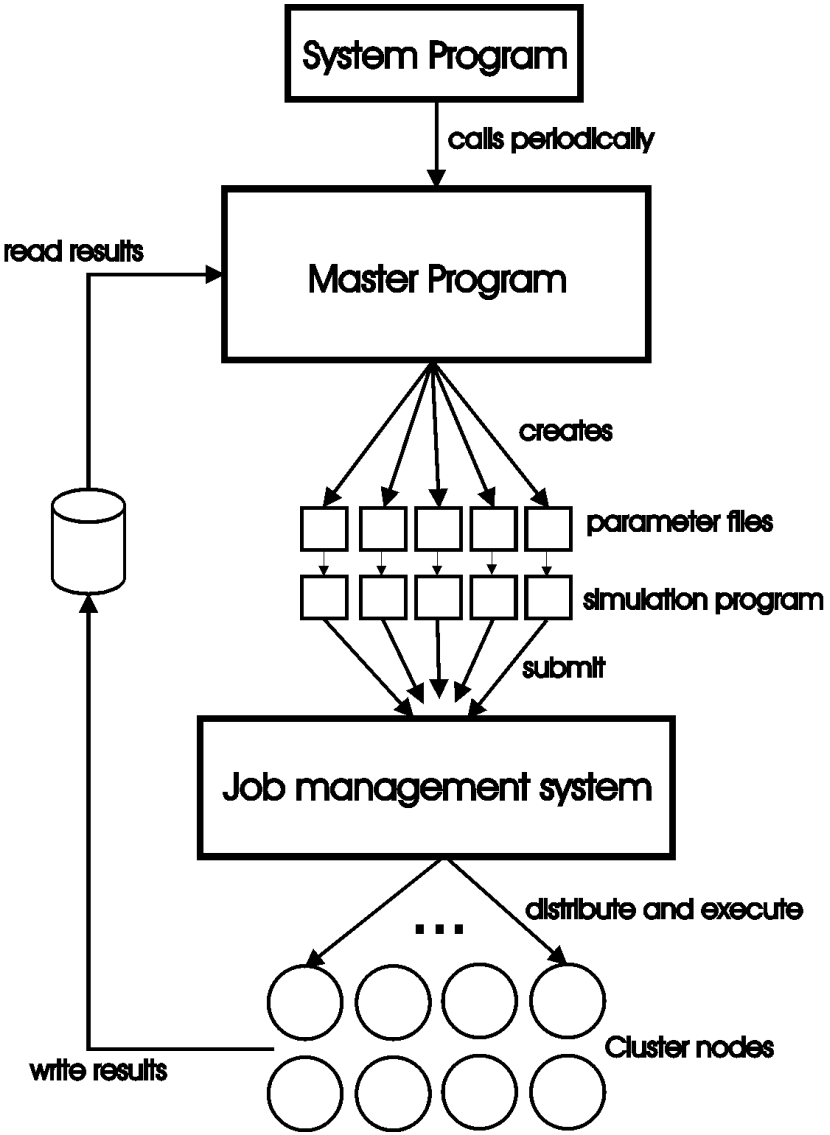
- One agent pretty good (90% food)
- Two agents better (100% food, fast)
- Time becomes issue
- Evolving more states easy
- Programs robust against distortion

Evolving emergence - Framework

- Automated evolution



- Finding desired emergent behaviour
- In cluster and in GRID
- Open for other groups!
- Todo: screenshot gridsphere



Outlook

- Prototype chip implementation for application specific MP algorithm
- Evolution with more agents
- Evolution of different problems
  - More path planning
  - Gesture recognition
- Applications on framework
- Connect framework to generic hardware

The end

Questions?

Designparameter	Wert	Bedeutung
C_MOORE_SIZE	3	Grad der Nachbarschaft
C_STATE_LEN	4	Bitbreite eines Zustandes
C_GLOB_STATE_LEN	2	Bitbreite des globalen Zustandes
C_DATA_SIZE	20	Bitbreite eines Registers
C_CELL_REGS	5	Zahl der Register einer Zelle
C_STATE_COND_TABLE_SIZE	64	Zeilen in der Zustandsbedingungstabelle
C_REG_COND_TABLE_SIZE	16	Zeilen in der Registerbedingungstabelle
C_INS_TABLE_SIZE	32	Zeilen in der Befehlstabelle
C_REG_COND_SECT_SIZE	4	in einem Zeitschritt für eine Zelle überprüfbaren Bedingungen an Registerinhalte
C_INS_SECT_SIZE	7	in einem Zeitschritt für eine Zelle ausführbare Befehle
C_PROC_BLOCK	2	Anzahl der Befehlsverarbeitungseinheiten
C_ALU_CLASS_A_PORTS	2	durch eine ALU gleichzeitig ausführbare Klasse A Befehle
C_ALU_CLASS_B_PORTS	2	durch eine ALU gleichzeitig ausführbare Klasse B Befehle
C_ALU_CLASS_C_PORTS	0	durch eine ALU gleichzeitig ausführbare Klasse C Befehle

Komponente	A $[mm^2]$	Anzahl	A_{Total} $[mm^2]$	Anteil [%]
stage_cond_proc	0, 179413	1	0, 179413	24, 88
reg_cond_load	0, 030926	1	0, 030926	4, 29
fetch_operands	0, 023958	12	0, 287496	39, 86
reg_cond_check	0, 001340	4	0, 005360	0, 74
eval_cond_check	0, 000345	1	0, 000345	0, 05
instruction_load	0, 090882	1	0, 090882	12, 60
instruction_filter	0, 026721	2	0, 053442	7, 41
instruction_selection	0, 019733	2	0, 039466	5, 47
alu	0, 007475	2	0, 014950	2, 07
update_buffer	0, 009488	2	0, 018976	2, 63
Gesamt	-	-	0, 721256	100