Optimisation and Robustness of Cellular Neural Networks

Ph.D. Defence

Samuel Xavier de Souza

Promoters
Prof. Johan Suykens
Prof. Joos Vandewalle

Katholieke Universiteit Leuven
Departement Elektrotechniek (ESAT)
SCD - SISTA

June 11, 2007

Outline

1. Introduction
2. Robust CNN Tuning and Learning
3. Coupled Simulated Annealing
4. Applying CSA to CNN Optimisation
5. Conclusions and Challenges for Future Work

Outline

1 Introduction
- The potential of Cellular Neural Networks
- How does a CNN work?
- Goals of this thesis

2 Robust CNN Tuning and Learning

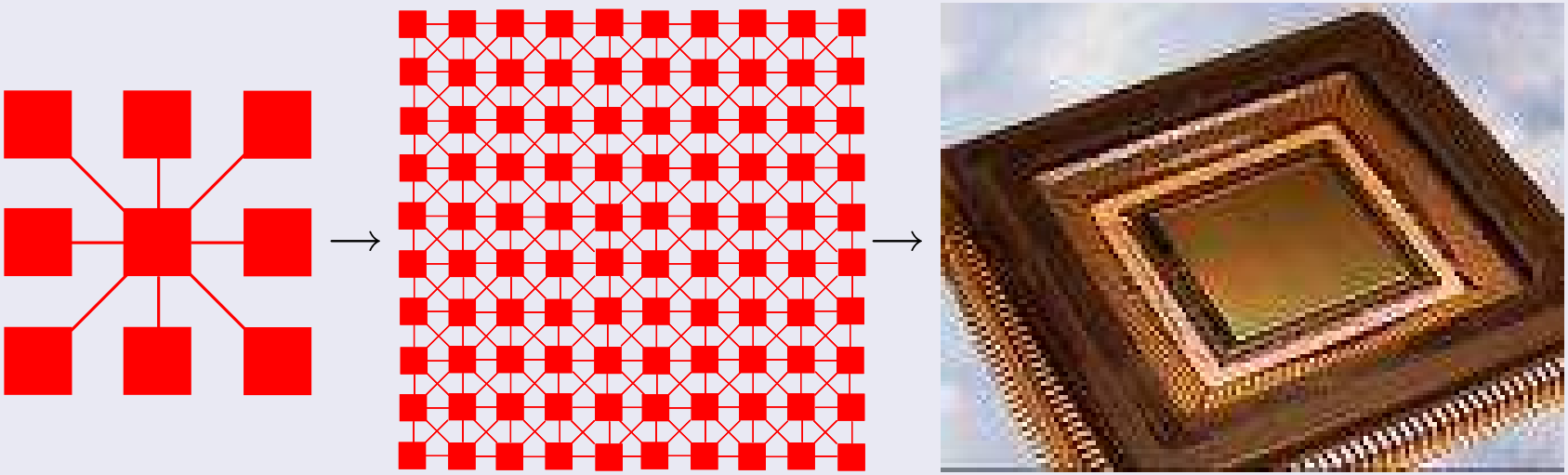
3 Coupled Simulated Annealing

4 Applying CSA to CNN Optimisation

5 Conclusions and Challenges for Future Work

CNN as an alternative to classical digital computation

Cells replication



Local connections → Large number of cells → High parallelism

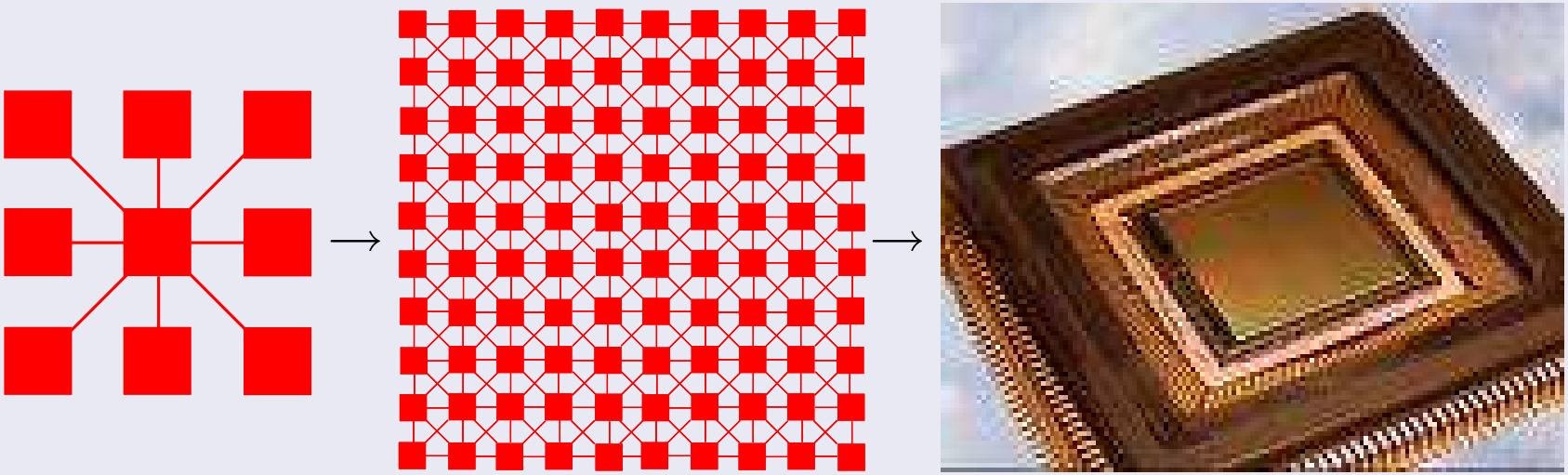
Local connections & analog signals → Low area & Low power

High parallelism → supercomputer power → order of Tera ops.

[Chua and Yang, IEEE TCAS, 1988]

CNN as an alternative to classical digital computation

Cells replication

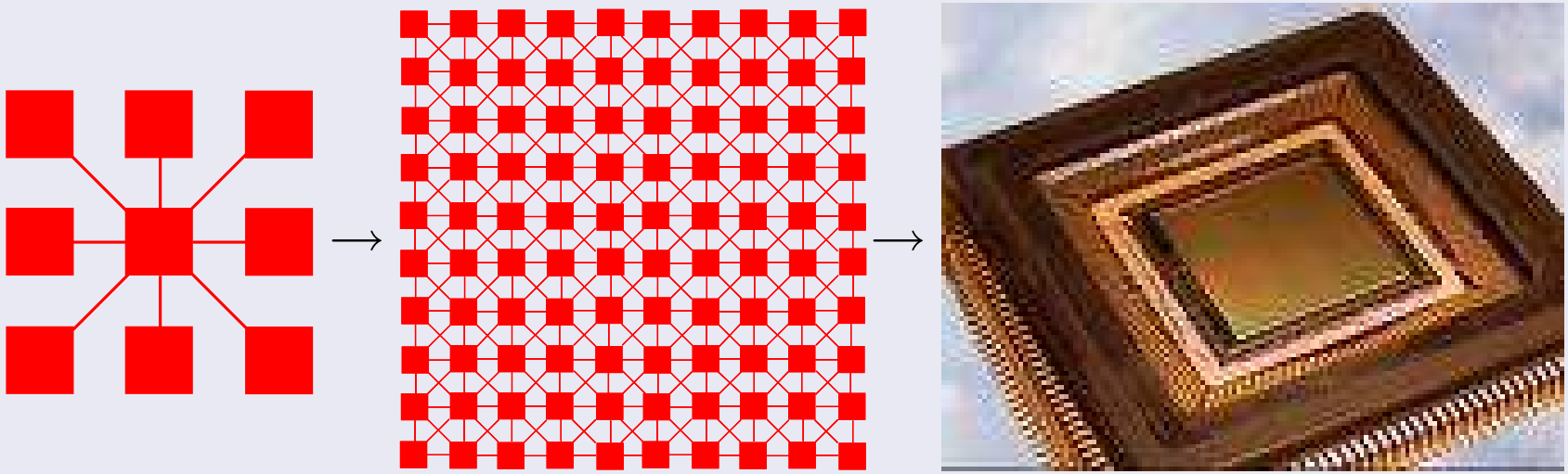


Local connections → Large number of cells → High parallelism
Local connections & analog signals → Low area & Low power
High parallelism → supercomputer power → order of Tera ops.

[Chua and Yang, IEEE TCAS, 1988]

CNN as an alternative to classical digital computation

Cells replication

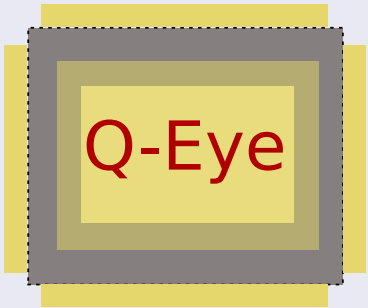


Local connections → Large number of cells → High parallelism
Local connections & analog signals → Low area & Low power
High parallelism → supercomputer power → order of Tera ops.

[Chua and Yang, IEEE TCAS, 1988]

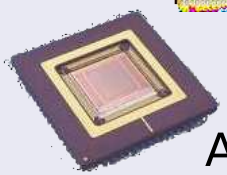
CNN are very suitable for image processing

Evolution of CNN chips

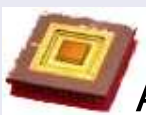


ACE16k





ACE4k



ACE400

New Q-Eye CNN chip

176×144 cells

$> 10,000$ fps

$< 100mW$

- Early vision processing
- Redundant image data $\longrightarrow$ relevant information
- Wide application range:

Visual applications demanding high throughput speed

A wide range of applications

- Ultra-high speed event/fault monitoring in production lines
- Smart security cameras
- Automotive applications such as
  - smart airbag deployment
  - blind spot detection
  - navigation
  - collision detection
- 3D Echocardiogram
- Vision for industrial/domestic robots
- Object tracking
- Eye tracking and gaze estimation
- Visual game/computer interfaces
- Handsfree wheelchair driving, etc.

A wide range of applications

- Ultra-high speed event/fault monitoring in production lines
- Smart security cameras



- Automotive applications such as
  - smart airbag deployment
  - blind spot detection
  - navigation
  - collision detection
- 3D Echocardiogram
- Vision for industrial/domestic robots
- Object tracking
- Eye tracking and gaze estimation
- Visual game/computer interfaces
- Handsfree wheelchair driving, etc.

A wide range of applications

- Ultra-high speed event/fault monitoring in production lines
- Smart security cameras
- Automotive applications such as
  - smart airbag deployment
  - blind spot detection
  - navigation
  - collision detection
- 3D Echocardiogram
- Vision for industrial/domestic robots
- Object tracking
- Eye tracking and gaze estimation
- Visual game/computer interfaces
- Handsfree wheelchair driving, etc.







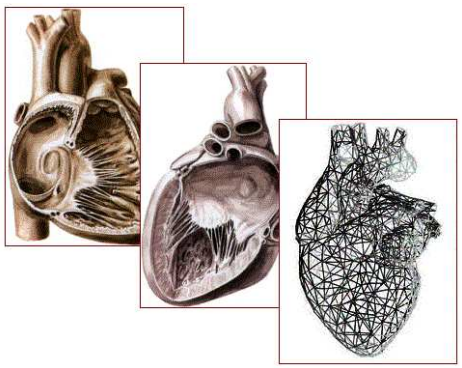
A wide range of applications

- Ultra-high speed event/fault monitoring in production lines
- Smart security cameras
- Automotive applications such as
  - smart airbag deployment
  - blind spot detection
  - navigation
  - collision detection
- 3D Echocardiogram
- Vision for industrial/domestic robots
- Object tracking
- Eye tracking and gaze estimation
- Visual game/computer interfaces
- Handsfree wheelchair driving, etc.









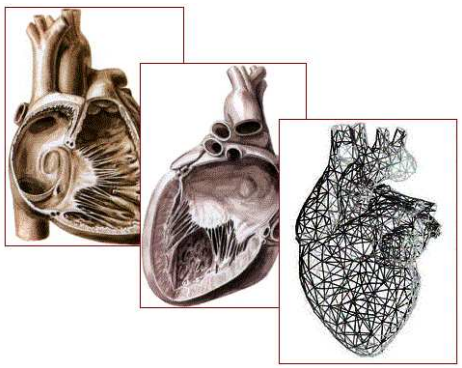
A wide range of applications

- Ultra-high speed event/fault monitoring in production lines
- Smart security cameras
- Automotive applications such as
  - smart airbag deployment
  - blind spot detection
  - navigation
  - collision detection
- 3D Echocardiogram
- Vision for industrial/domestic robots
- Object tracking
- Eye tracking and gaze estimation
- Visual game/computer interfaces
- Handsfree wheelchair driving, etc.









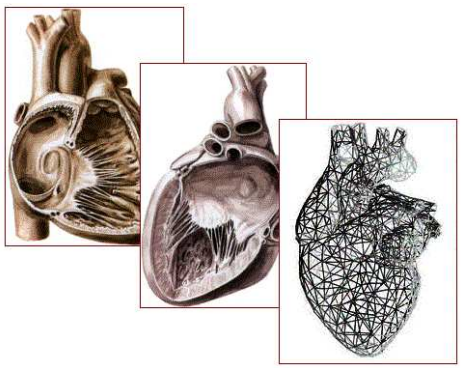
A wide range of applications

- Ultra-high speed event/fault monitoring in production lines
- Smart security cameras
- Automotive applications such as
  - smart airbag deployment
  - blind spot detection
  - navigation
  - collision detection
- 3D Echocardiogram
- Vision for industrial/domestic robots
- Object tracking
- Eye tracking and gaze estimation
- Visual game/computer interfaces
- Handsfree wheelchair driving, etc.









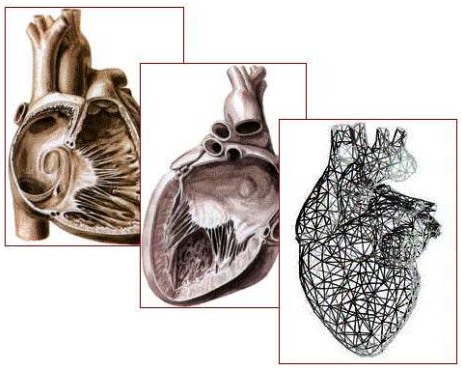
A wide range of applications

- Ultra-high speed event/fault monitoring in production lines
- Smart security cameras
- Automotive applications such as
  - smart airbag deployment
  - blind spot detection
  - navigation
  - collision detection
- 3D Echocardiogram
- Vision for industrial/domestic robots
- Object tracking
- Eye tracking and gaze estimation
- Visual game/computer interfaces
- Handsfree wheelchair driving, etc.









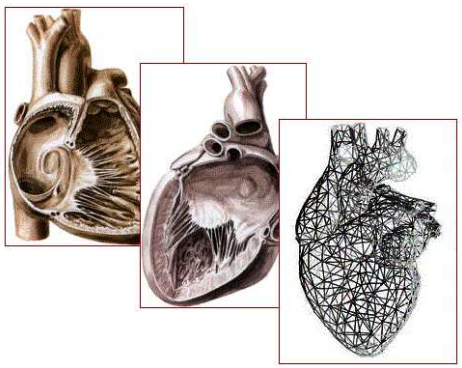
A wide range of applications

- Ultra-high speed event/fault monitoring in production lines
- Smart security cameras
- Automotive applications such as
  - smart airbag deployment
  - blind spot detection
  - navigation
  - collision detection
- 3D Echocardiogram
- Vision for industrial/domestic robots
- Object tracking
- Eye tracking and gaze estimation
- Visual game/computer interfaces
- Handsfree wheelchair driving, etc.









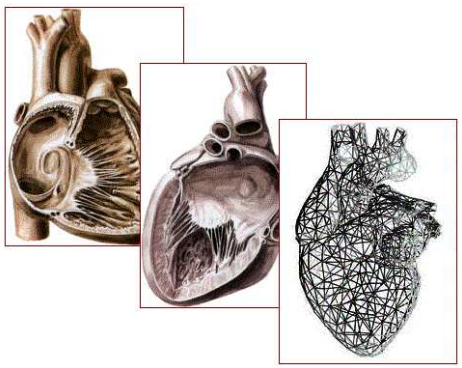
A wide range of applications

- Ultra-high speed event/fault monitoring in production lines
- Smart security cameras
- Automotive applications such as
  - smart airbag deployment
  - blind spot detection
  - navigation
  - collision detection
- 3D Echocardiogram
- Vision for industrial/domestic robots
- Object tracking
- Eye tracking and gaze estimation
- Visual game/computer interfaces
- Handsfree wheelchair driving, etc.









Outline

1 Introduction
- The potential of Cellular Neural Networks
- How does a CNN work?
- Goals of this thesis

2 Robust CNN Tuning and Learning

3 Coupled Simulated Annealing

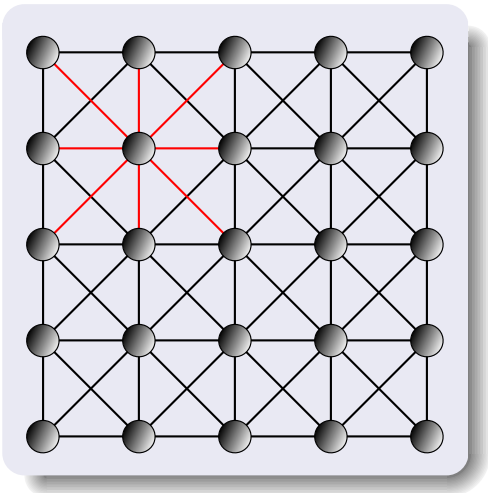
4 Applying CSA to CNN Optimisation

5 Conclusions and Challenges for Future Work

A template controls space-invariant CNNs

The equation of a cell

$$\frac{dx_{i,j}}{dt} = -x_{i,j} + \sum_{|k,l|\leqslant r} a_{k,l} y_{i-k,j-l} + \sum_{|k,l|\leqslant r} b_{k,l} u_{i-k,j-l} + z, \text{ with } y_{i,j}(t) = f(x_{i,j})$$



- A template specifies a coupling law
- 19 template values $(9 + 9 + 1)$
- Controls the entire network

CNN computer

Template

Digital computer

Instruction

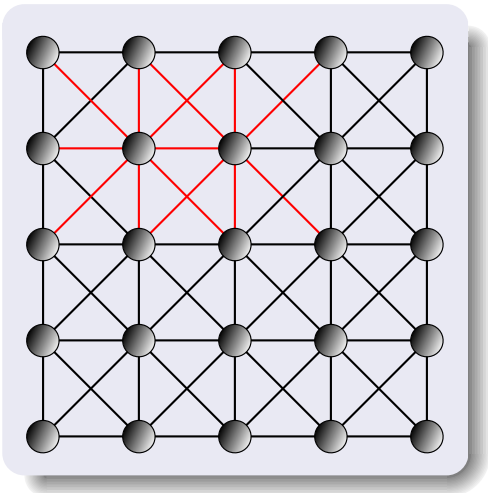
CNN template values

$$A = \begin{bmatrix} a_{-1,-1} & a_{-1,0} & a_{-1,1} \\ a_{0,-1} & a_{0,0} & a_{0,1} \\ a_{1,-1} & a_{1,0} & a_{1,1} \end{bmatrix}, B = \begin{bmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & b_{0,0} & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{bmatrix}, z$$

A template controls space-invariant CNNs

The equation of a cell

$$\frac{dx_{i,j}}{dt} = -x_{i,j} + \sum_{|k,l|\leqslant r} a_{k,l} y_{i-k,j-l} + \sum_{|k,l|\leqslant r} b_{k,l} u_{i-k,j-l} + z, \text{ with } y_{i,j}(t) = f(x_{i,j})$$



- A template specifies a coupling law
- 19 template values $(9 + 9 + 1)$
- Controls the entire network

CNN computer

Template

Digital computer

Instruction

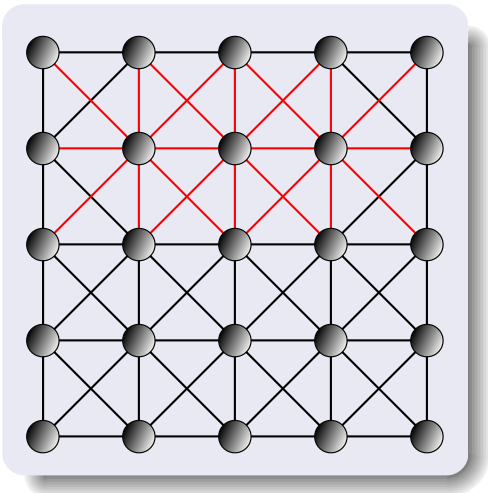
CNN template values

$$A = \begin{bmatrix} a_{-1,-1} & a_{-1,0} & a_{-1,1} \\ a_{0,-1} & a_{0,0} & a_{0,1} \\ a_{1,-1} & a_{1,0} & a_{1,1} \end{bmatrix}, B = \begin{bmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & b_{0,0} & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{bmatrix}, z$$

A template controls space-invariant CNNs

The equation of a cell

$$\frac{dx_{i,j}}{dt} = -x_{i,j} + \sum_{|k,l|\leqslant r} a_{k,l} y_{i-k,j-l} + \sum_{|k,l|\leqslant r} b_{k,l} u_{i-k,j-l} + z, \text{ with } y_{i,j}(t) = f(x_{i,j})$$



- A template specifies a coupling law
- 19 template values $(9 + 9 + 1)$
- Controls the entire network

CNN computer

Template

Digital computer

Instruction

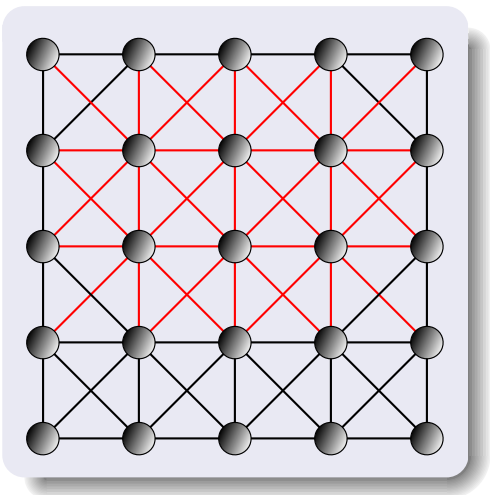
CNN template values

$$A = \begin{bmatrix} a_{-1,-1} & a_{-1,0} & a_{-1,1} \\ a_{0,-1} & a_{0,0} & a_{0,1} \\ a_{1,-1} & a_{1,0} & a_{1,1} \end{bmatrix}, B = \begin{bmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & b_{0,0} & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{bmatrix}, z$$

A template controls space-invariant CNNs

The equation of a cell

$$\frac{dx_{i,j}}{dt} = -x_{i,j} + \sum_{|k,l|\leqslant r} a_{k,l} y_{i-k,j-l} + \sum_{|k,l|\leqslant r} b_{k,l} u_{i-k,j-l} + z, \text{ with } y_{i,j}(t) = f(x_{i,j})$$



- A template specifies a coupling law
- 19 template values $(9 + 9 + 1)$
- Controls the entire network

CNN computer

Template

Digital computer

Instruction

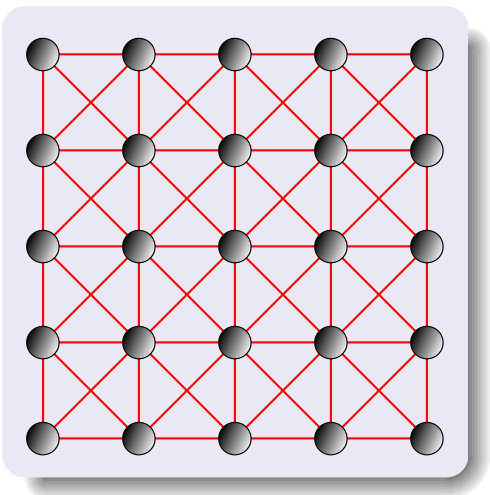
CNN template values

$$A = \begin{bmatrix} a_{-1,-1} & a_{-1,0} & a_{-1,1} \\ a_{0,-1} & a_{0,0} & a_{0,1} \\ a_{1,-1} & a_{1,0} & a_{1,1} \end{bmatrix}, B = \begin{bmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & b_{0,0} & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{bmatrix}, z$$

A template controls space-invariant CNNs

The equation of a cell

$$\frac{dx_{i,j}}{dt} = -x_{i,j} + \sum_{|k,l|\leqslant r} a_{k,l} y_{i-k,j-l} + \sum_{|k,l|\leqslant r} b_{k,l} u_{i-k,j-l} + z, \text{ with } y_{i,j}(t) = f(x_{i,j})$$



- A template specifies a coupling law
- 19 template values $(9 + 9 + 1)$
- Controls the entire network

CNN computer

Template

Digital computer

Instruction

CNN template values

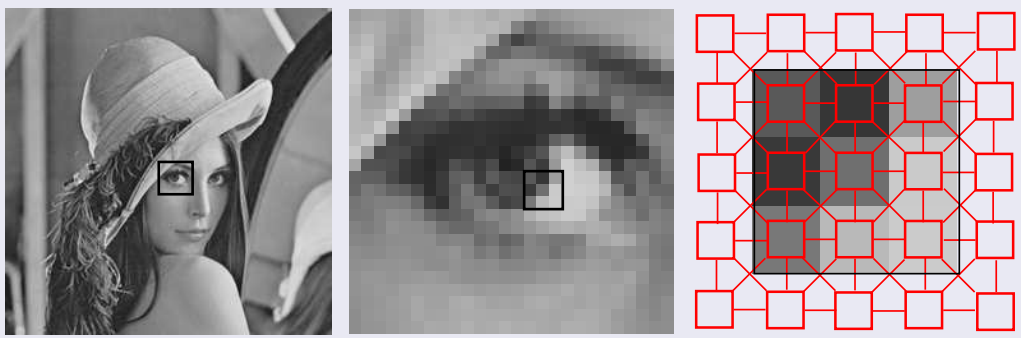
$$A = \begin{bmatrix} a_{-1,-1} & a_{-1,0} & a_{-1,1} \\ a_{0,-1} & a_{0,0} & a_{0,1} \\ a_{1,-1} & a_{1,0} & a_{1,1} \end{bmatrix}, B = \begin{bmatrix} b_{-1,-1} & b_{-1,0} & b_{-1,1} \\ b_{0,-1} & b_{0,0} & b_{0,1} \\ b_{1,-1} & b_{1,0} & b_{1,1} \end{bmatrix}, z$$

A programmable CNN is ideal for image processing

- Templates provide an ideal framework for Single-Instruction Multiple-Data computation
- Image processing operations are mostly SIMD
- Most of these tasks have a corresponding CNN template
- CNN-UM allows execution of a sequence of templates

  [Roska and Chua, IEEE TCAS, 1993]

One-to-one cell-pixel mapping



Advantages

- high parallelism
- much faster
- less silicon area
- very low power

- The **only** drawback is **robustness**

CNN chips do not behave as ideal CNNs

Example

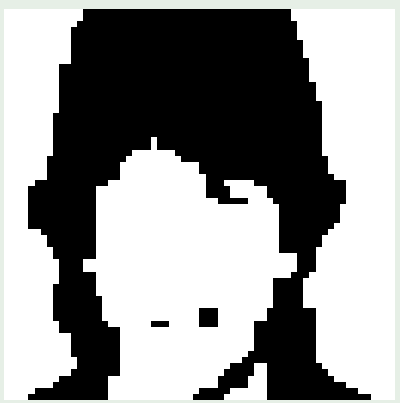
(a)



(c)



(b)



(d)



Causes of errors

- Parameter spread
- Electrical noise
- Imperfect loading from off-chip to on-chip memory
- Temperature variation

Chip-independent approaches

- Design of robust templates
- Template decomposition
- Stochastic template optimisation

Outline

1 Introduction
- The potential of Cellular Neural Networks
- How does a CNN work?
- Goals of this thesis

2 Robust CNN Tuning and Learning

3 Coupled Simulated Annealing

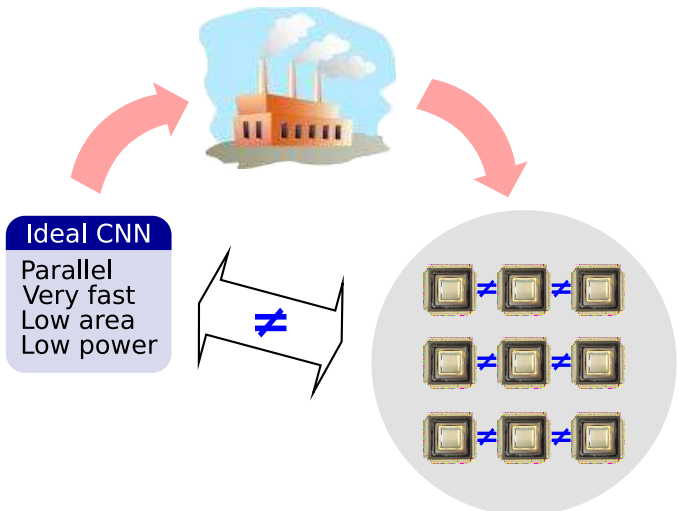
4 Applying CSA to CNN Optimisation

5 Conclusions and Challenges for Future Work

Methodologies for robustness in CNN chips

Goals

- Robust tuning of CNN chips
- Learning of complex behaviour in CNNs
- Optimisation techniques for robust CNN tuning and learning
- Integrate the methodologies
- Develop proof-of-concept applications



Outline

1 Introduction

2 Robust CNN Tuning and Learning
- Chip-specific Methods
- Learning complex dynamics
- Experiments and Results

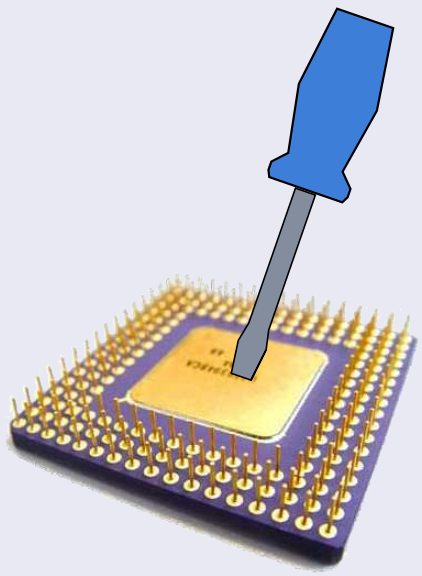
3 Coupled Simulated Annealing

4 Applying CSA to CNN Optimisation

5 Conclusions and Challenges for Future Work

Optimising with chip measurements works better

Chip-specific



[Földesy et al., IEEE TCAS-I,1999]

Optimise-decompose-optimise approach

- Uses chip measurements to calculate error
- Searches the minimum cost via gradient search
- Non-optimal templates are decomposed

Disadvantages

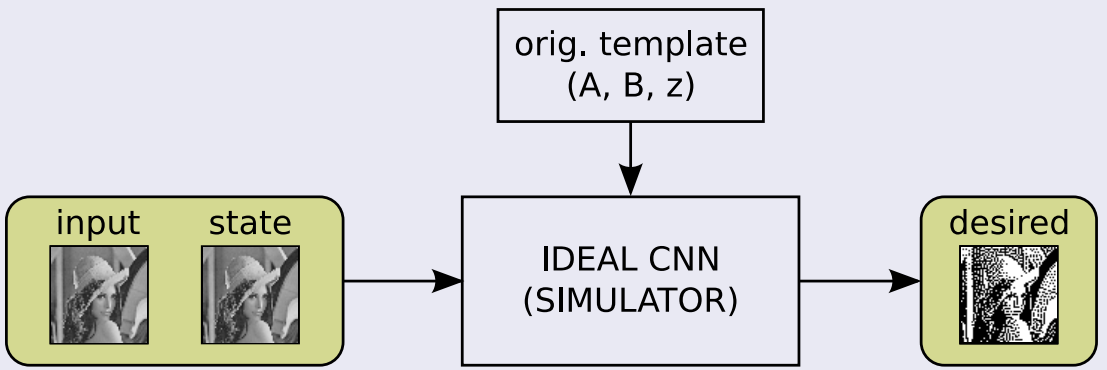
- Needs explicit calculation of error gradient
- Only optimises B and z: uncoupled templates
- May cause unnecessary template decomposition

The use of chip-measurements does improve robustness

Training set is generated using an ideal CNN simulator

[Xavier-de-Souza et al., 2003]

Generating a training set triplet



- Training set triplet: input, initial state, and desired images
- Original template $\mathbf{p}_{init}$
- Ideal CNN simulator

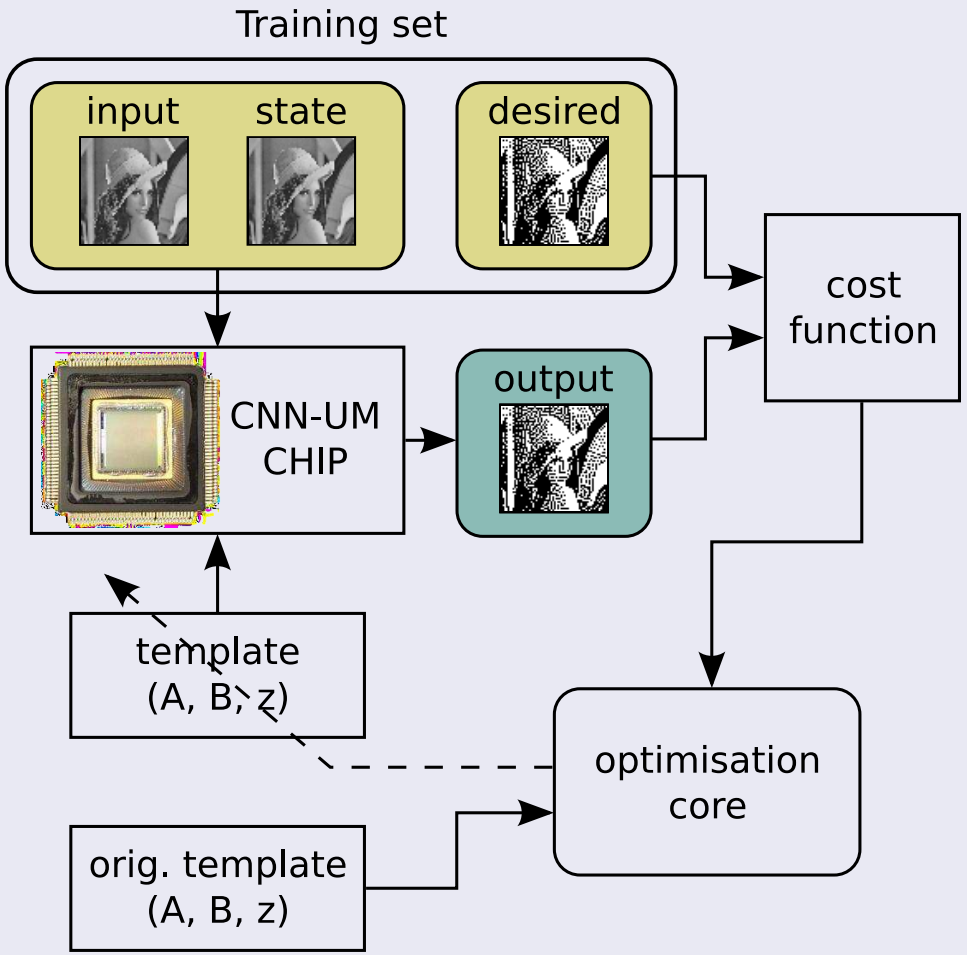
A good starting point

- Original template $\mathbf{p}_{init}$ is a good initial approximation
- Search can be limited to $[\mathbf{p}_{init} - \mathbf{b}, \mathbf{p}_{init} + \mathbf{b}]$

Presenting our tuning setup

[Xavier-de-Souza et al., IEEE TCAS-I, 2004]

Chip-specific CNN tuning setup



Optimisation core

- A global optimisation method
- Generates and accepts, or not, a new template **p**

Optimisation problem

$$\min_{\mathbf{p}} g(\mathbf{p}), \text{ where } g(\mathbf{p}) = \frac{1}{\sqrt{k}} \sqrt{\sum_{i=1}^{k} (y_i^d - y_i(\infty))^2}$$

Major advantages, a minor drawback

Advantages of this tuning setup

- No restrictions related to template type
- Propagating templates can be tuned too
- Poor local solutions are avoided due to global optimisation
- No premature template decomposition

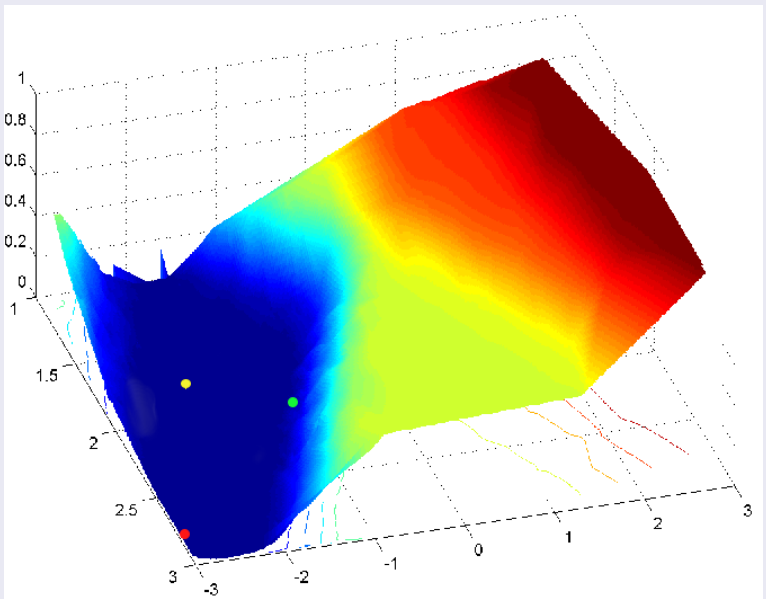
- The drawback: global optimisation is slower

What makes it worth?

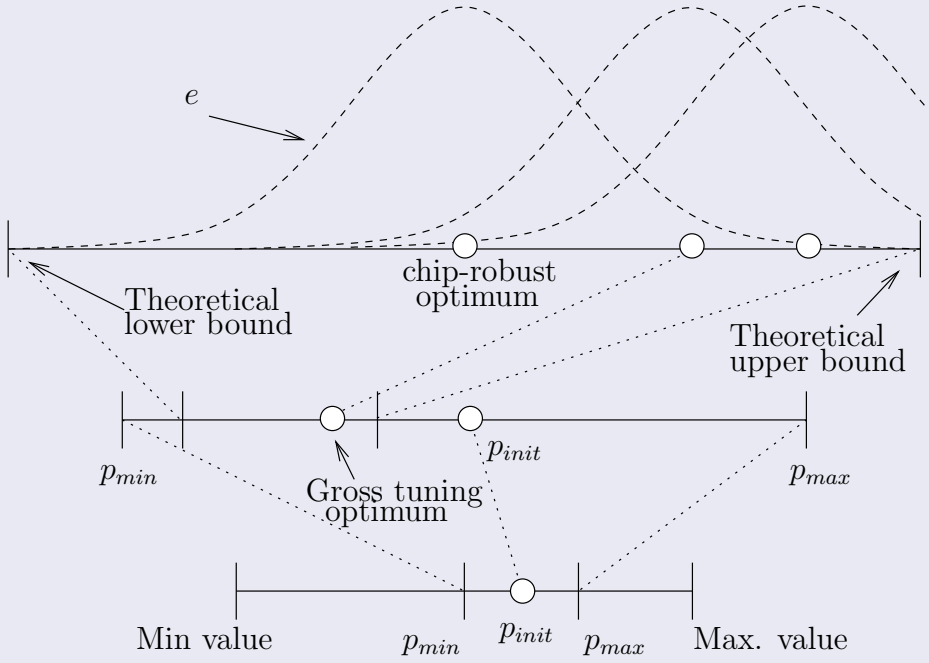
- Number of unknowns $\leqslant 19$ compensates this drawback
- Original template is a good starting point
- We only search around the original template values

Optimising for robustness

Typical template chip error



Chip-specific robustness



Optimisation problem

$$\min_{\mathbf{p}} \frac{1}{r} \sum_{j=1}^{r} g(\mathbf{p} + \mathbf{e}_j)$$

Outline

1 Introduction

2 Robust CNN Tuning and Learning
- Chip-specific Methods
- Learning complex dynamics
- Experiments and Results

3 Coupled Simulated Annealing

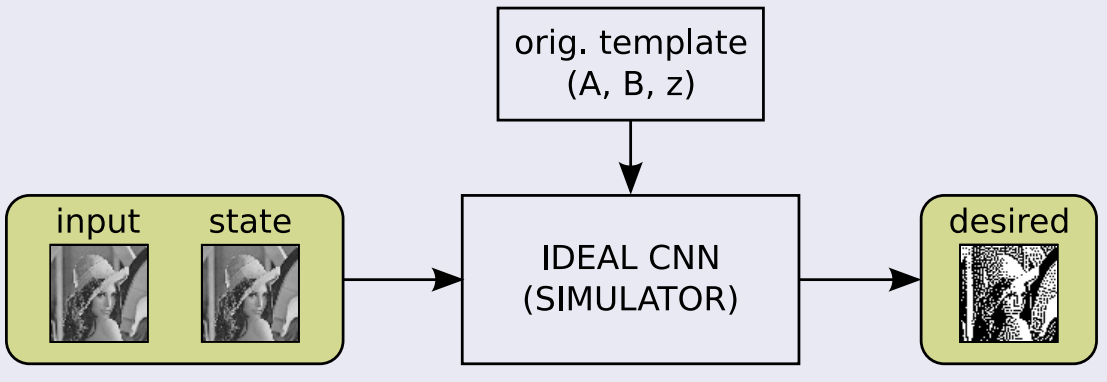
4 Applying CSA to CNN Optimisation

5 Conclusions and Challenges for Future Work

Learning is more difficult than tuning

[Xavier-de-Souza et al., 2005]

No know good starting point



- Desired output is obtained either manually or via other methodologies
- Potentially, the entire template parameter space needs to be searched

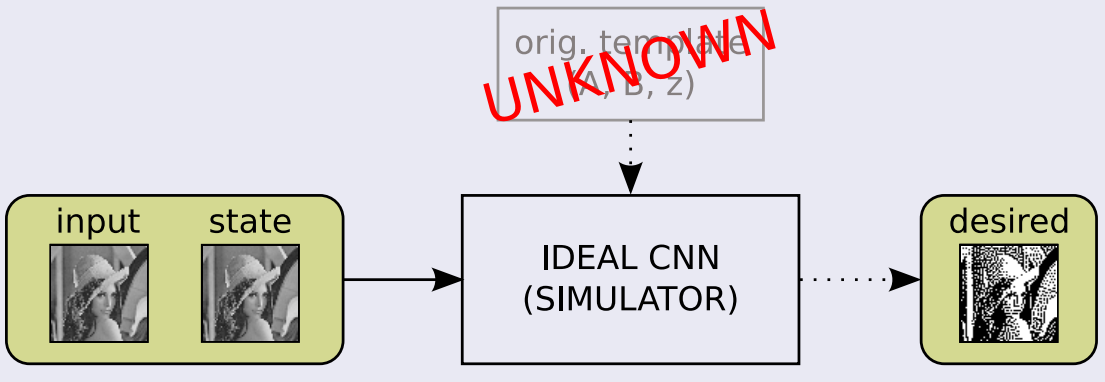
Helpful strategies

- The search space can be reduced with simple initial assumptions, e.g. symmetry, zero values
- These assumptions can be relaxed in subsequent optimisation runs

Learning is more difficult than tuning

[Xavier-de-Souza et al., 2005]

No know good starting point



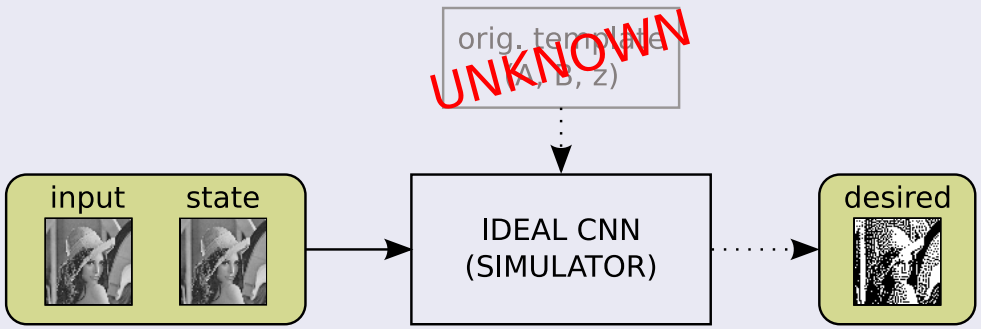
- Desired output is obtained either manually or via other methodologies
- Potentially, the entire template parameter space needs to be searched

Helpful strategies

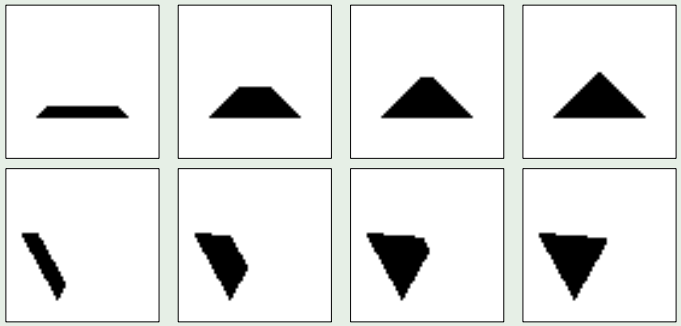
- The search space can be reduced with simple initial assumptions, e.g. symmetry, zero values
- These assumptions can be relaxed in subsequent optimisation runs

Learning dynamics requires more desired outputs

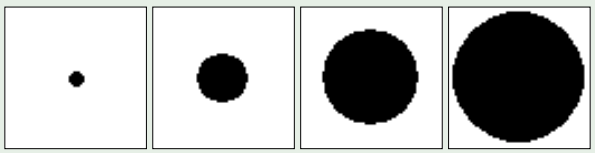
Desired output is a sequence of images



Example



Example



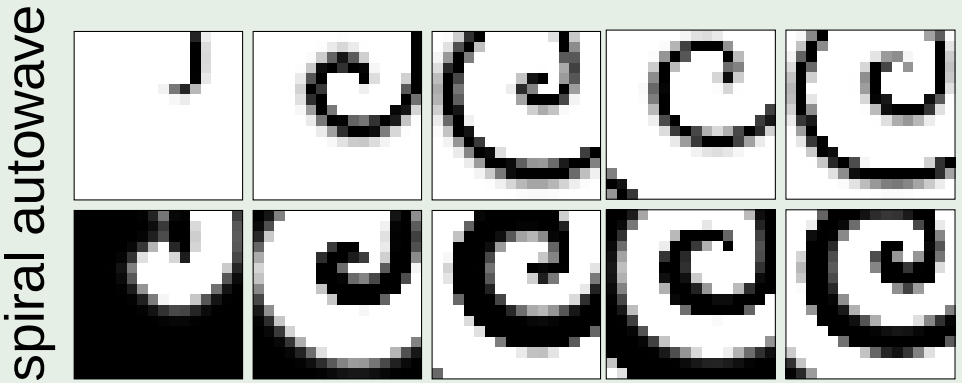
Example



[Xavier-de-Souza et al., JCTA, 2006]

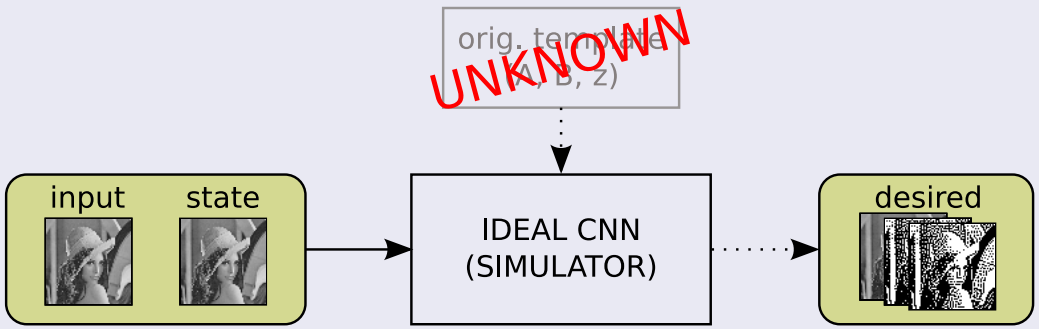
- New Active Wave Computing operations

Example

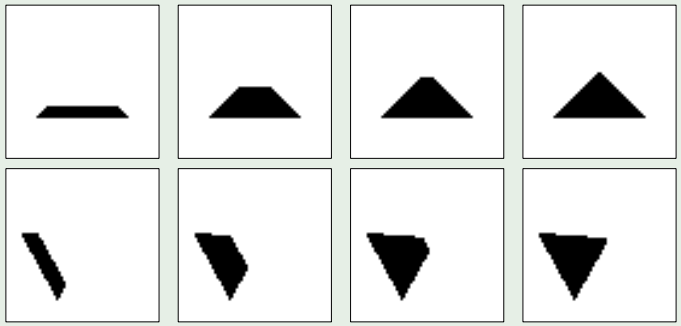


Learning dynamics requires more desired outputs

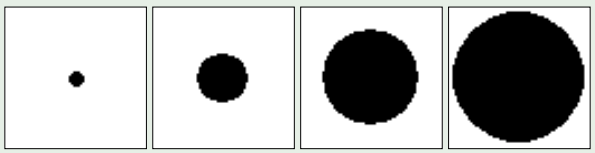
Desired output is a sequence of images



Example



Example



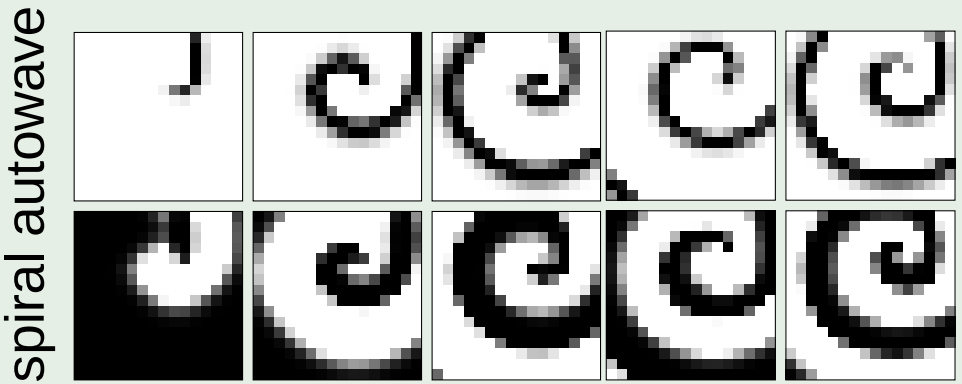
Example



[Xavier-de-Souza et al., JCTA, 2006]

- New Active Wave Computing operations

Example



Outline

1 Introduction

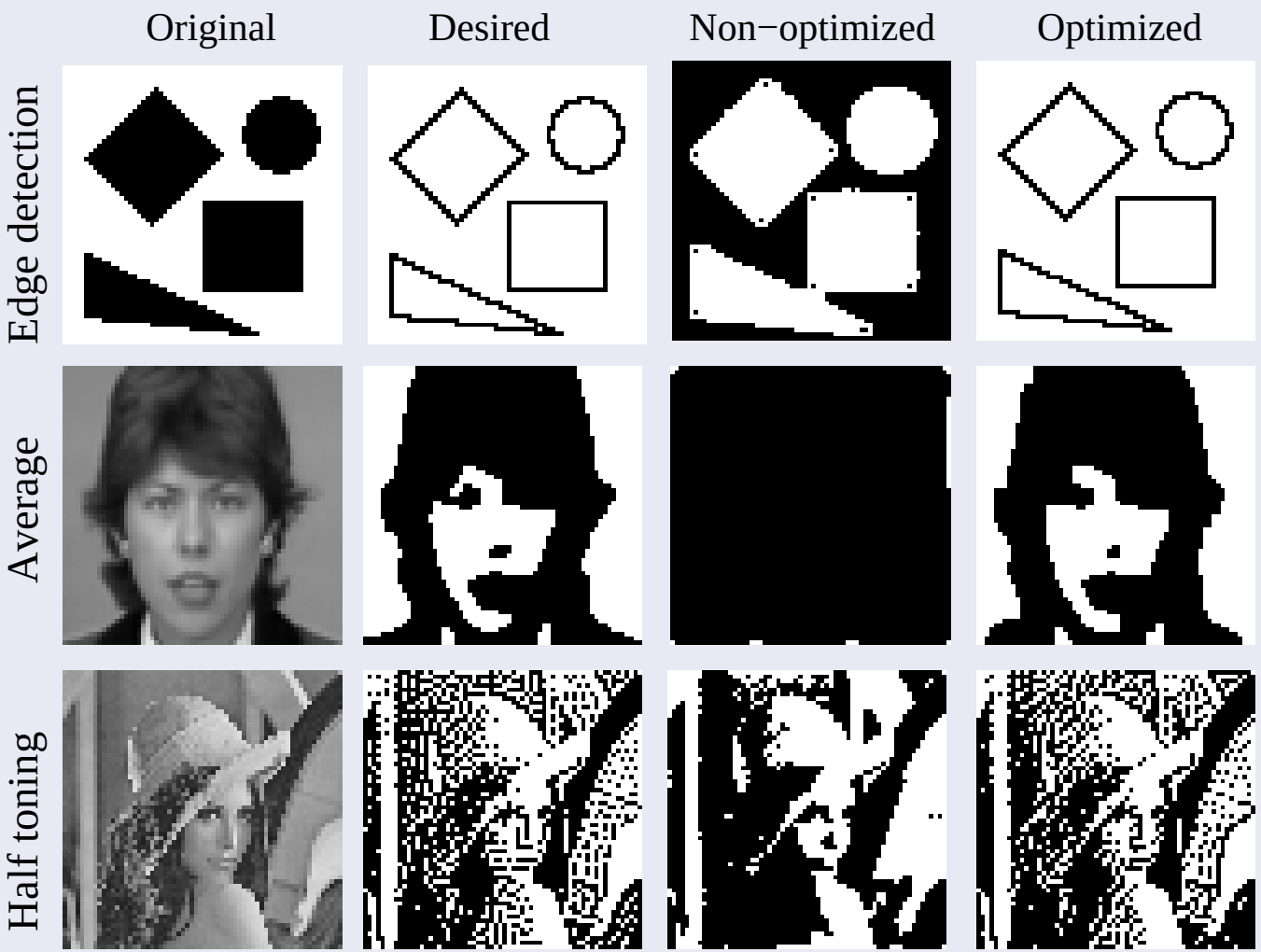
2 Robust CNN Tuning and Learning
- Chip-specific Methods
- Learning complex dynamics
- Experiments and Results

3 Coupled Simulated Annealing

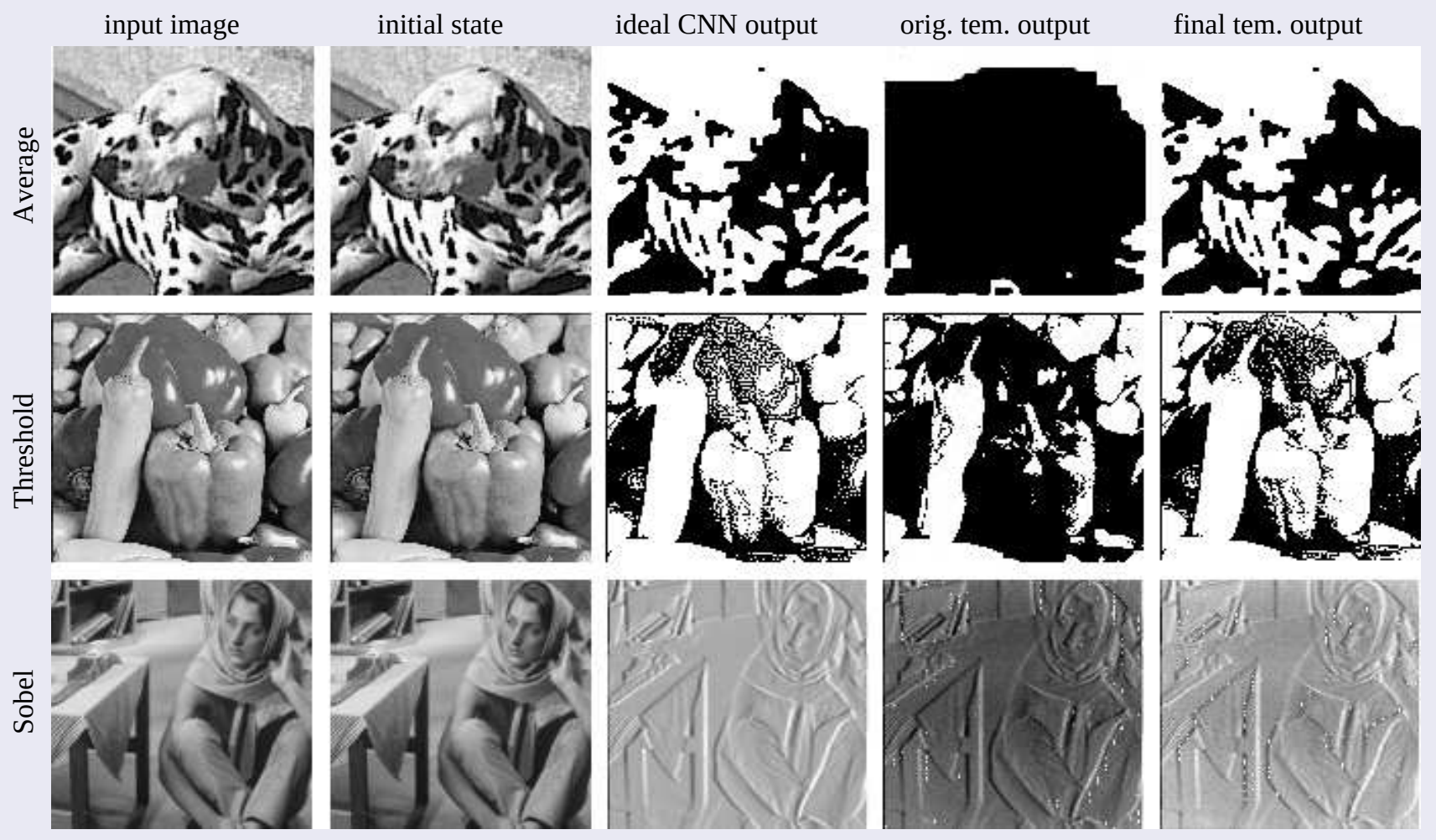
4 Applying CSA to CNN Optimisation

5 Conclusions and Challenges for Future Work

Tuning results for ACE4k chip (64 $\times$ 64 cells)

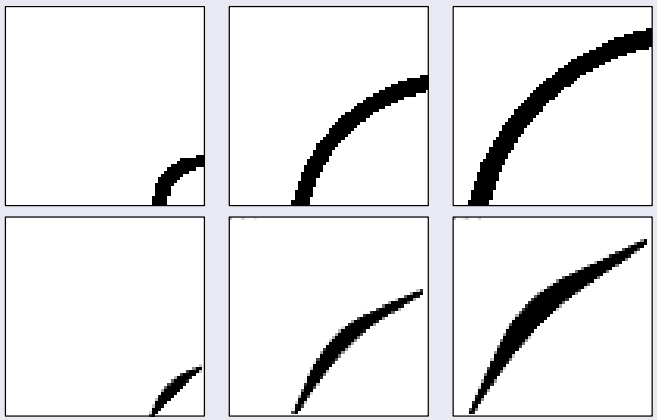


Tuning results for ACE16k chip (128 $\times$ 128 cells)

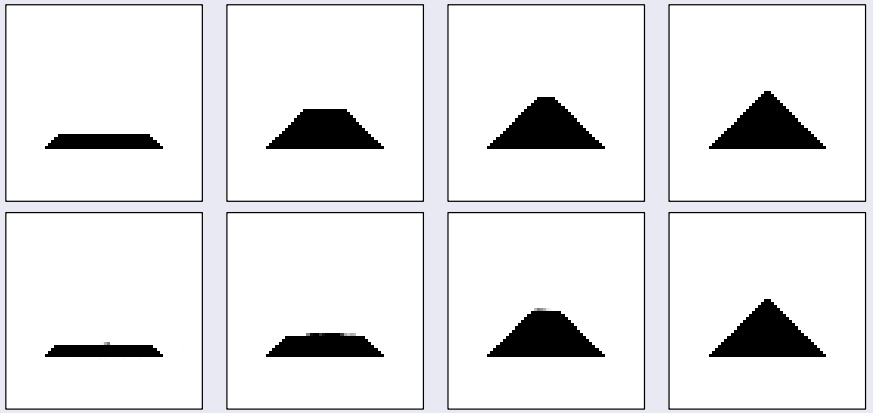


On-chip learning results for ACE4k (64×64 cells)

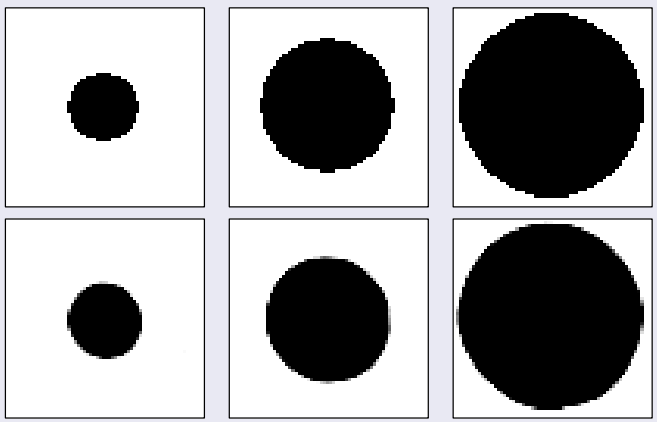
Travelling wave



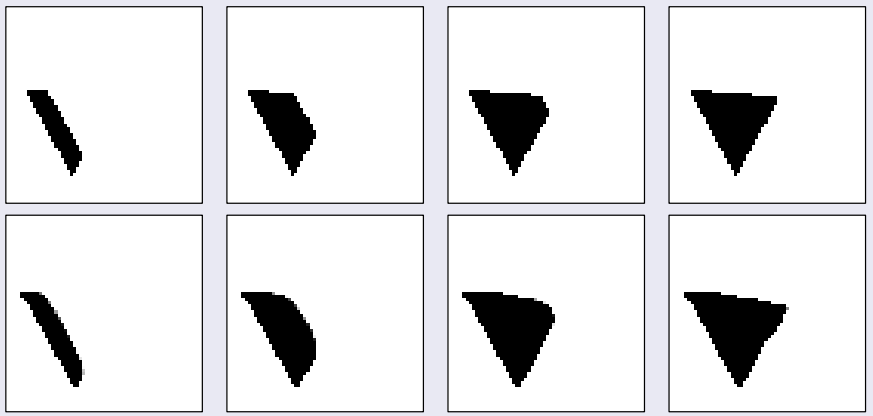
Pyramiding up



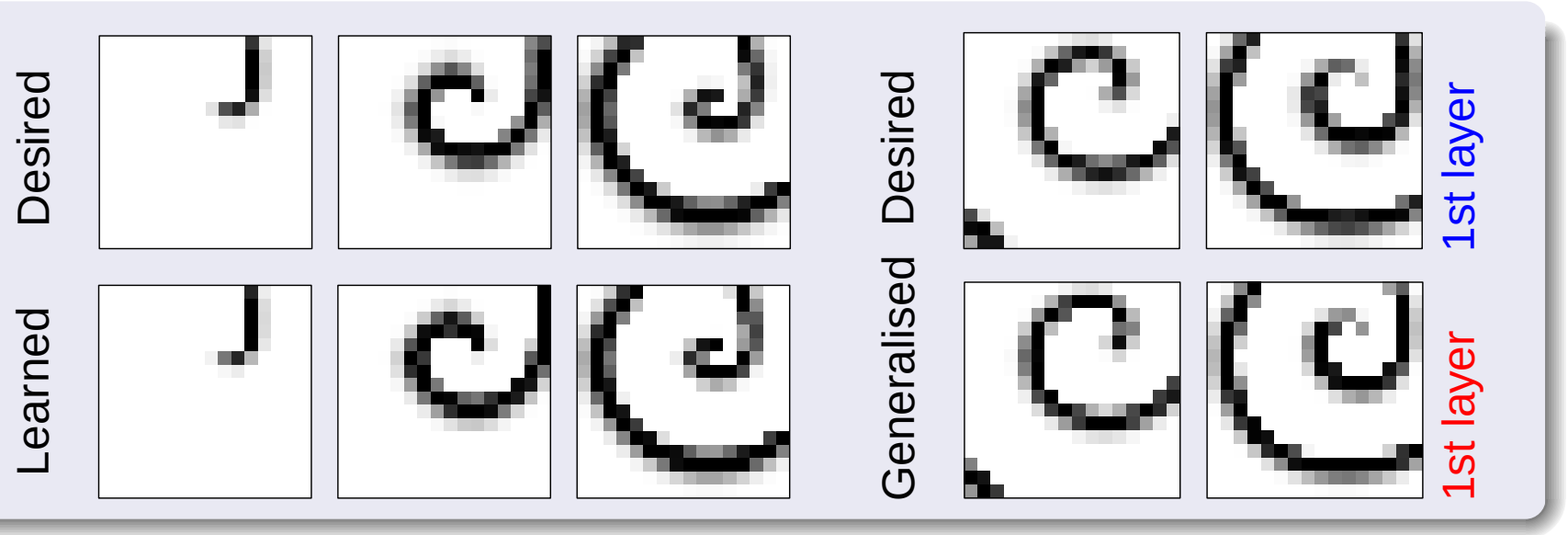
Combustion wave

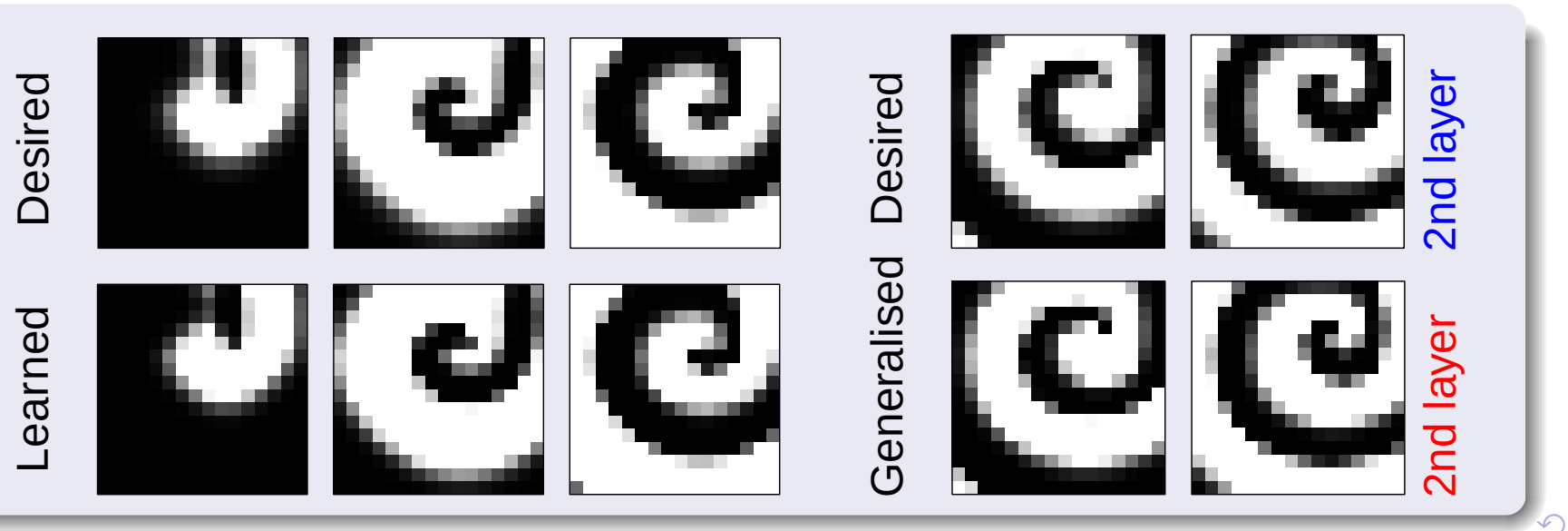


Pyramiding in angle



Simulation learning of a spiral autowave (16 × 16 cells)





Outline

1 Introduction

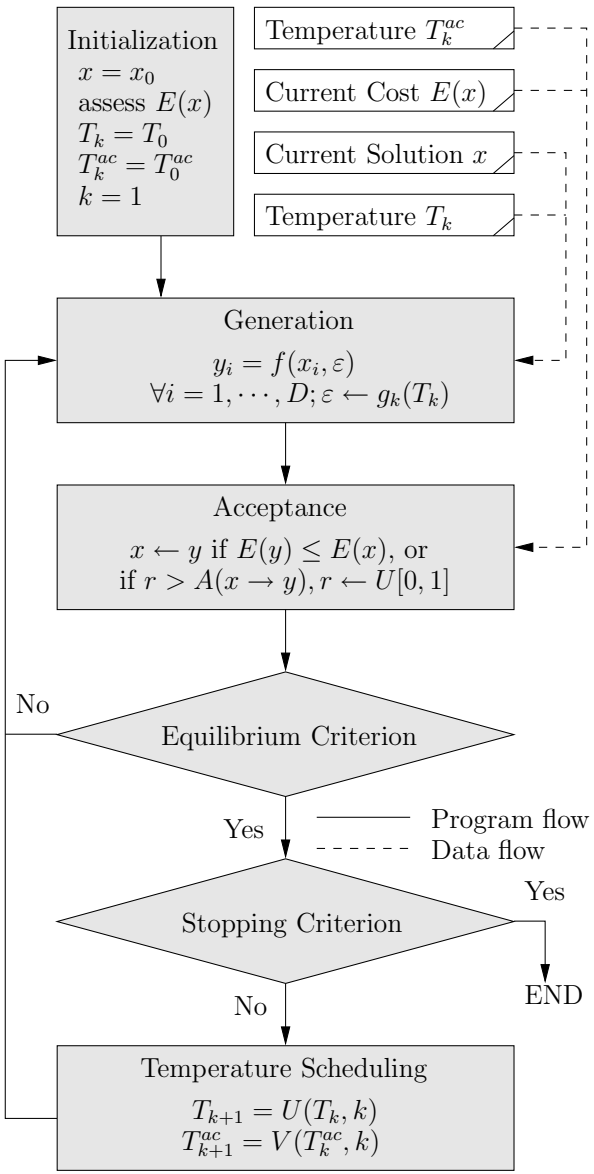
2 Robust CNN Tuning and Learning

3 Coupled Simulated Annealing
- Simulated annealing and cooperative behaviour
- General principles of Coupled SA
- Variance control of acceptances
- Experiments and results

4 Applying CSA to CNN Optimisation

5 Conclusions and Challenges for Future Work

SA is based on two stochastic processes



Simulating physical annealing

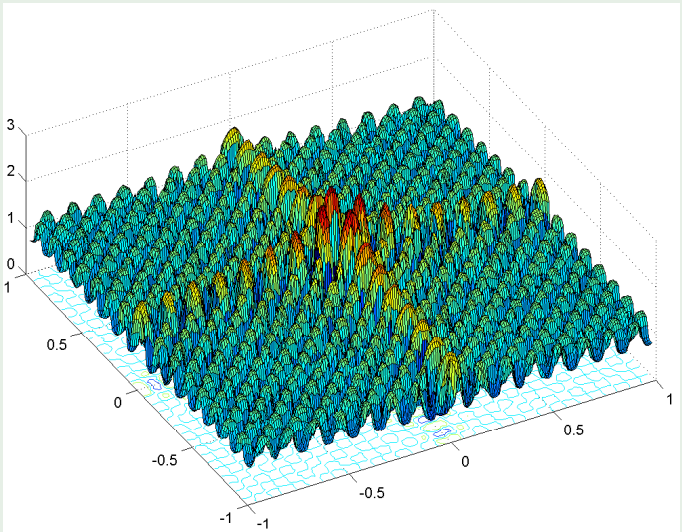
- Physical process:
  heat, hold and cool slowly down
- Mathematical modelling:

Material states	$\longrightarrow$	Problem solutions
Energy of a state	$\longrightarrow$	Cost of a solution
Temperature	$\longrightarrow$	Control parameter

- Convergence proof exists for $t \rightarrow \infty$
- Many speed up versions exist:
  quality of solution $\leftrightarrow$ speed
- With distributed versions:
  quality $\leftrightarrow$ speed $\leftrightarrow$ resources

Why a new Simulated Annealing (SA) method?

Example



- Adaptive SA: less suitable for CNN tuning
- Too many run-modify-rerun loops
- Problem representing template solutions in Genetic Algorithms

The coupling idea

- Coupled Local Minimisers couples local gradient searching processes
- What would be the effect of coupling in SA processes?

Cooperative behaviour in SA

Parallel SA versions

- Concurrent processes move independently
- Best solution is exchanged among process

Ensemble SA

- No exchange of best solutions
- Entire ensemble moves according to average cost

[Xavier-de-Souza et al., 2006a]

The new Coupled Simulated Annealing class of methods

- Coupled acceptance probabilities for SA
- Efficient parallel/distributed implementation
- Less sensitive to initialisation parameters

Outline

1 Introduction

2 Robust CNN Tuning and Learning

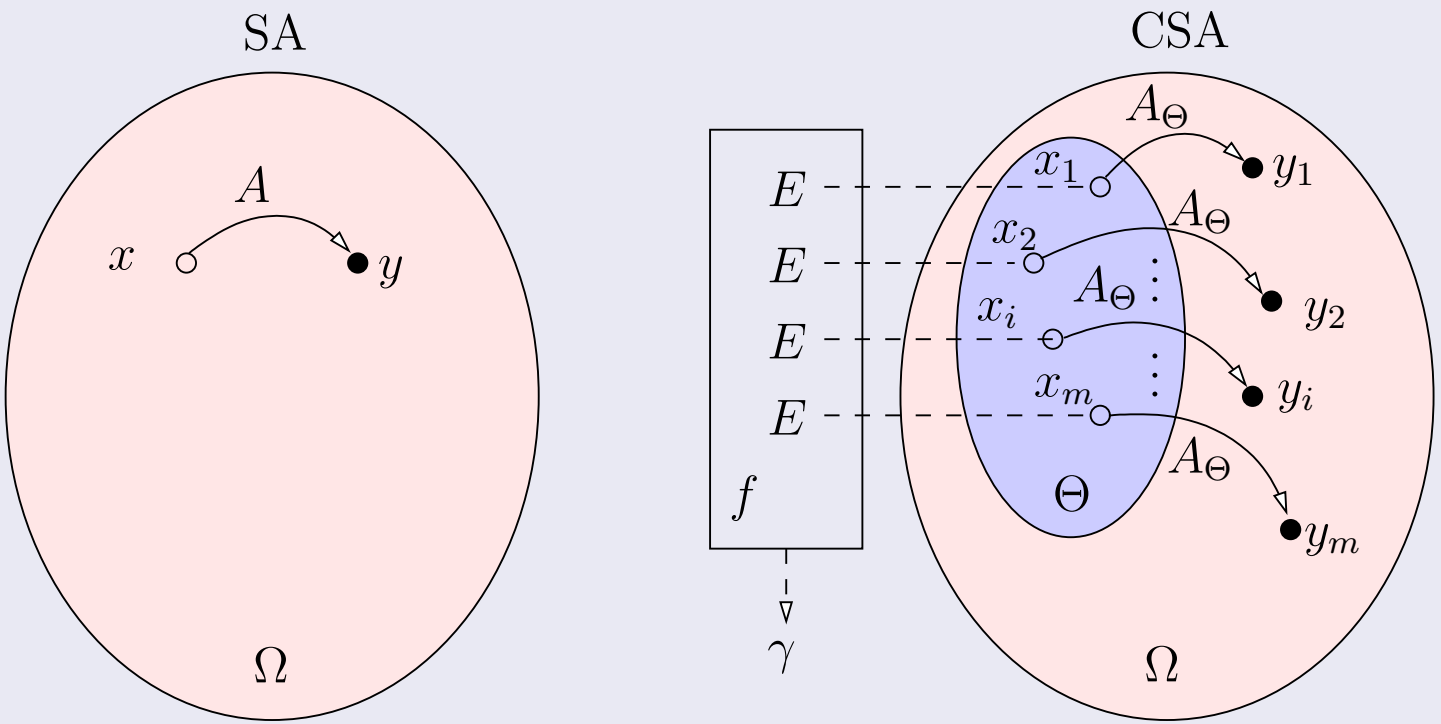
3 Coupled Simulated Annealing
- Simulated annealing and cooperative behaviour
- General principles of Coupled SA
- Variance control of acceptances
- Experiments and results

4 Applying CSA to CNN Optimisation

5 Conclusions and Challenges for Future Work

Comparing SA and Coupled SA

Properties of the CSA class of methods



$0 \leq A(x \to y) \leq 1, \quad \forall\ x, y \in \Omega.$

$0 \leq A_\Theta(\gamma, x_i \to y_i) \leq 1,\ \forall\ x \in \Theta, y \in \Omega,$ with $\Theta \subset \Omega,$ and $\gamma = f\left[E(x_1), E(x_2), \cdots, E(x_m)\right].$

Three CSA instance methods

Method 1

Multi-state SA

$$A_\Theta(\gamma, x_i \to y_i) = \frac{\exp\left(\frac{-E(y_i)}{T_k^{ac}}\right)}{\exp\left(\frac{-E(y_i)}{T_k^{ac}}\right) + \exp\left(\frac{-E(x_i)}{T_k^{ac}}\right)}$$

- Generalises classical SA acceptance for $\Theta \equiv x_1$

Three CSA instance methods

Method 1

Multi-state SA

$$A_\Theta(\gamma, x_i \to y_i) = \frac{\exp\left(\frac{-E(y_i)}{T_k^{ac}}\right)}{\exp\left(\frac{-E(y_i)}{T_k^{ac}}\right) + \sum_{x_j \in \Theta} \exp\left(\frac{-E(x_j)}{T_k^{ac}}\right)}$$

- Generalises classical SA acceptance for $\Theta \equiv x_1$

Three CSA instance methods

Method 1

Multi-state SA

$$A_\Theta(\gamma, x_i \to y_i) = \frac{\exp\left(\frac{-E(y_i)}{T_k^{ac}}\right)}{\exp\left(\frac{-E(y_i)}{T_k^{ac}}\right) + \underbrace{\sum_{x_j \in \Theta} \exp\left(\frac{-E(x_j)}{T_k^{ac}}\right)}_{\gamma}}$$

- Generalises classical SA acceptance for $\Theta \equiv x_1$

Three CSA instance methods

Method 2

Blind acceptance

$$A_\Theta(\gamma, x_i \to y_i) = 1 - \frac{\exp\left(\frac{-E(x_i)}{T_k^{ac}}\right)}{\sum_{x_j \in \Theta} \exp\left(\frac{-E(x_j)}{T_k^{ac}}\right)}$$

- Acceptance of y_i is independent of its energy when $E(x_i) < E(y_i)$

Three CSA instance methods

Method 2

Blind acceptance

$$A_\Theta(\gamma, x_i \to y_i) = 1 - \frac{\exp\left(\frac{-E(x_i)}{T_k^{ac}}\right)}{\underbrace{\sum_{x_j \in \Theta} \exp\left(\frac{-E(x_j)}{T_k^{ac}}\right)}_{\gamma}}$$

- Acceptance of y_i is independent of its energy when $E(x_i) < E(y_i)$

Three CSA instance methods

Method 3

Coupled SA-Modified

$$A_{\Theta}(\gamma, x_i \rightarrow y_i) = \frac{\exp\left(\frac{E(x_i)}{T_k^{ac}}\right)}{\sum_{x_j \in \Theta} \exp\left(\frac{E(x_j)}{T_k^{ac}}\right)}$$

- Also perform blind acceptance
- Strategy: probability of any current state leave to a new state sums to 1

Three CSA instance methods

Method 3

Coupled SA-Modified

$$A_{\Theta}(\gamma, x_i \rightarrow y_i) = \frac{\exp\left(\frac{E(x_i)}{T_k^{ac}}\right)}{\underbrace{\sum_{x_j \in \Theta} \exp\left(\frac{E(x_j)}{T_k^{ac}}\right)}_{\gamma}}$$

- Also perform blind acceptance
- Strategy: probability of any current state leave to a new state sums to 1

Three CSA instance methods

Method 3

Coupled SA-Modified

$$A_\Theta(\gamma, x_i \to y_i) = \frac{\exp\left(\frac{E(x_i) - \max_{z\in\Theta}(E(z))}{T_k^{ac}}\right)}{\underbrace{\sum_{x_j\in\Theta} \exp\left(\frac{E(x_j) - \max_{z\in\Theta}(E(z))}{T_k^{ac}}\right)}_{\gamma}}$$

- Also perform blind acceptance
- Strategy: probability of any current state leave to a new state sums to 1

Outline

1 Introduction

2 Robust CNN Tuning and Learning

3 Coupled Simulated Annealing
- Simulated annealing and cooperative behaviour
- General principles of Coupled SA
- Variance control of acceptances
- Experiments and results

4 Applying CSA to CNN Optimisation

5 Conclusions and Challenges for Future Work

Adapting the schedule of the acceptance temperature

Acceptance variance in CSA-M

$$0 \leqslant \sigma^2(A_\Theta) \leqslant \frac{m-1}{m^2}$$

Simple way to control the variance

if $\sigma^2(A_\Theta) < \sigma_D^2$,
$$T_k^{ac} = T_{k-1}^{ac}(1-\alpha)$$
if $\sigma^2(A_\Theta) > \sigma_D^2$,
$$T_k^{ac} = T_{k-1}^{ac}(1+\alpha)$$

- Reduces sensitivity to initial parameters
- Adaptive schedule for T_k^{ac}
- Experimentally:

Best values

$$\begin{aligned} \sigma_D^2 &\approx 99\% \text{ of } \left(\tfrac{m-1}{m^2}\right) \\ \alpha &\approx 0.05 \end{aligned}$$

Outline

1 Introduction

2 Robust CNN Tuning and Learning

3 Coupled Simulated Annealing
- Simulated annealing and cooperative behaviour
- General principles of Coupled SA
- Variance control of acceptances
- Experiments and results

4 Applying CSA to CNN Optimisation

5 Conclusions and Challenges for Future Work

Presenting the test problems

Test problems group 1

- Unimodal and simple multi-modal problems

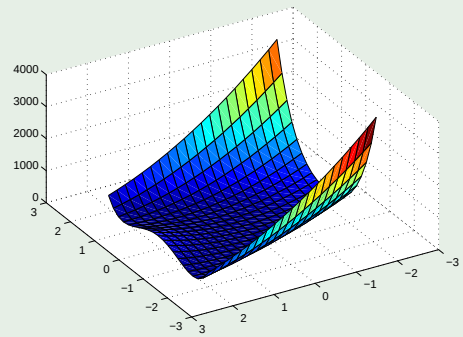
Test problems group 2

- 6 multidimensional and multi-modal problems
- Many local minima

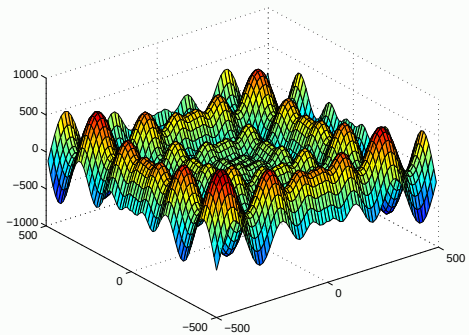
Test problems group 3

- 6 rotated problems from group 2
- Difficult for multiple 1-D searches

Example



Example



Presenting the test problems

Test problems group 1

- Unimodal and simple multi-modal problems

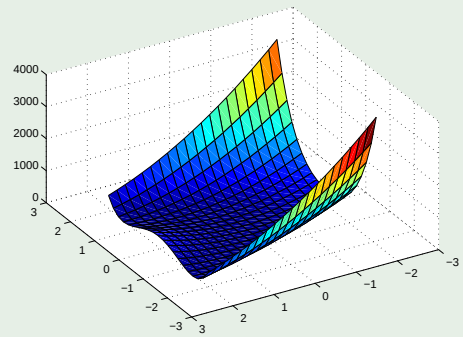
Test problems group 2

- 6 multidimensional and multi-modal problems
- Many local minima

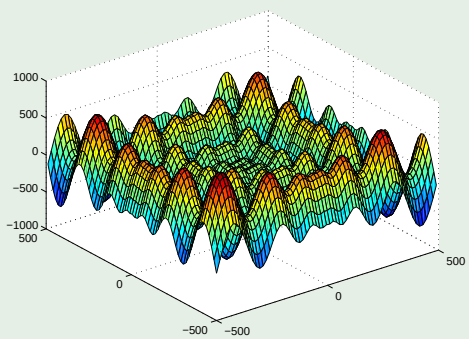
Test problems group 3

- 6 rotated problems from group 2
- Difficult for multiple 1-D searches

Example



Example



Presenting the test problems

Test problems group 1

- Unimodal and simple multi-modal problems

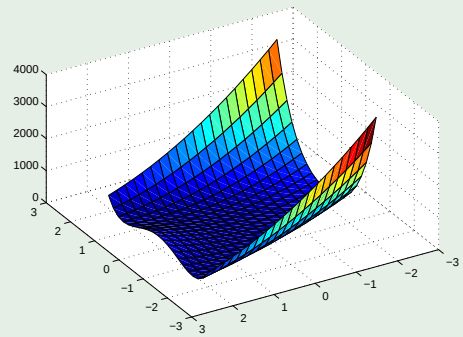
Test problems group 2

- 6 multidimensional and multi-modal problems
- Many local minima

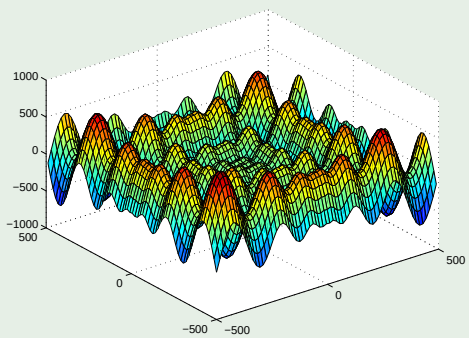
Test problems group 3

- 6 rotated problems from group 2
- Difficult for multiple 1-D searches

Example

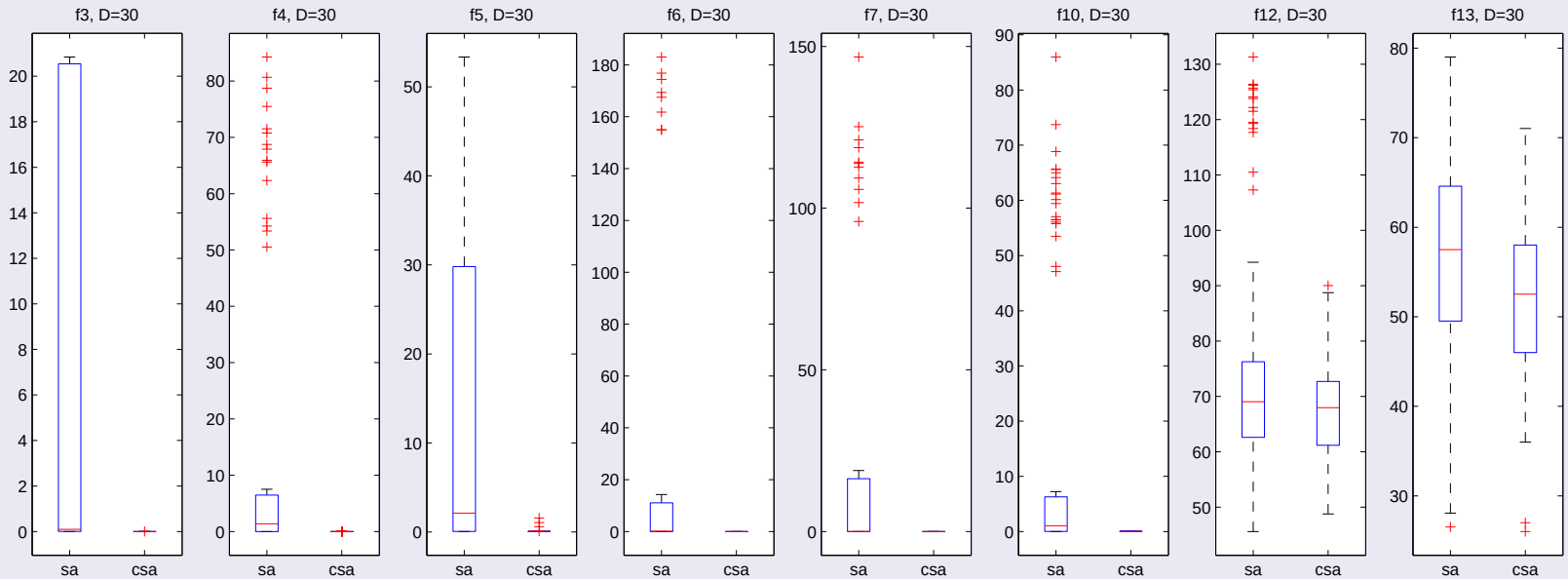


Example



Comparison experiments

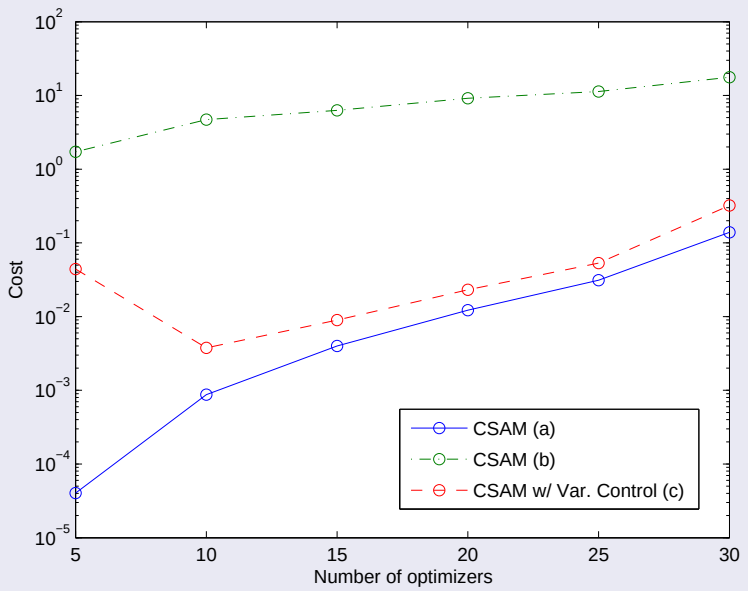
Multi-start SA versus CSA-M with variance control, D=30



- Comparable performance without variance control
- When testing sensitivity against initialisation parameters Clear CSA-M superiority with variance control

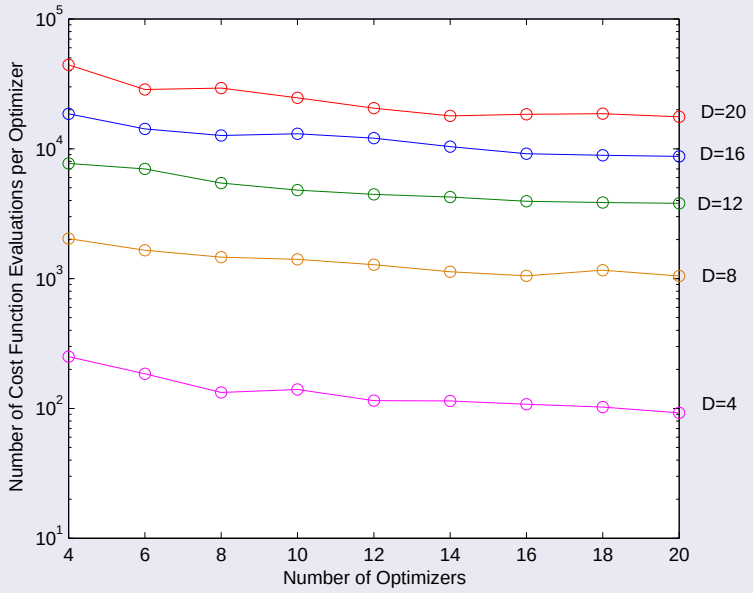
Other experiments

Effect of variance control



- Variance control approximates best average run

Scaling with dimension



- Final cost improves consistently with m
- Scales with higher dimensions

Outline

1 Introduction

2 Robust CNN Tuning and Learning

3 Coupled Simulated Annealing

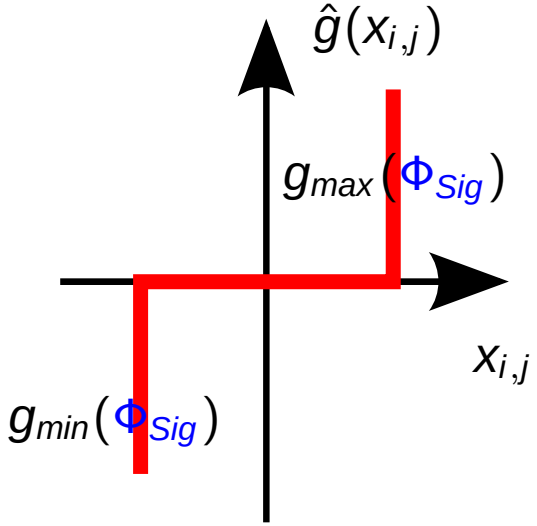
4 Applying CSA to CNN Optimisation
- General approach to CNN optimisation
- A systematic approach
- Applications

5 Conclusions and Challenges for Future Work

CSA for tuning and learning

Cell model

$$\frac{dx_{i,j}}{dt} = -\hat{g}(x_{i,j}, \Phi_{Sig}) + z(\Phi_{Tem}) + \sum_{|k,l|\leqslant r} a_{k,l}(\Phi_{Tem}) x_{i-k,j-l} + \sum_{|k,l|\leqslant r} b_{k,j}(\Phi_{Tem}) u_{i-k,j-l}(\Phi_{Opt}, \Phi_{Sig})$$



Hardware parameters

$$\Phi = [\Phi_{Sig}, \Phi_{Tem}, \Phi_{Opt}]$$

The optimisation problem within the unified framework

$$\min_{A,B,z,\Phi,\Delta t_1,\cdots,\Delta t_T} \frac{1}{N_S N_R N_T} \sum_{s=1}^{N_S} \sum_{n=1}^{N_R} \sum_{k=1}^{N_T} \sum_{i,j} (x^d_{i,j;s,k,n} - x_{i,j}(A, B, z, \Phi, t_k))^2$$

Outline

1 Introduction

2 Robust CNN Tuning and Learning

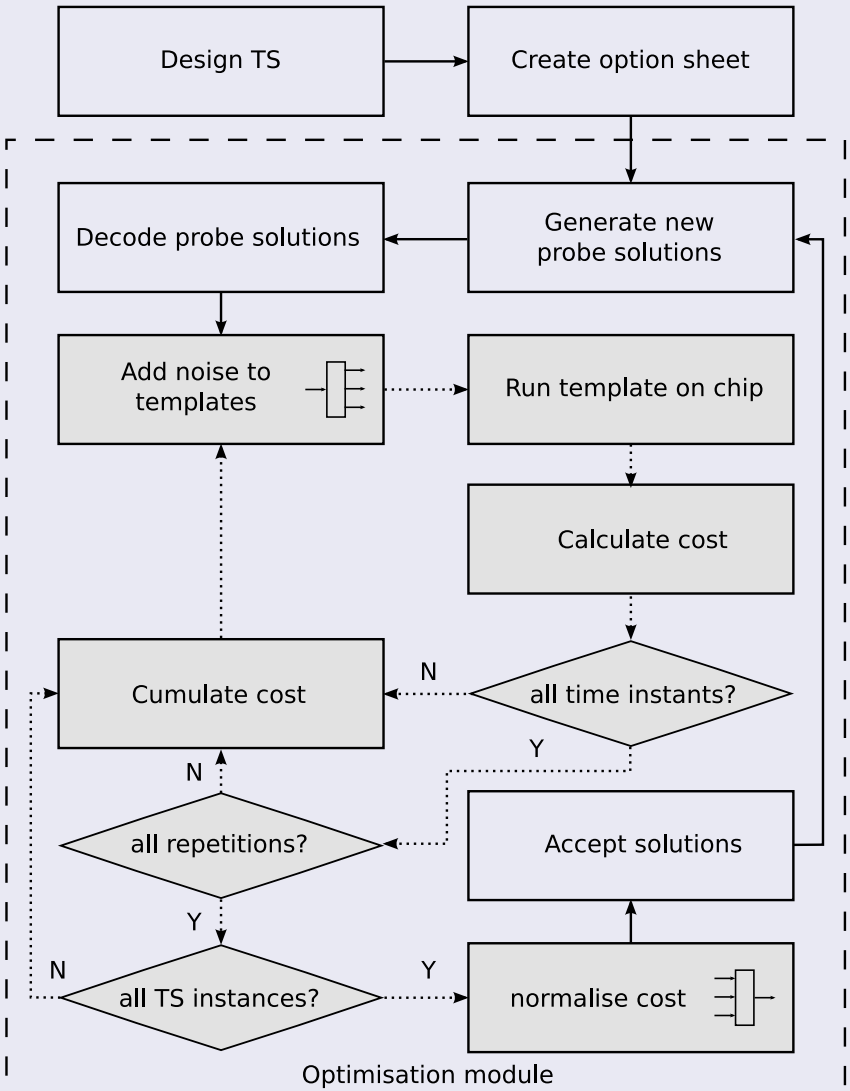
3 Coupled Simulated Annealing

4 Applying CSA to CNN Optimisation
- General approach to CNN optimisation
- A systematic approach
- Applications

5 Conclusions and Challenges for Future Work

Unified CNN optimisation framework

Block diagram



[Hillier et al., 2006]

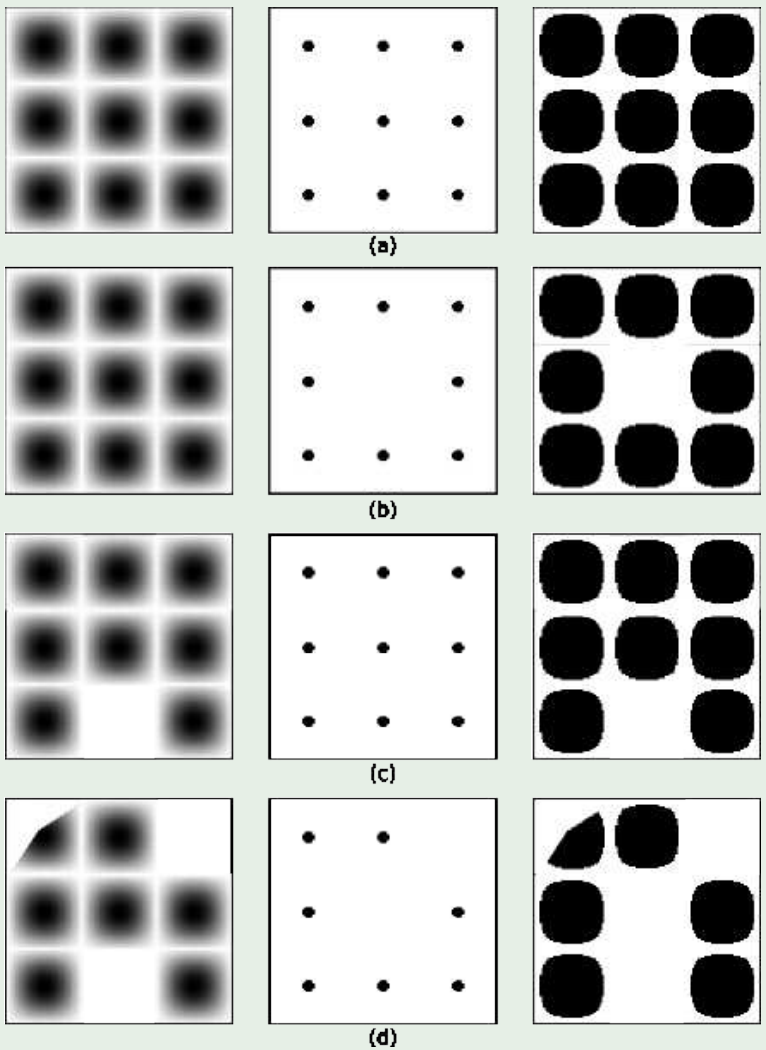
CNN optimisation cases

- Chip-specific template tuning
- Learning of fixed-point templates
- Learning of spatio-temporal dynamics

- Modular, replicable to many CSA processes
- CSA reduces run-modify-rerun loops

Guidelines for defining a training set

Example



- Include all essential input-output mapping functionalities
- Consider also bias and fixed state images
- Balance dark and light pixels
- Embed many functionalities in a single TS instance
- Consider exchanging input and initial state
- Ensure continuity of dynamics

CNNOPT: Matlab toolbox

http://www.esat.kuleuven.be/sista/chaoslab/cnnopt

Outline

1 Introduction

2 Robust CNN Tuning and Learning

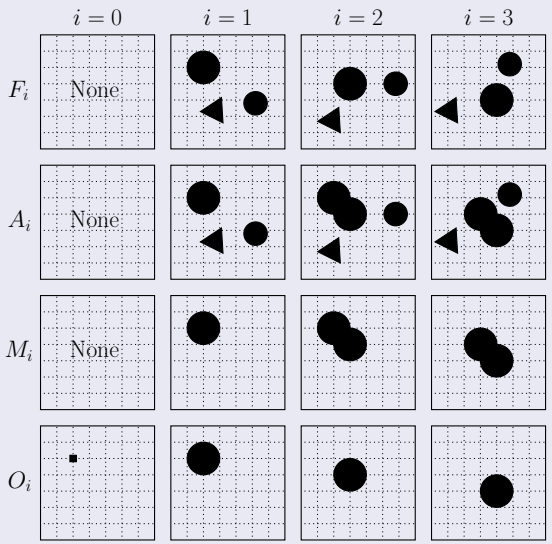
3 Coupled Simulated Annealing

4 Applying CSA to CNN Optimisation
- General approach to CNN optimisation
- A systematic approach
- Applications

5 Conclusions and Challenges for Future Work

Real-time object tracking

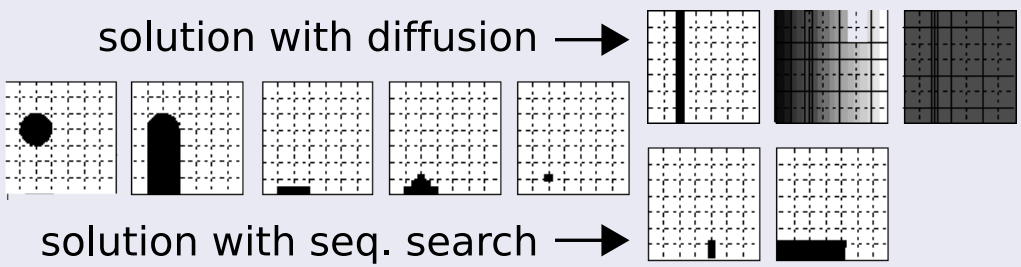
Tracking the object



[Xavier-de-Souza et al., 2004]

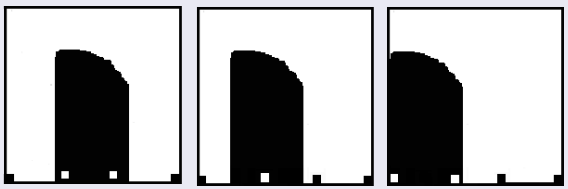
- Building block for other applications e.g. computer/game interfaces, eye tracking, face tracking, gaze estimation, target tracking, multiple object tracking.
- More than 300 frames/sec. in 2nd generation CNN chips

Obtaining the coordinates



Handsfree wheelchair driving

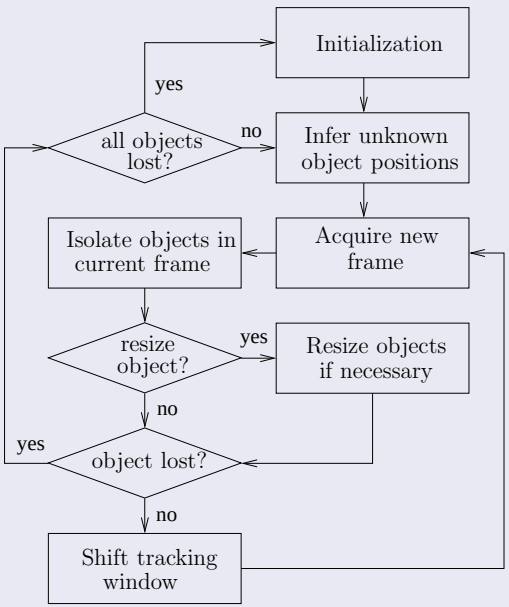
Shifting chip window



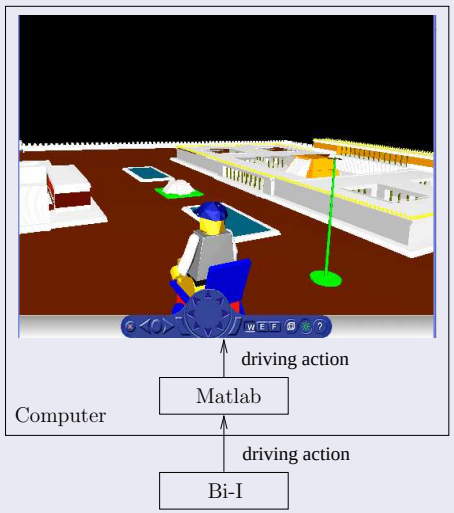
Adaptive resizing



Face tracking



3-D Testing environment



Video 1 Video 2

[Xavier-de-Souza et al., 2006b]

Outline

1 Introduction

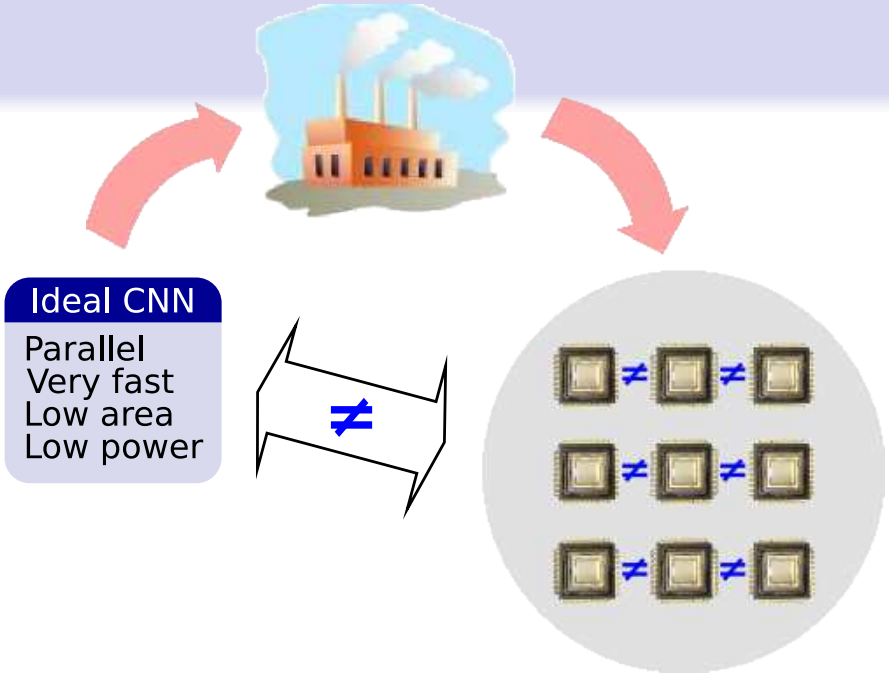
2 Robust CNN Tuning and Learning

3 Coupled Simulated Annealing

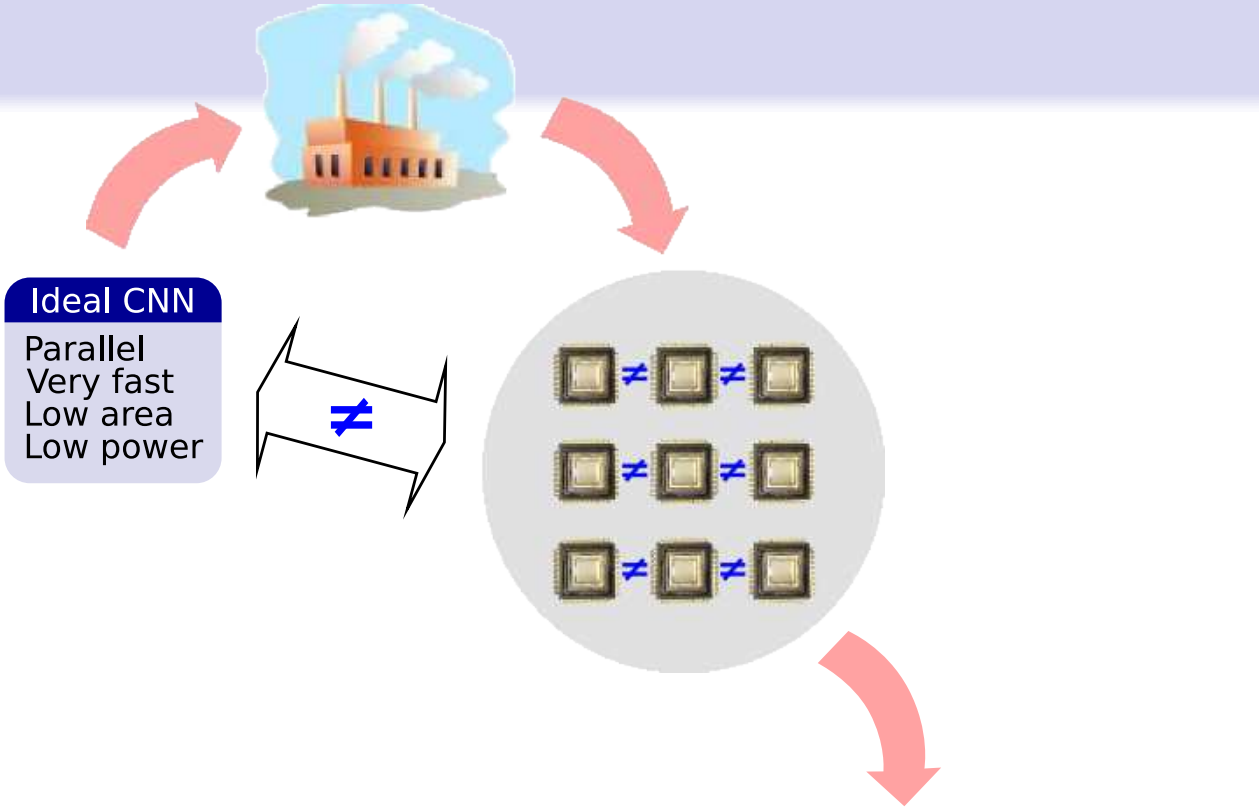
4 Applying CSA to CNN Optimisation

5 Conclusions and Challenges for Future Work
- Conclusions
- Challenges for Future Work

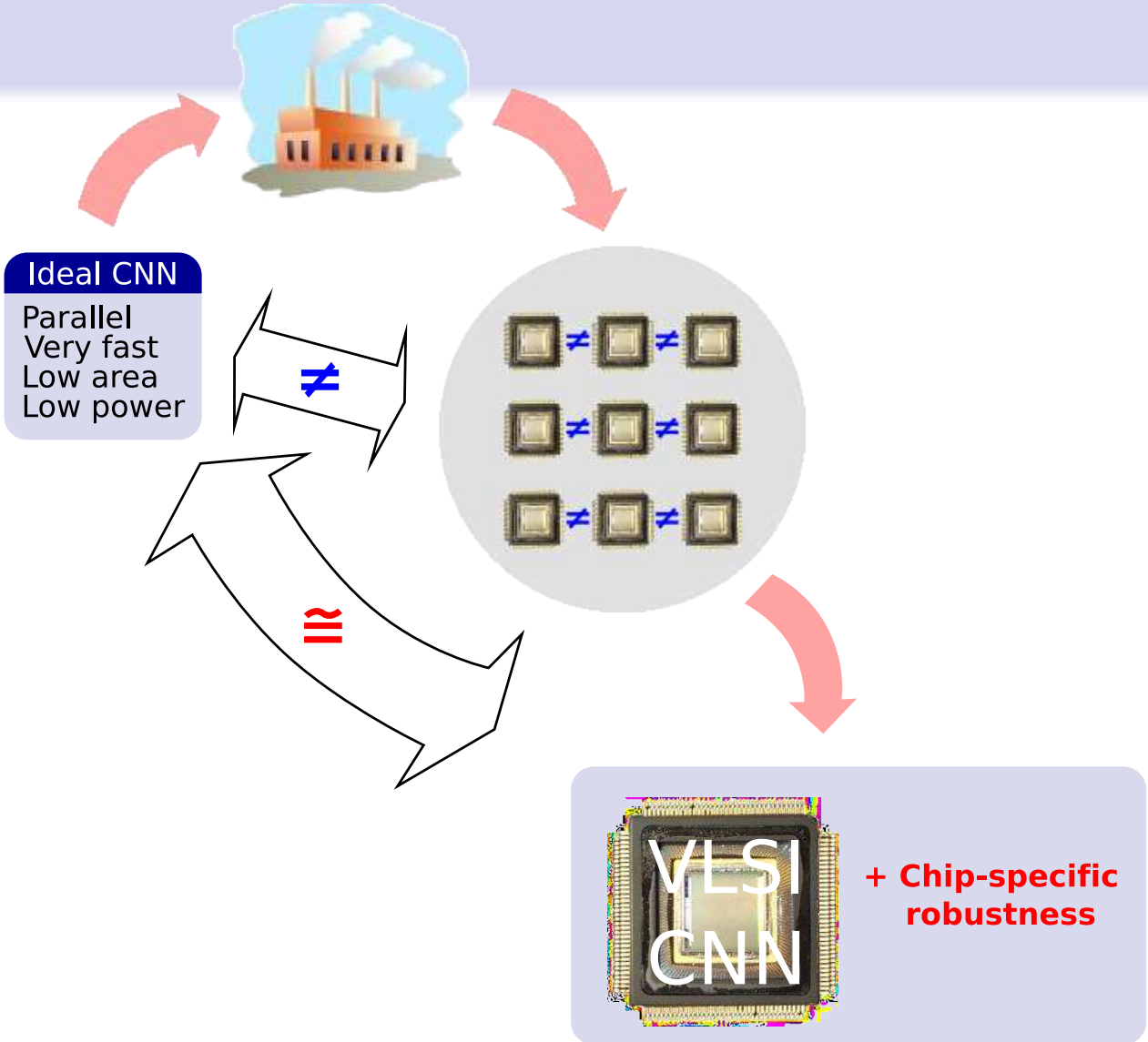
Conclusions



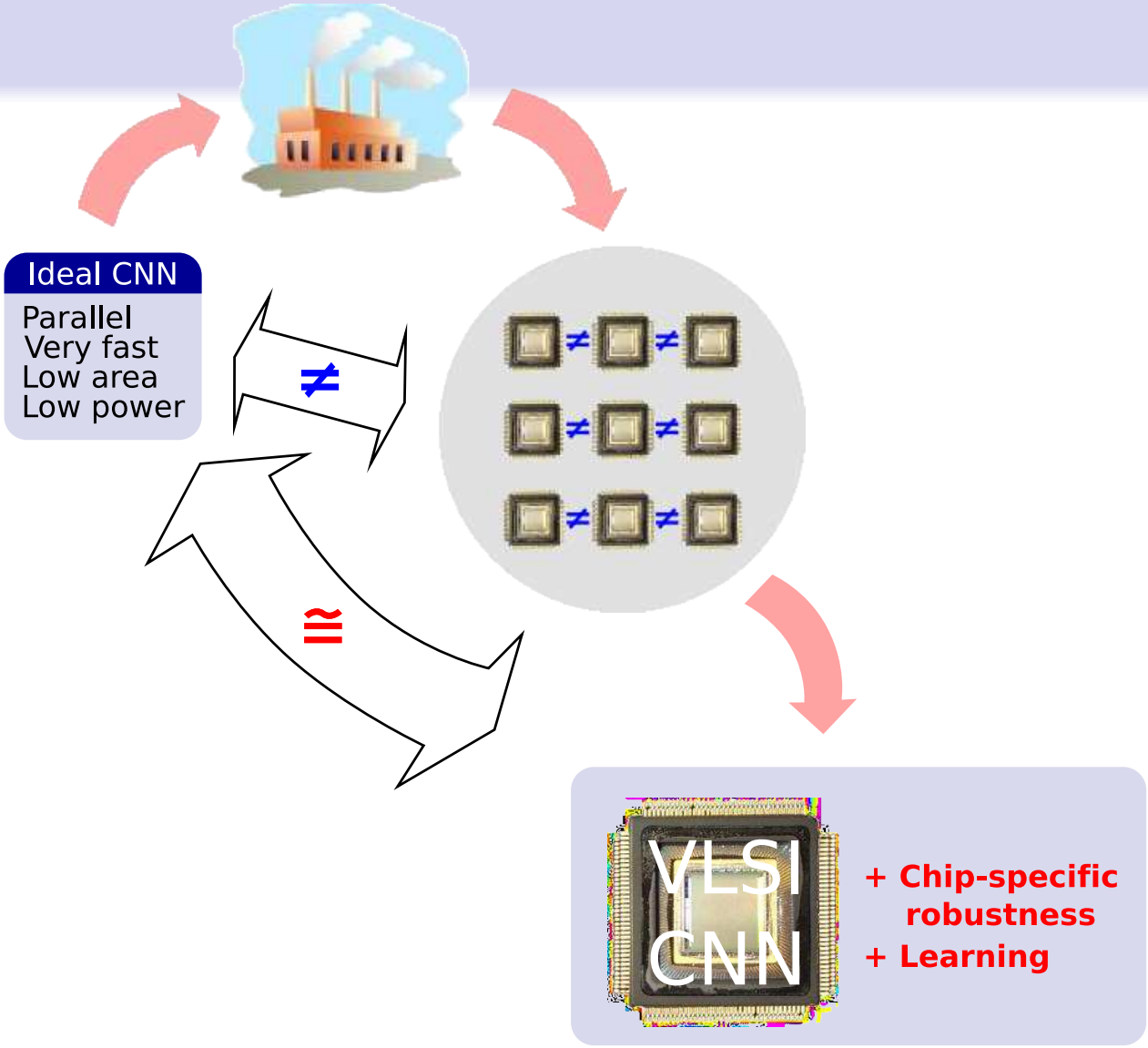
Conclusions



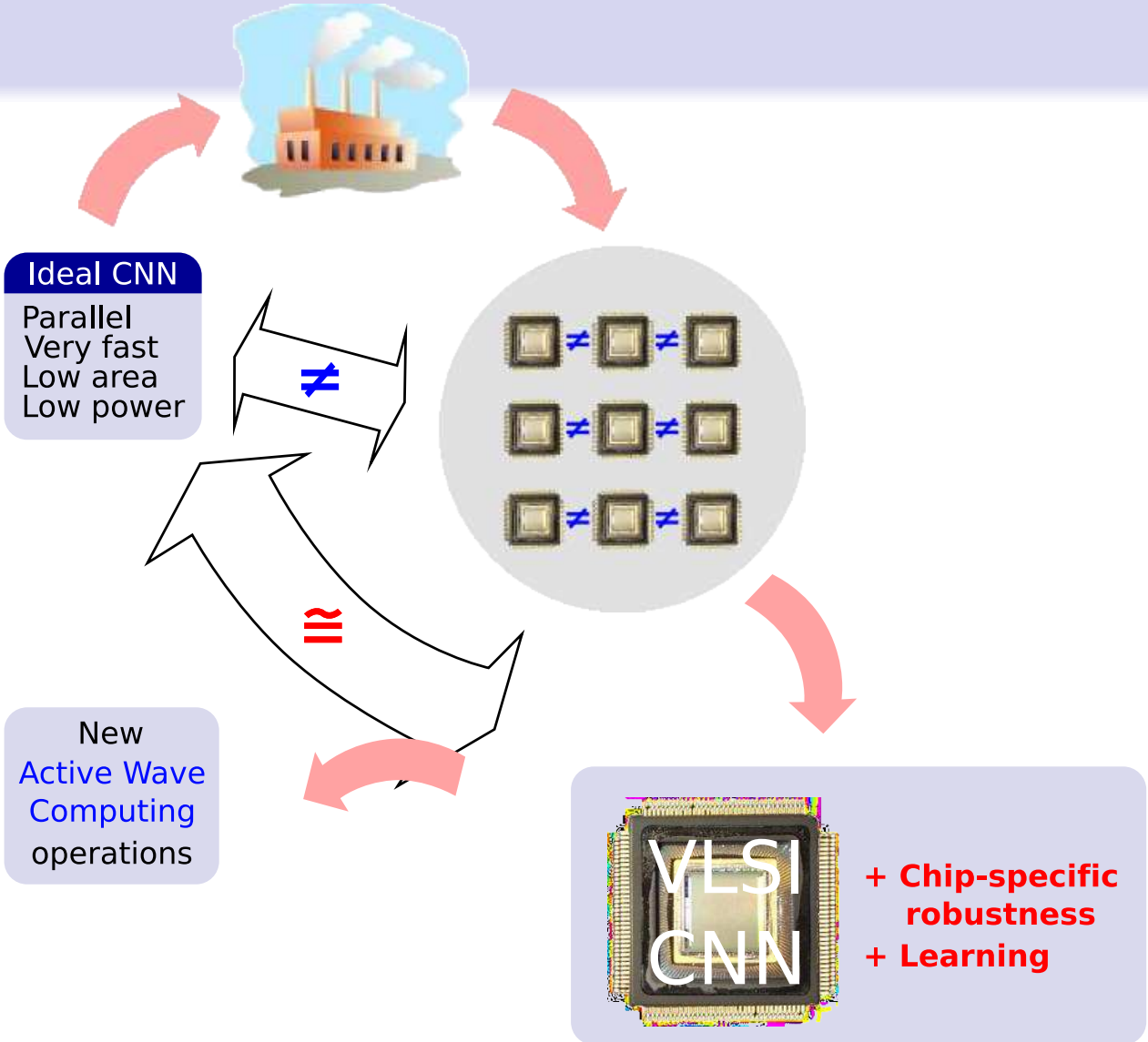
Conclusions



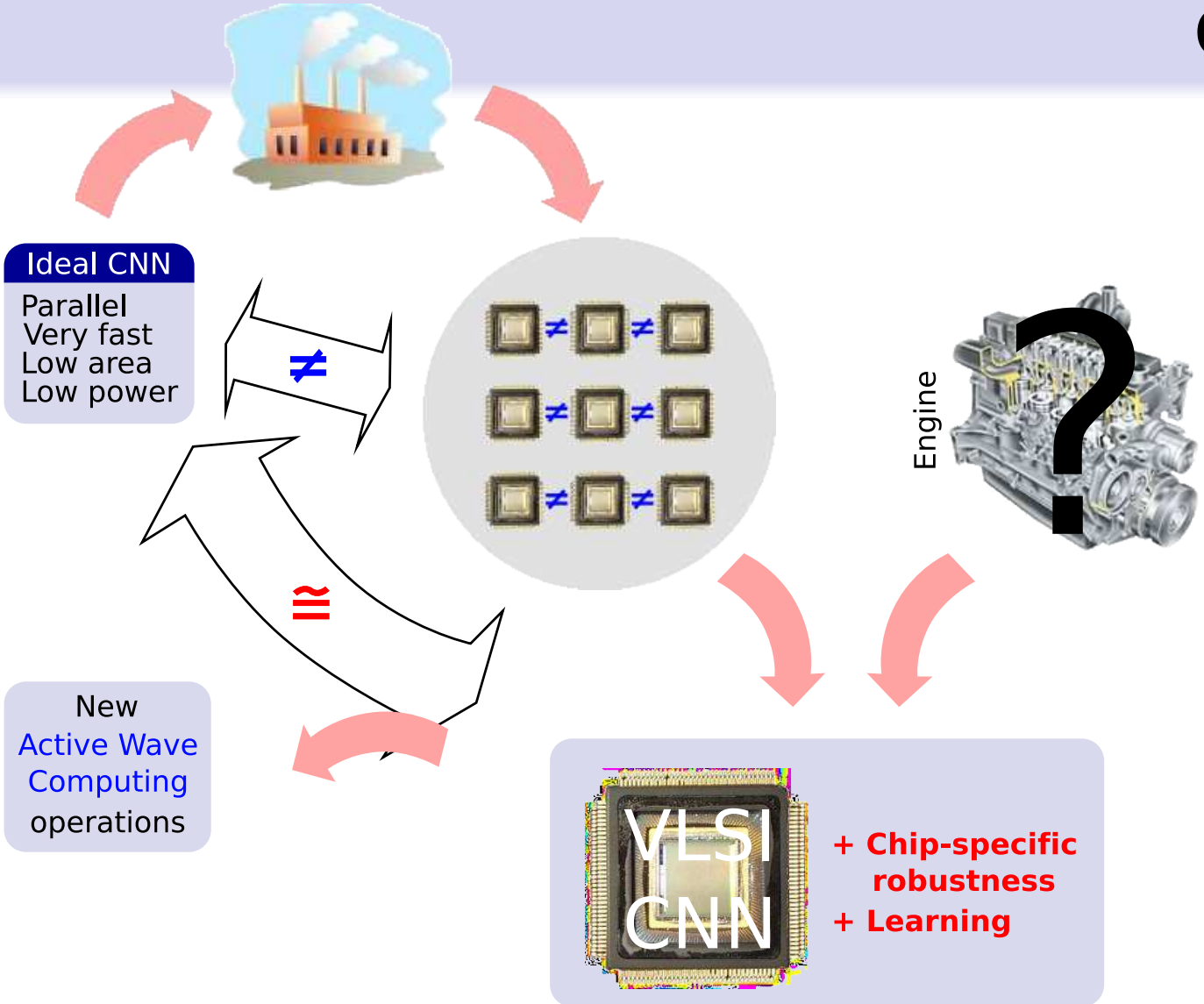
Conclusions



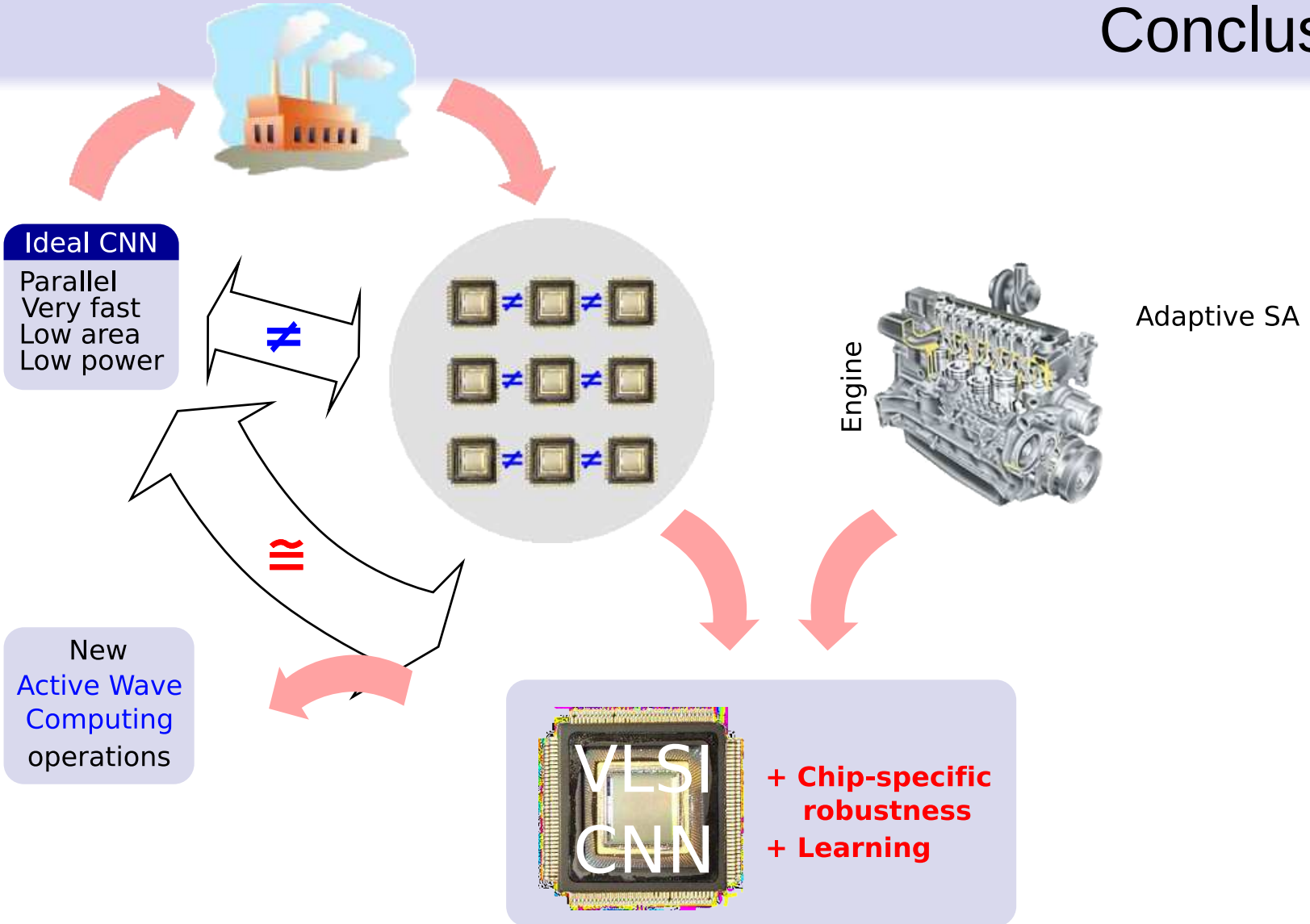
Conclusions



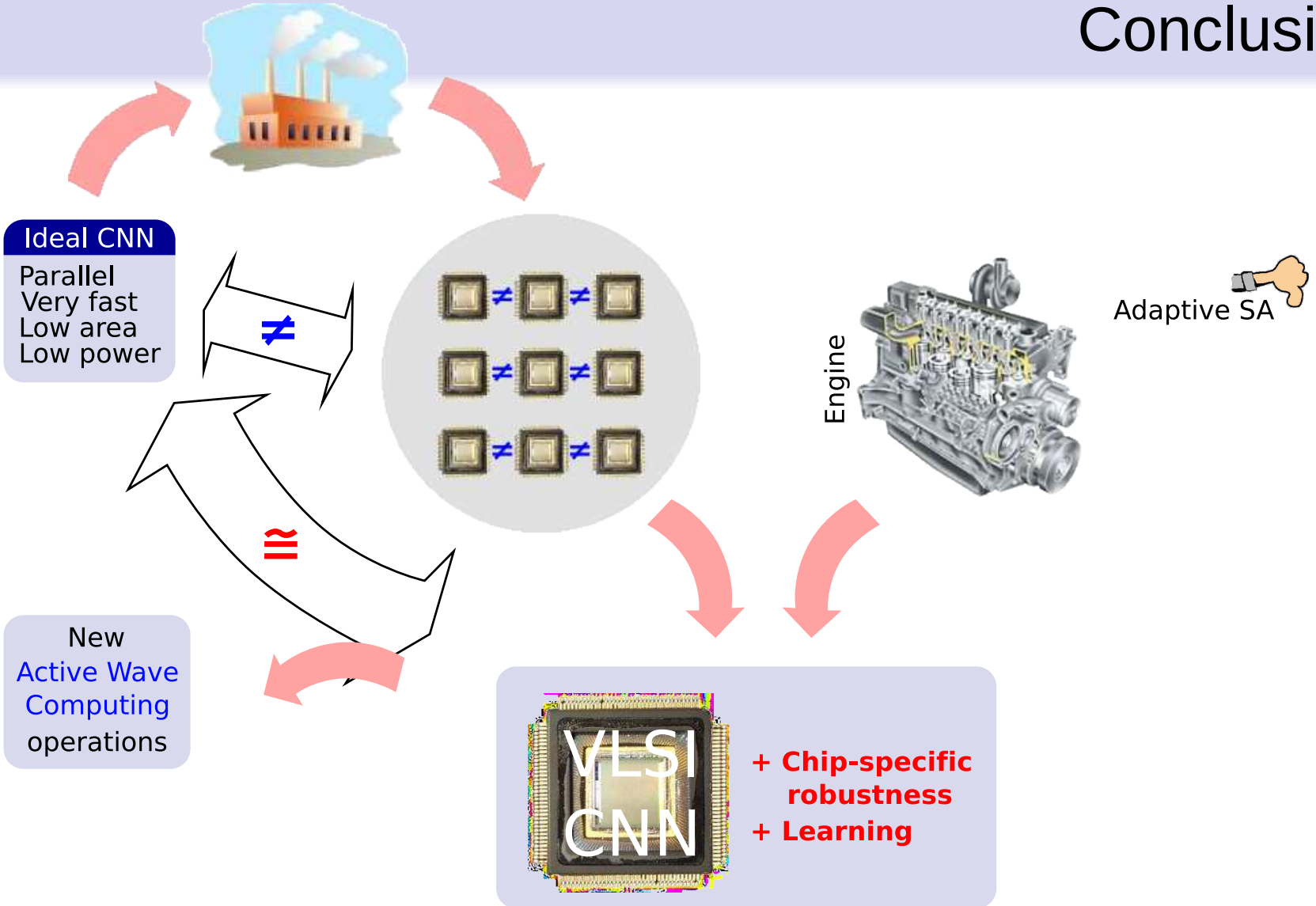
Conclusions



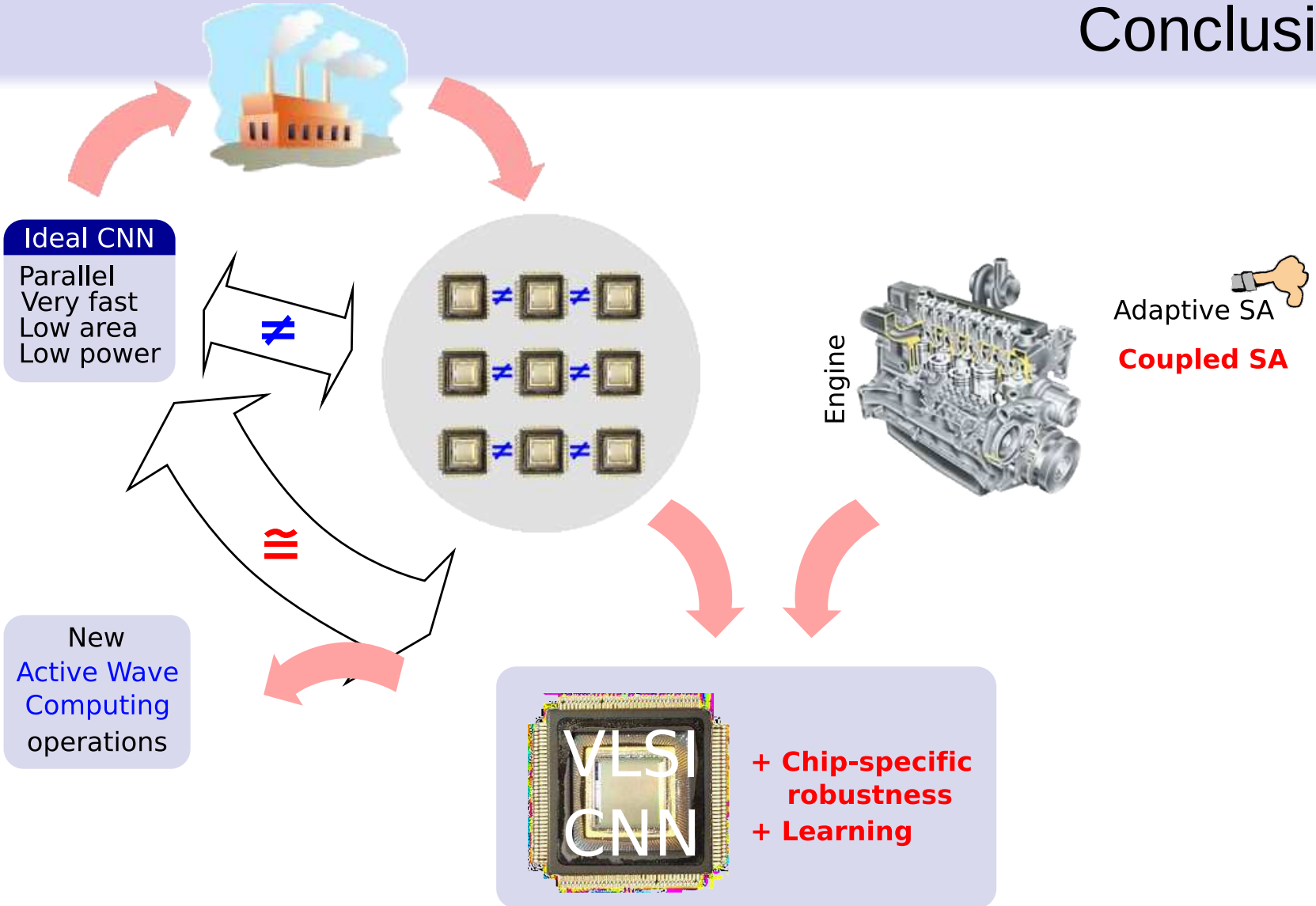
Conclusions



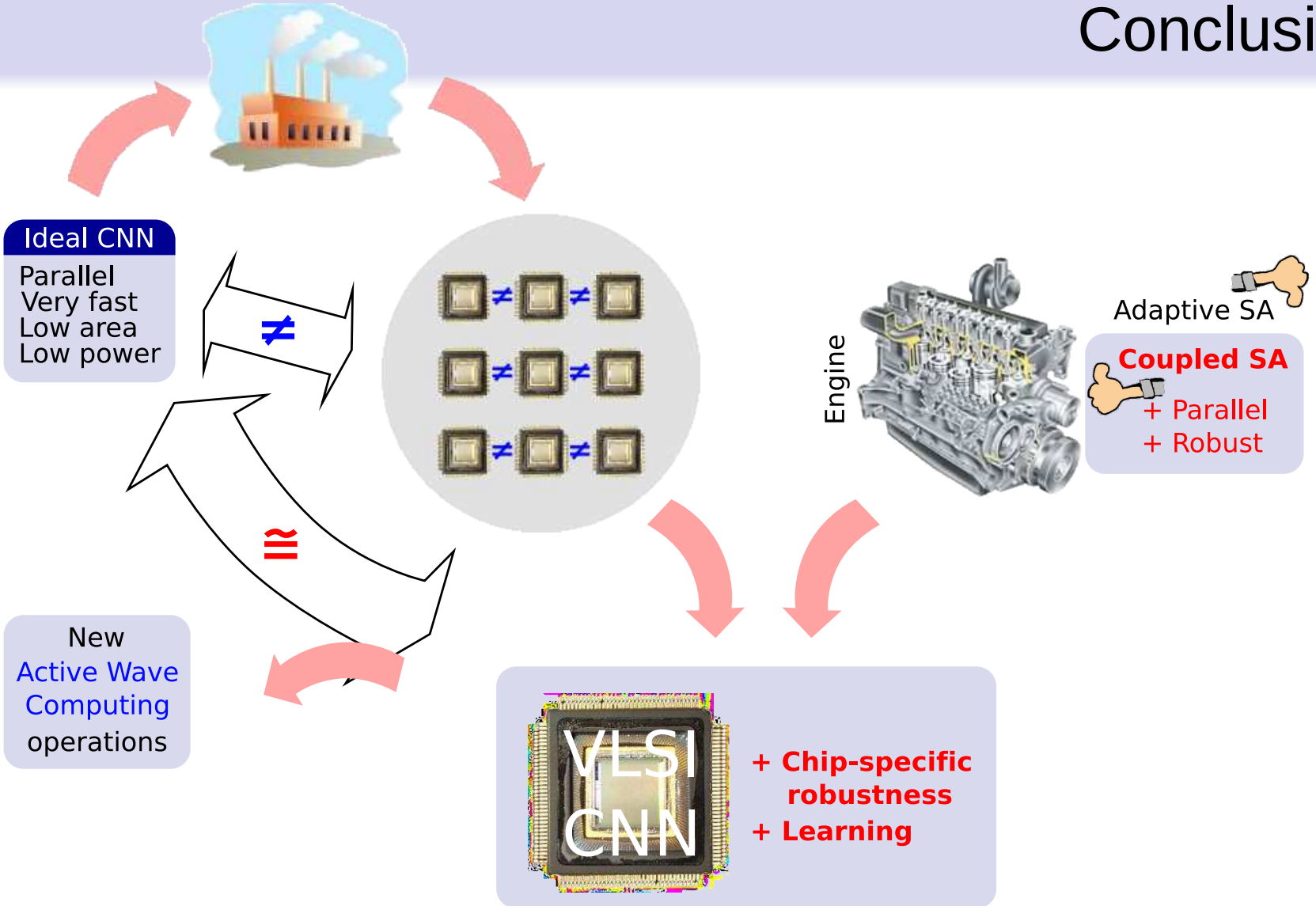
Conclusions



Conclusions



Conclusions

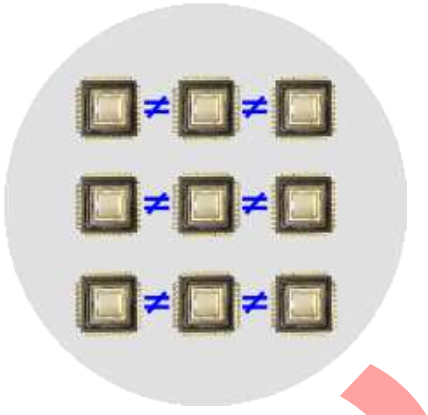


Conclusions



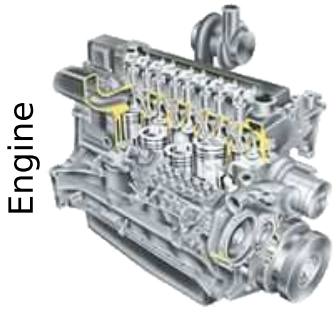
Ideal CNN
Parallel
Very fast
Low area
Low power

$\neq$



$\cong$

New
Active Wave
Computing
operations

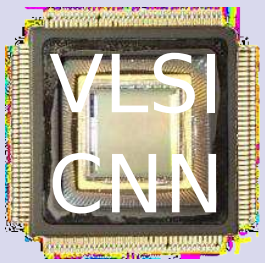


Engine

Adaptive SA

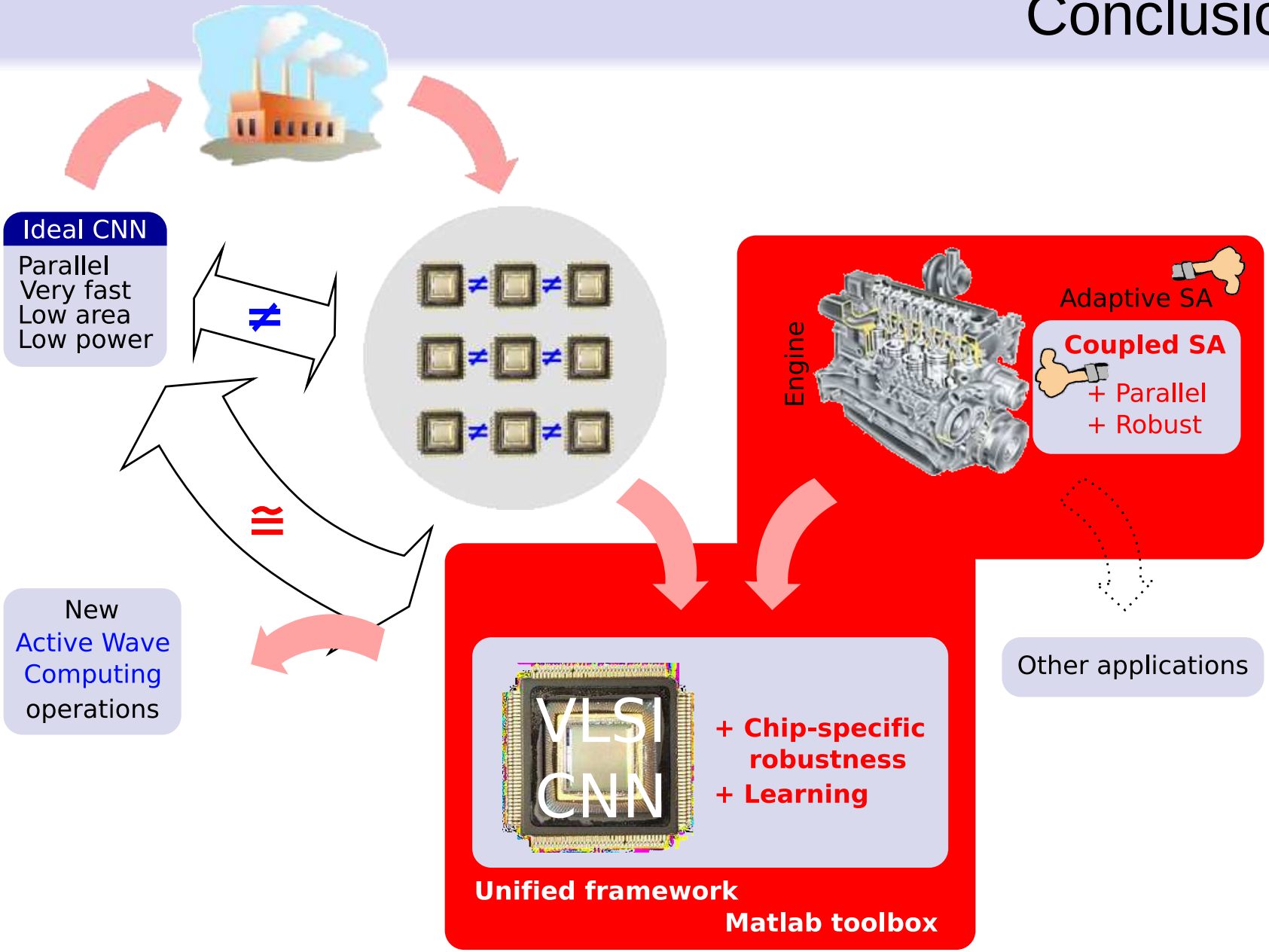
Coupled SA
+ Parallel
+ Robust

Other applications



+ Chip-specific robustness
+ Learning

Conclusions

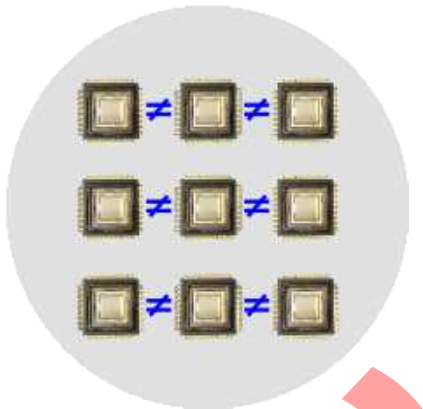


Conclusions



Ideal CNN
Parallel
Very fast
Low area
Low power

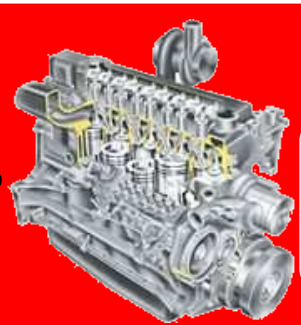
$\neq$



$\cong$

New
Active Wave
Computing
operations

Prototype applications
+ Real-time object tracking
+ Handsfree wheelchair
+ Airbourne mousing
+ Gaze estimation

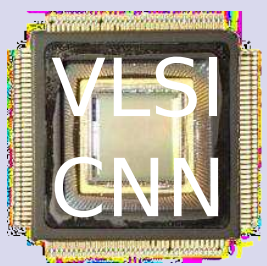


Engine

Adaptive SA

Coupled SA
+ Parallel
+ Robust

Other applications



+ Chip-specific robustness
+ Learning

Unified framework
Matlab toolbox

Outline

1 Introduction

2 Robust CNN Tuning and Learning

3 Coupled Simulated Annealing

4 Applying CSA to CNN Optimisation

5 Conclusions and Challenges for Future Work
- Conclusions
- Challenges for Future Work

"To be continued..."

- New template decomposition techniques
- Multi-layer emulation in single-layer CNN chips
- General tuning of hardware parameters
- On-chip tuning of CNN algorithms
- Self-tuning of CNN chips

Acknowledgements

- Promoters
  Johan Suykens, Joos Vandewalle
- Advisory committee
  André Barbé, Marc Van Hulle, Jan Van Impe
- Co-authors
  Desiré Bollé, Dániel Hillier,
  Michiel Van Dyck, Müştak E. Yalçın
- Prof. Támas Roska
  for the opportunity to visit his lab in Hungary and for the honor to have him in my Ph.D. jury
- SISTA colleagues
  for the quality work-time spent together