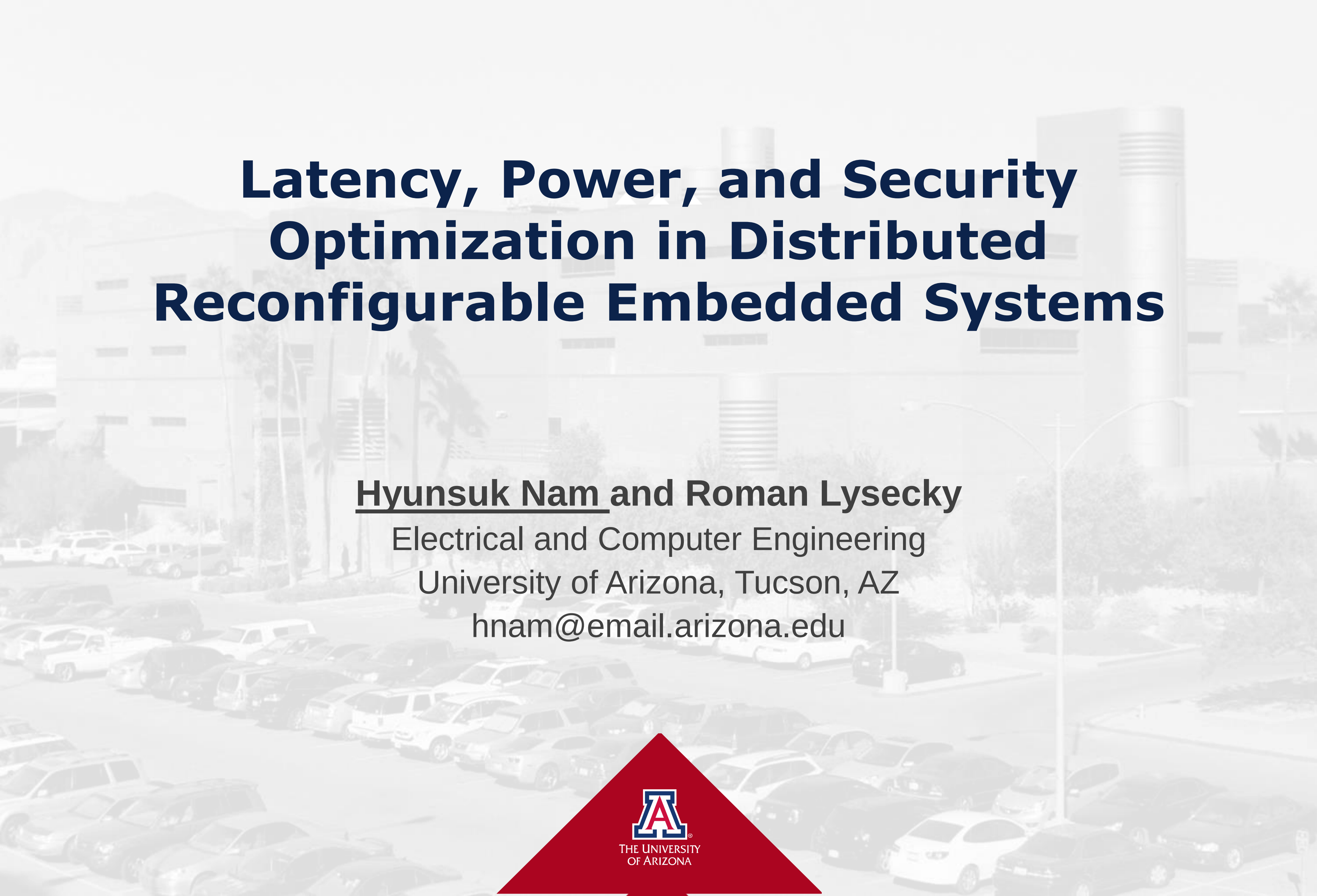
THE UNIVERSITY OF ARIZONA
College of Engineering

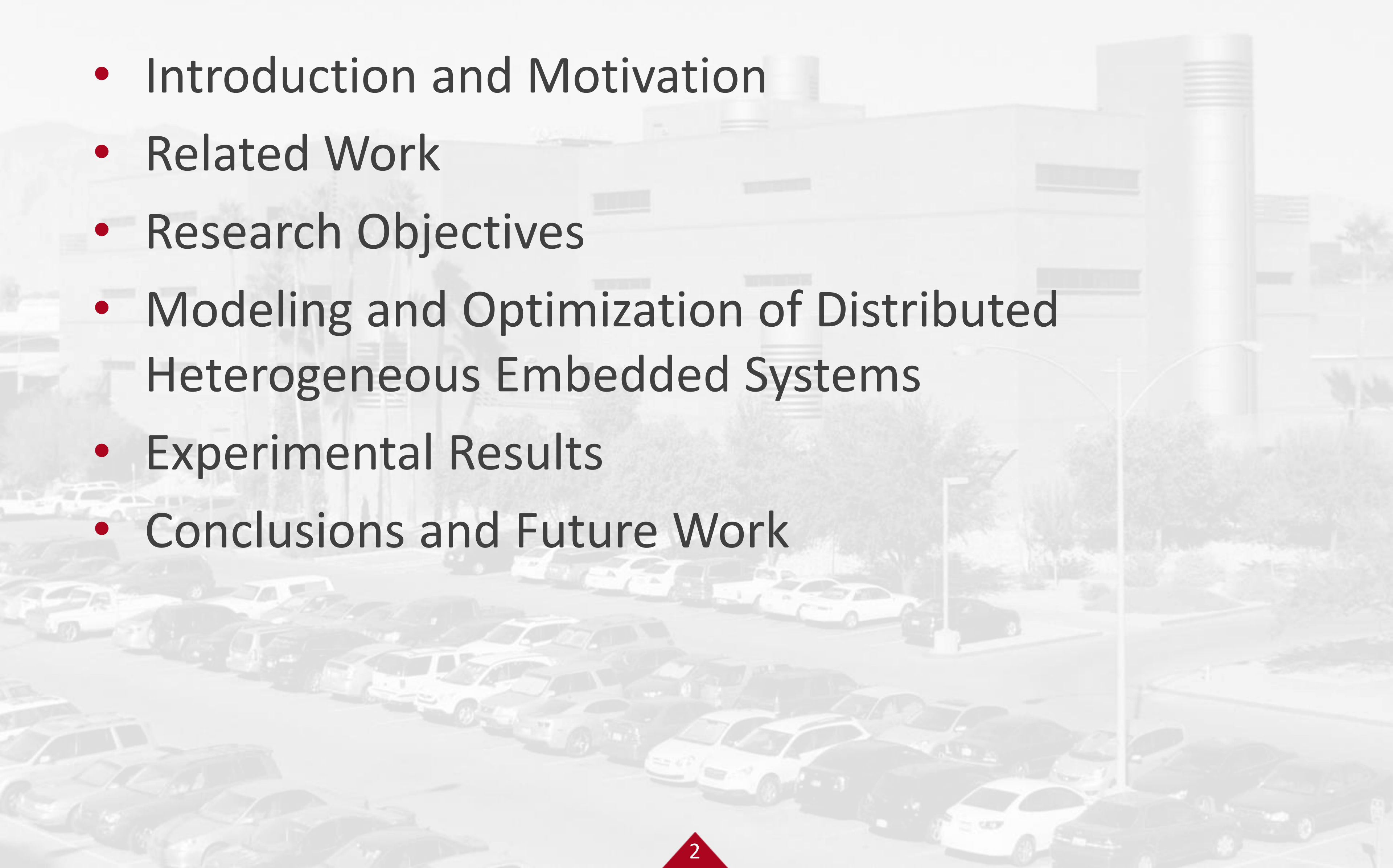


Latency, Power, and Security Optimization in Distributed Reconfigurable Embedded Systems

Hyunsuk Nam and Roman Lysecky
Electrical and Computer Engineering
University of Arizona, Tucson, AZ
hnam@email.arizona.edu

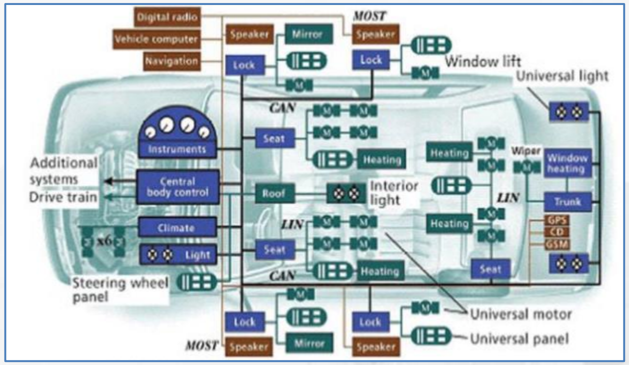
THE UNIVERSITY OF ARIZONA

Outline

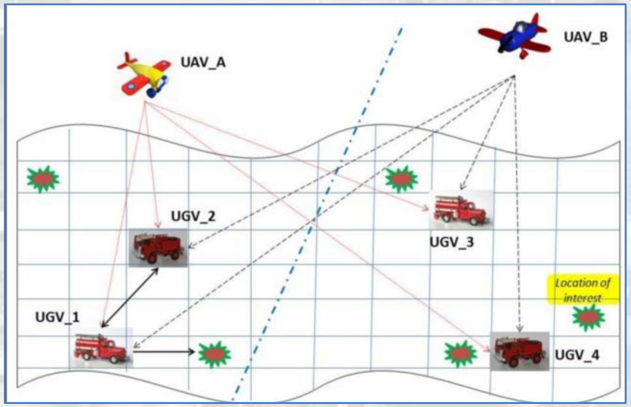


- Introduction and Motivation
- Related Work
- Research Objectives
- Modeling and Optimization of Distributed Heterogeneous Embedded Systems
- Experimental Results
- Conclusions and Future Work

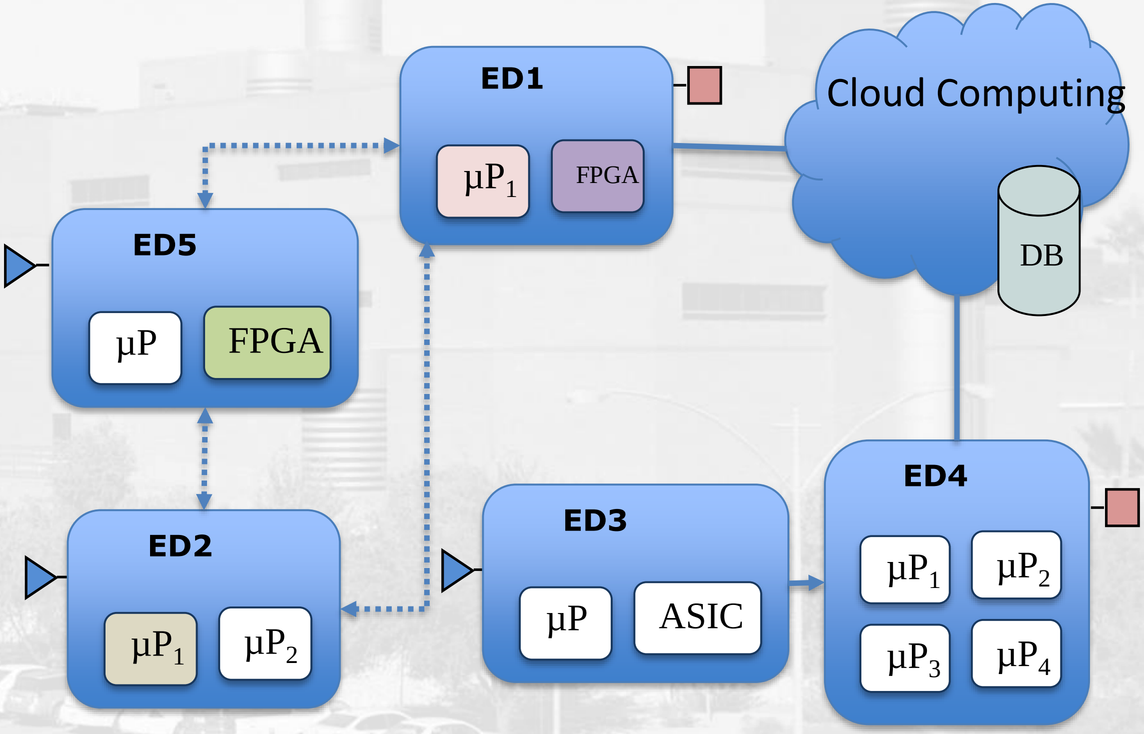
Introduction – Distributed Heterogeneous Embedded Systems



Automotive systems

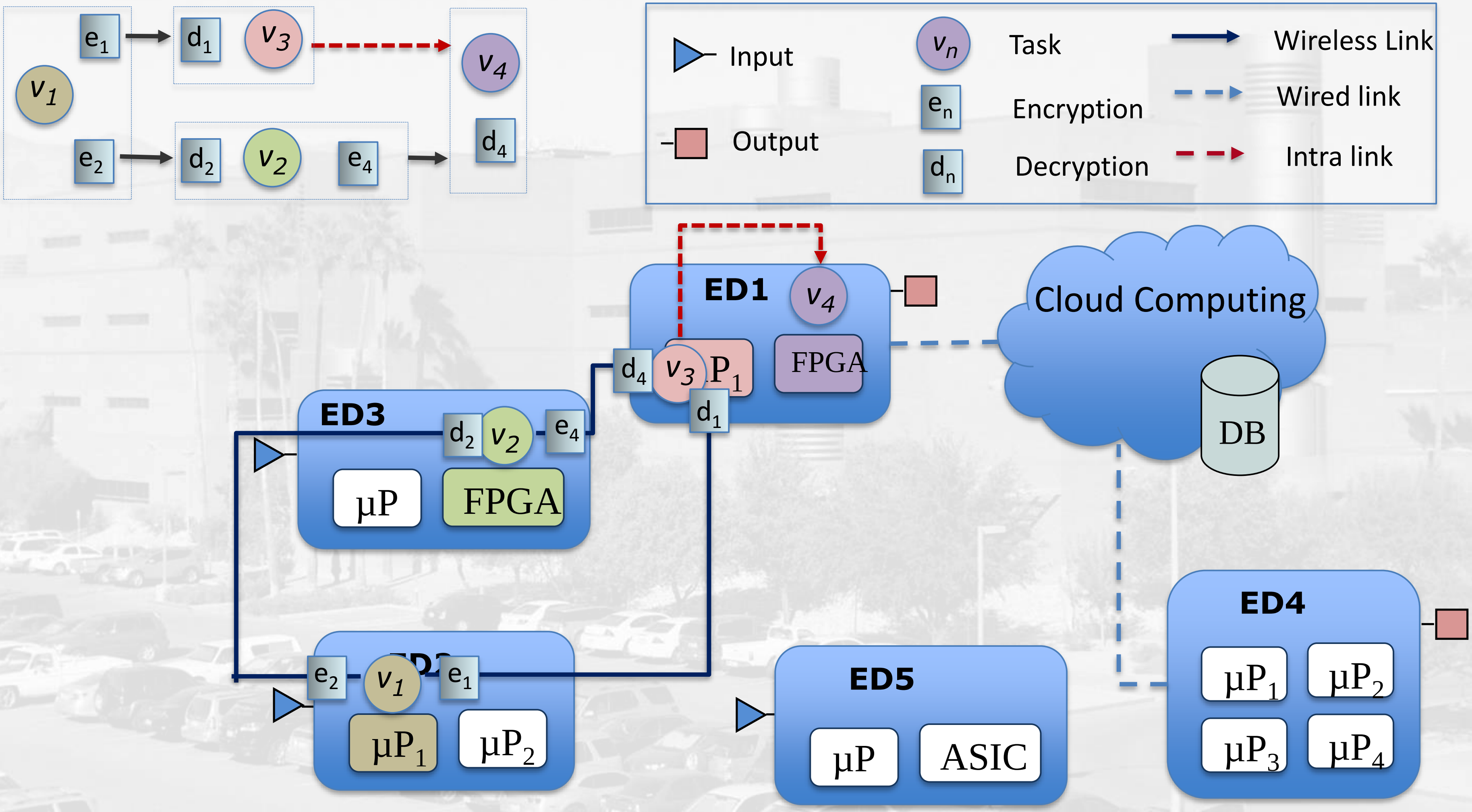


Object Detection & Tracking



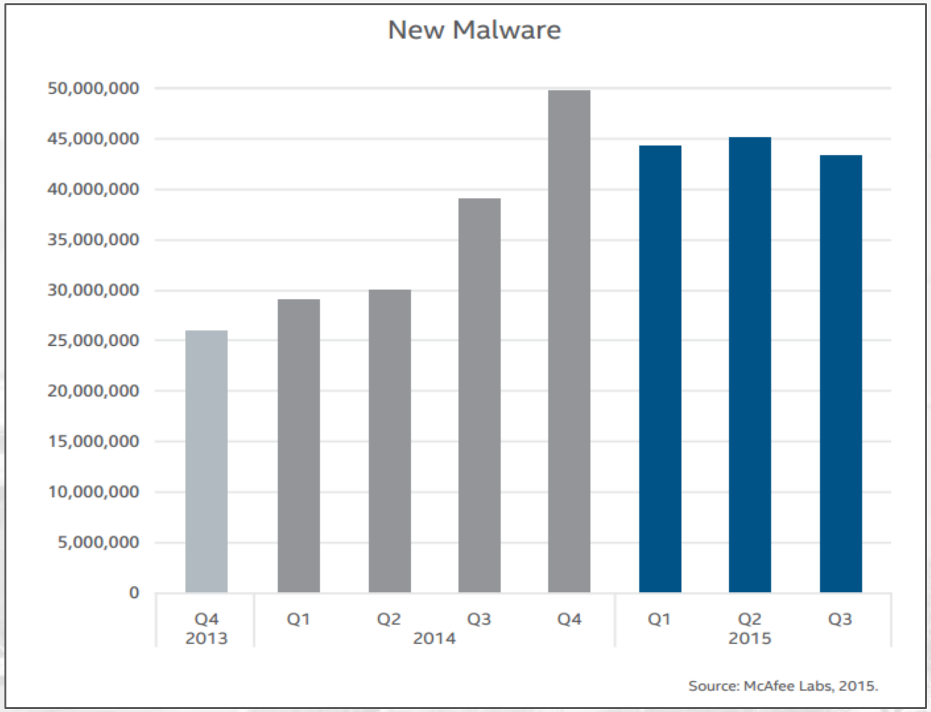
Distributed embedded system are composed of heterogeneous computing resources including processors, FPGA, and custom HW

Introduction – Distributed Heterogeneous Embedded Systems

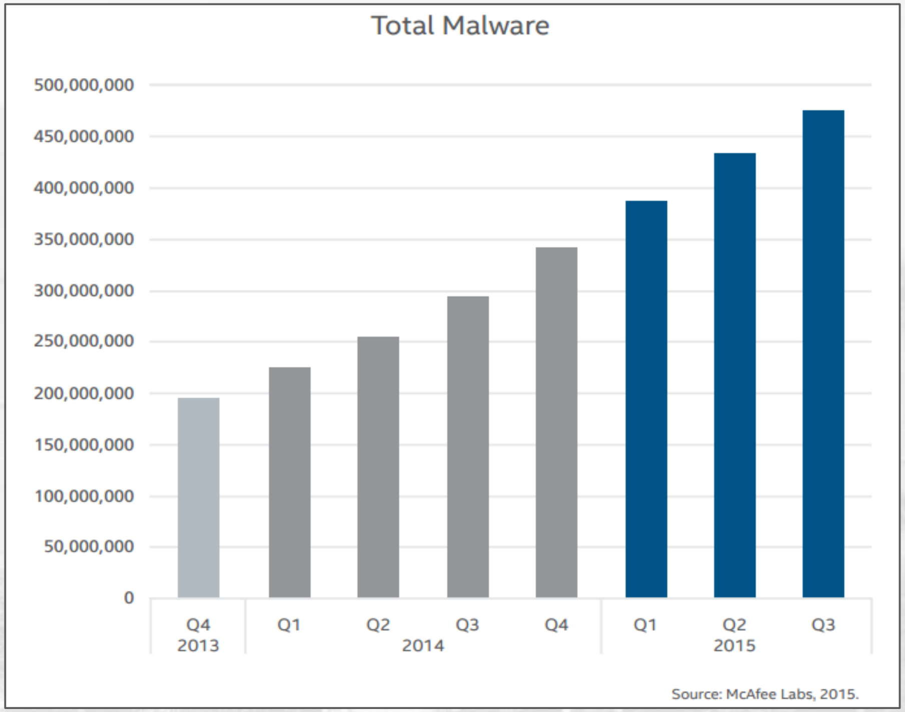


Traditional approaches to security focus on cryptography only for inter-device communication

Introduction - Malware



Source: McAfee Labs, 2015.



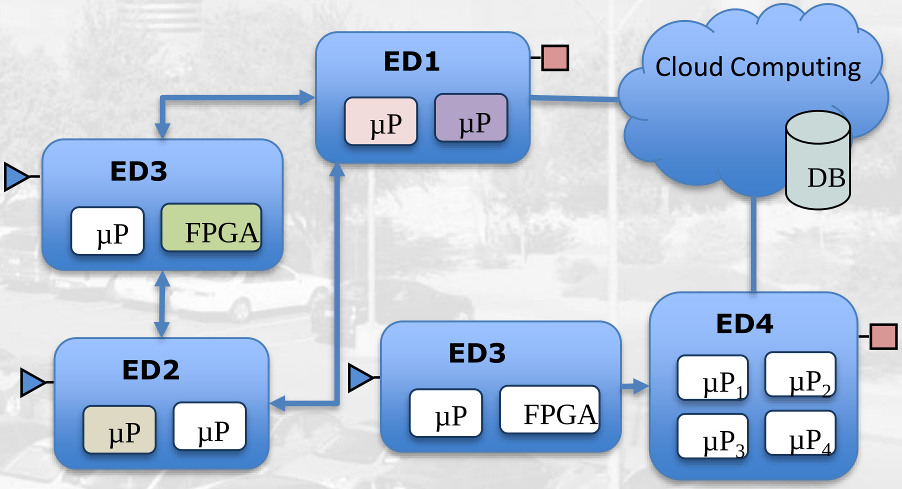
Source: McAfee Labs, 2015.

McAfee Labs Threats Report: Nov. 2015

- Malware growing at an alarming rate
  - 100,000 new malware every day
- Malware can affect both SW and HW
  - FPGAs are reconfigurable and can potentially be reconfigured by malicious software

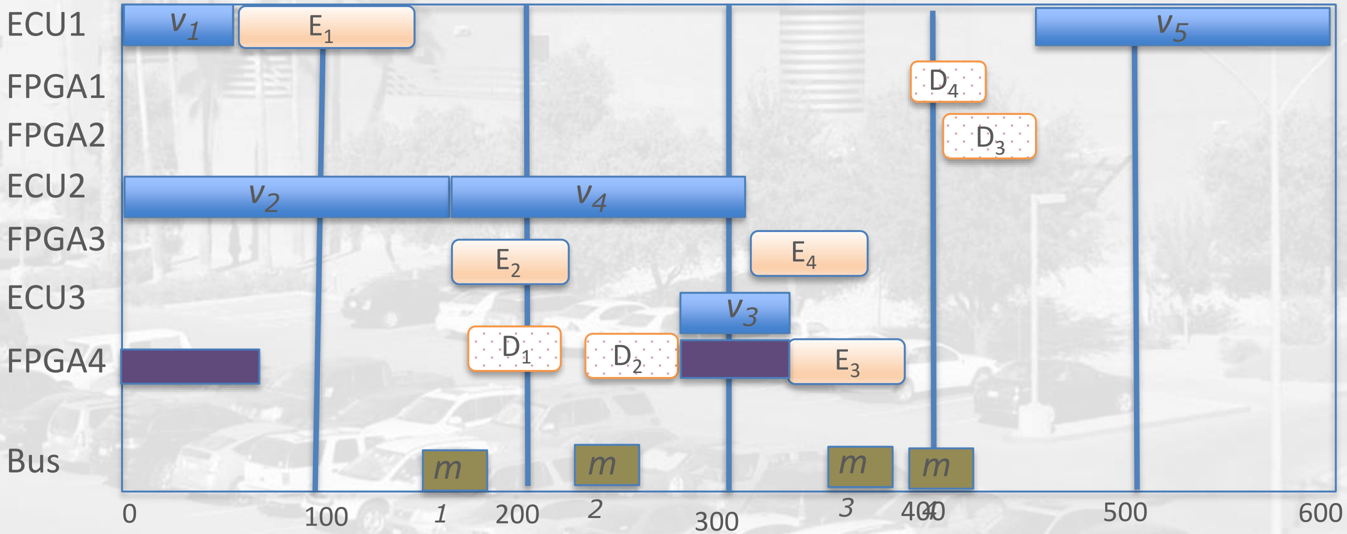
Introduction – Integrating Security within Design Process

- Goal: Security needs to be integrated within the design and optimization
  - Equally as important as other evaluation metrics (e.g., latency, energy)
- Need for method to quantify security within the design process
  - Enable ability to analyze the impact of different cryptographic implementations



Related Work - Integrating Security within Design Process (1)

- Hardware/software co-design for secure automotive systems [Jiang et al., DATE 2012]
  - Mapping and scheduling of tasks for ECUs
  - Cryptography used for inter-ECU communication
  - Goal is to optimize the number AES cryptography rounds



K. Jiang, P. Eles, and Z. Peng, "Co-Design Techniques for Distributed Real-Time Embedded Systems with Communication Security Constraints. *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 947-952, March 2012.

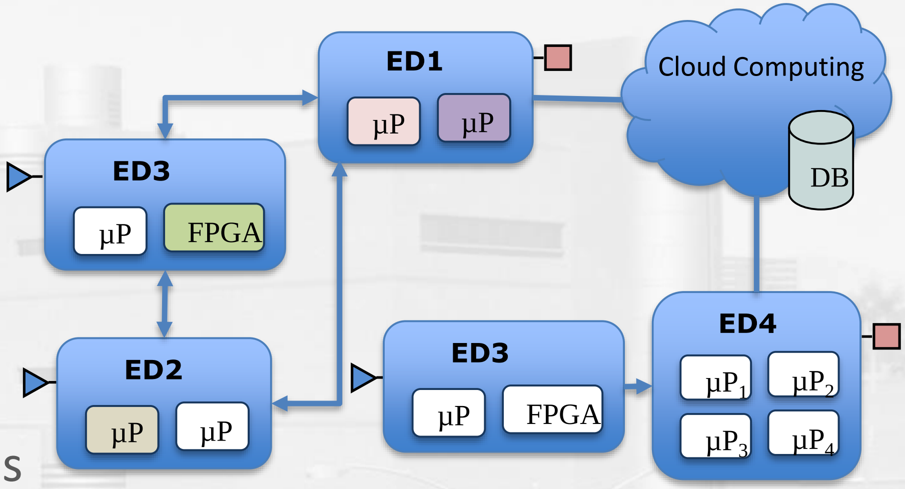
Related Work - Integrating Security within Design Process (1)

- Task allocation and network scheduling [Selicean & Pop, ACM TECS 2015]
  - Design Methodology security-aware authentication supporting FlexRay, Time Triggered protocol
  - Determine task allocation, priority assignment, network scheduling, and key release
  - Goal is to minimize the summation of the worst-case latency
  - Seek to optimize the cryptography and authentication methods utilized within distributed automotive electronics
  - Does not consider intra-device cryptography, wireless communication/ or energy constraints

D. T. Selicean and D. P. Pop, “Design Optimization of Mixed-Criticality Real-Time Embedded Systems,” ACM Transactions on Embedded Computing Systems (TECS), vol. 14, no. 50, May 2015.

Objectives

- **Objectives and goals:**
  - Design methodology for optimizing dataflow application using distributed, heterogeneous, and reconfigurable embedded systems
  - Consider embedded devices incorporating reconfigurable FPGAs, supporting mapping of tasks between SW and HW alternatives
  - Support cryptography between all tasks implementations, including inter- and intra-device, SW and HW
  - Develop integrated modeling framework for computation, communication, security, and power
  - Optimize security, latency, power consumption given constraints on other metrics
  - Define security levels for quantify to trade-off power and security



Security Levels

- Security Level
  - Defines a relative ranking of strength of the selected cryptography method
  - Can be used to rank different cryptographic alternatives and configurations thereof

Security Level	Key Size (bits)	Number Rounds
12	256	14
11	256	13
10	256	12
9	256	11
8	256	10
7	192	12
6	192	11
5	192	10
4	192	9
3	128	10
2	128	9
1	128	8
0	0	0

Parameterized Dataflow Application Model

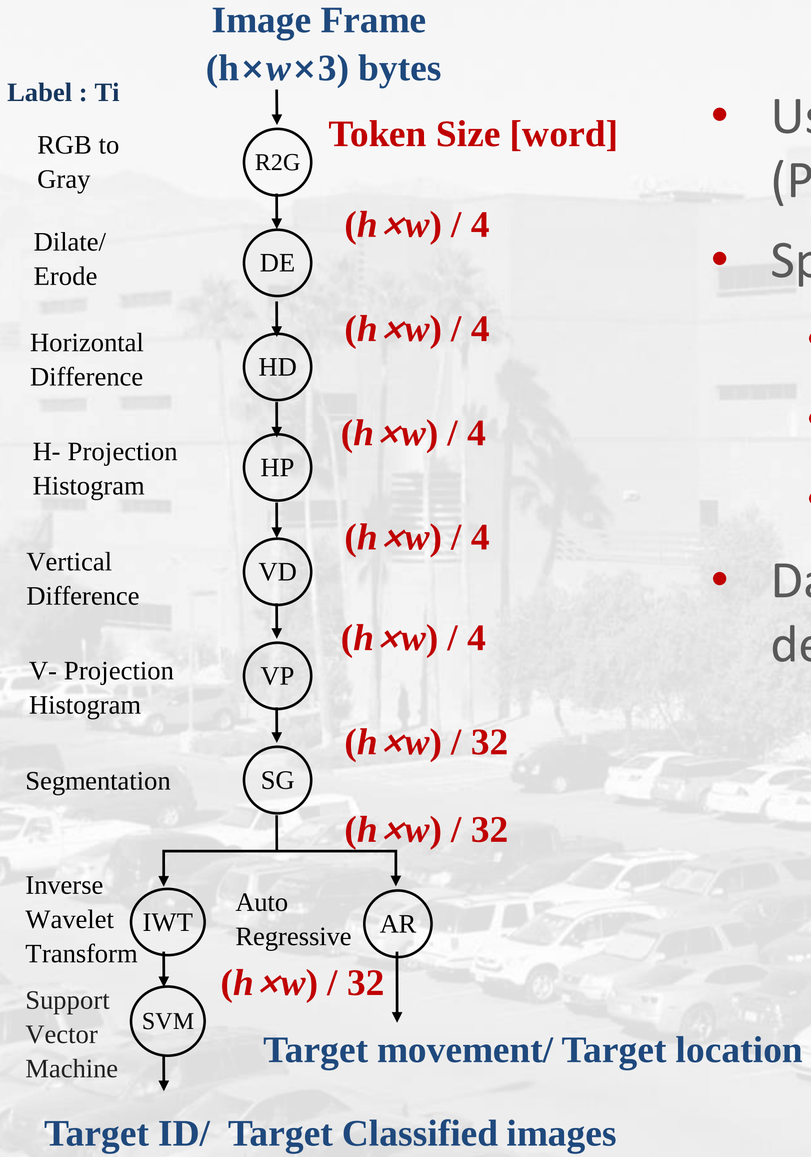


Image Frame
(h×*w*×3) bytes

Label : Ti

Token Size [word]

RGB to Gray — R2G

(*h* ×*w*) / 4

Dilate/ Erode — DE

(*h* ×*w*) / 4

Horizontal Difference — HD

(*h* ×*w*) / 4

H- Projection Histogram — HP

(*h* ×*w*) / 4

Vertical Difference — VD

(*h* ×*w*) / 4

V- Projection Histogram — VP

(*h* ×*w*) / 32

Segmentation — SG

(*h* ×*w*) / 32

Inverse Wavelet Transform — IWT

Auto Regressive — AR

(*h* ×*w*) / 32

Support Vector Machine — SVM

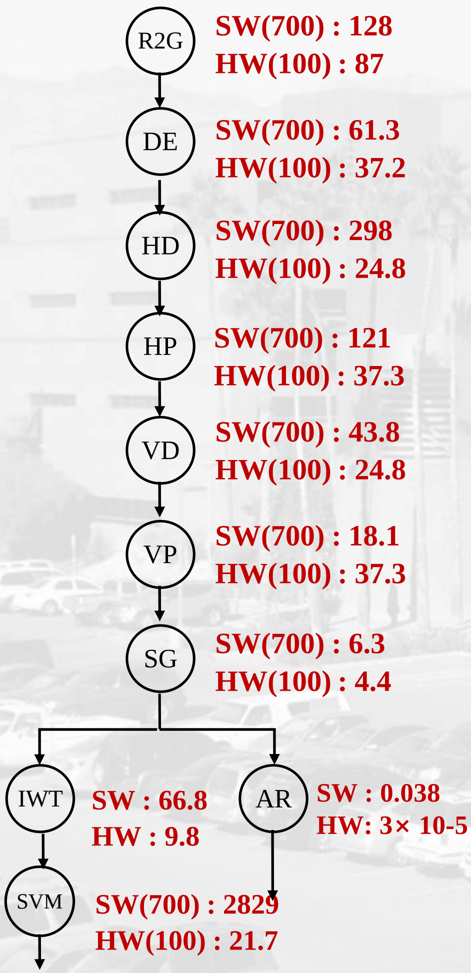
Target movement/ Target location

Target ID/ Target Classified images

- Use Parameterized synchronous dataflow (PSDF) model
- Specify
  - System tasks
  - Parameterizable data sizes
  - Tokens transmitted between tasks
- Dataflow model for a video-based object detection and tracking application

Execution Latency Model

Task Execution latency [ms]



R2G
SW(700) : 128
HW(100) : 87

DE
SW(700) : 61.3
HW(100) : 37.2

HD
SW(700) : 298
HW(100) : 24.8

HP
SW(700) : 121
HW(100) : 37.3

VD
SW(700) : 43.8
HW(100) : 24.8

VP
SW(700) : 18.1
HW(100) : 37.3

SG
SW(700) : 6.3
HW(100) : 4.4

IWT
SW : 66.8
HW : 9.8

AR
SW : 0.038
HW: 3× 10-5

SVM
SW(700) : 2829
HW(100) : 21.7

- Specifies software and hardware task alternatives
  - Assumes all tasks can be implemented in HW or SW
- Software Latency
  - Latency of a task is based upon physical measurement from specific device
  - Linear scaling is applied to adjust for specific processor frequency
- Hardware Latency
  - Latency is measured in clock cycles based on RTL simulation
  - Frequency of hardware is limited by ED’s system bus or synthesis results
  - Hardware size if constrained to size of reconfigurable region/tile

Communication Latency Model

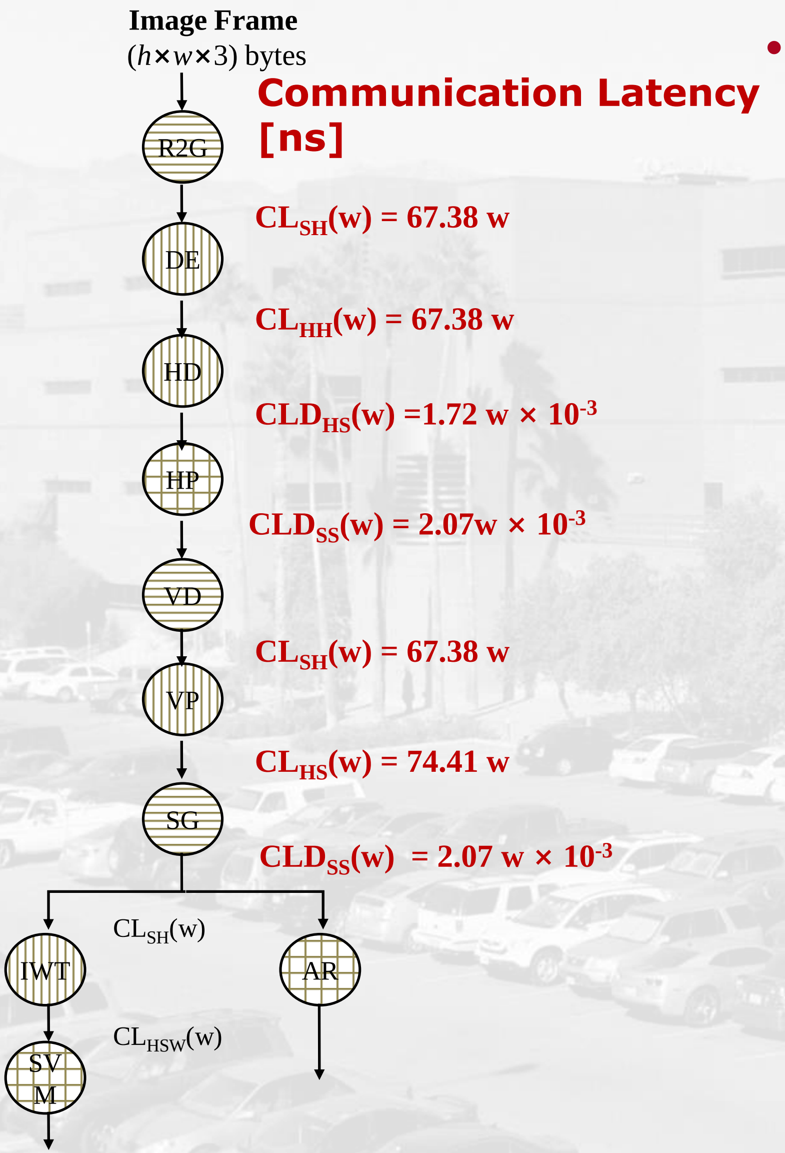


Image Frame
(h×w×3) bytes

Communication Latency [ns]

$CL_{SH}(w) = 67.38\ w$

$CL_{HH}(w) = 67.38\ w$

$CLD_{HS}(w) = 1.72\ w \times 10^{-3}$

$CLD_{SS}(w) = 2.07w \times 10^{-3}$

$CL_{SH}(w) = 67.38\ w$

$CL_{HS}(w) = 74.41\ w$

$CLD_{SS}(w) = 2.07\ w \times 10^{-3}$

$CL_{SH}(w)$

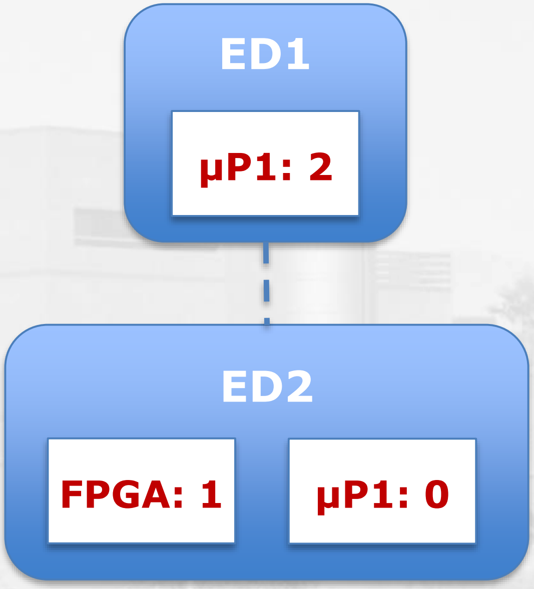
$CL_{HSW}(w)$

- Communication latency model
  - Use physical measurements to determine latency for different modes for communication and size of tokens
  - Using IEEE 802.11g
  - Eight possible communication modes for transferring data between tasks, which depends on the task implementation

$CL_{SS}(w)$	Intra-device communication
$CL_{SH}(w)$	
$CL_{HS}(w)$	
$CL_{HH}(w)$	
$CLD_{SS}(w)$	Inter-device communication
$CLD_{HS}(w)$	
$CLD_{SH}(w)$	
$CLD_{HH}(w)$	

Power Model

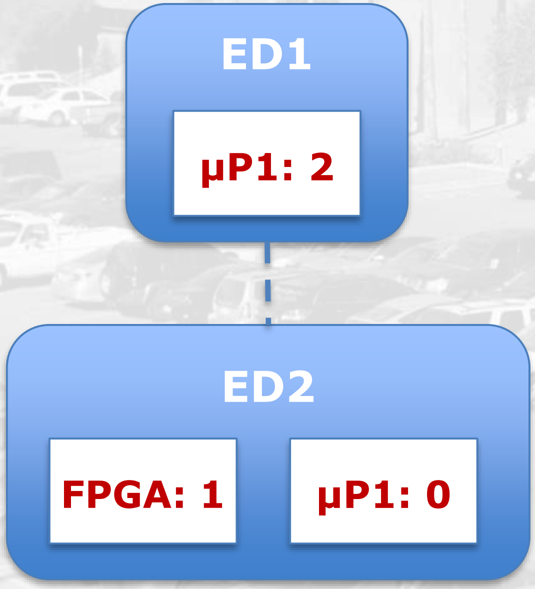
$$P_{ED} = P_{SW} + P_{HW} + P_C + P_S$$

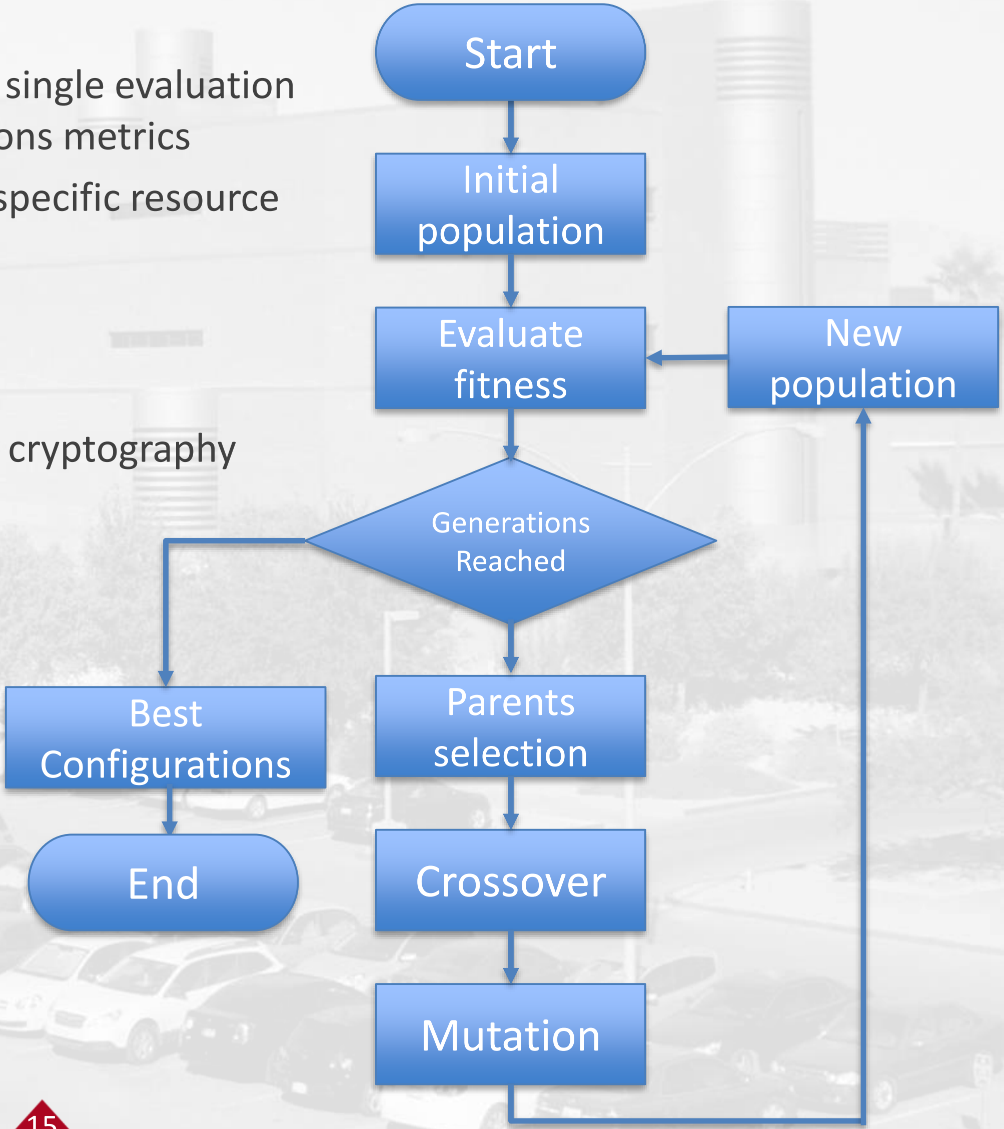


- SW Power (P_{SW})
  - Characterizes the active and idle power consumption of each µP
- HW Power (P_{HW})
  - RTL implementation for each hardware task
  - Post-synthesis power estimation
- Communication Power (P_C)
  - Physical measurements of communication middleware on EDs
  - Latency based on data transferred, operating frequency, and communication mode
- Security Power consumption (P_S)
  - Utilized prototype SW and HW implementations for each SL
  - Created regression model based and key, rounds, and data size

Design Space Exploration Methodology

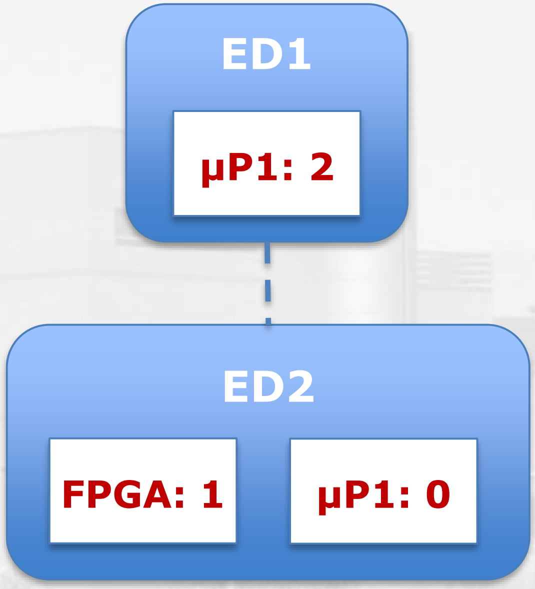
- Genetic Optimization Algorithm
  - Single-objective optimization optimizes a single evaluation metric with constraints on other evaluations metrics
  - **Task Chromosome** : Mapping of tasks to specific resource type on specific ED
    - **0 for software on ED2**
    - **1 for hardware on ED2**
    - **2 for software on ED1**
  - **Security Chromosome :** Security level for cryptography between all tasks communications





Experimental Setup

- Distributed Heterogeneous Embedded Systems Architecture
  - Embedded Device 1 (ED1)
    - 1.2 GHz ARM Cortex A-15
  - Embedded Device 2 (ED2)
    - 700 MHz ARM Cortex A-15
    - Reconfigurable FPGA (based on Spartan-6)
      - 100 MHz
      - Up to 8 reconfigurable regions
      - Hardware defines size of FPGA
  - Wireless Communication
    - IEEE 802.11 G
- Cryptography
  - AES: Rounds from 8-14 and key size of 128 to 256 bits
- End-to-end Latency Constraints
  - Base latency constraint: 8 sec
  - Relaxed latency constraint: 12 sec



Example task mapping for a particular system configuration

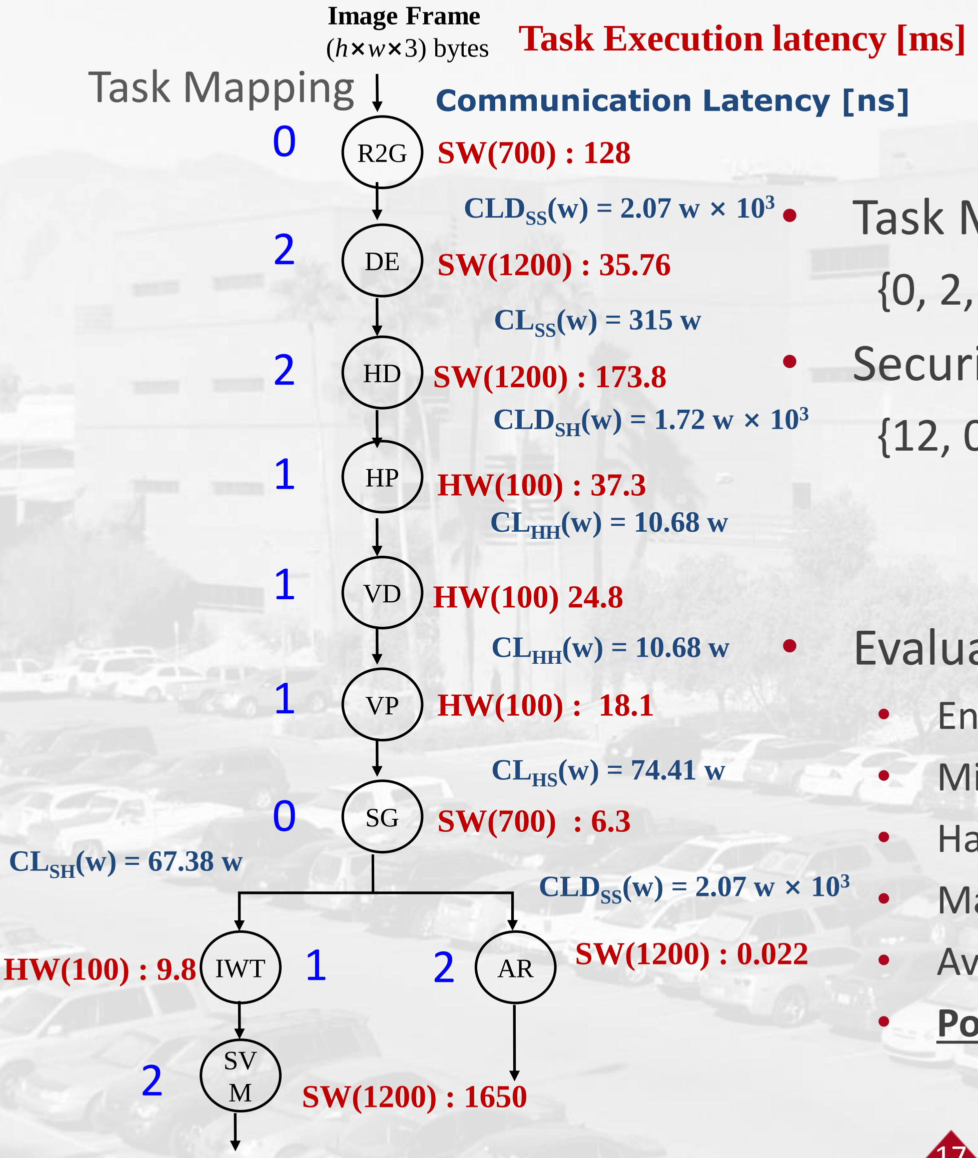


Image Frame
(h×w×3) bytes

Task Execution latency [ms]

Task Mapping

Communication Latency [ns]

0 R2G SW(700) : 128

$CLD_{SS}(w) = 2.07\ w \times 10^3$

2 DE SW(1200) : 35.76

$CL_{SS}(w) = 315\ w$

2 HD SW(1200) : 173.8

$CLD_{SH}(w) = 1.72\ w \times 10^3$

1 HP HW(100) : 37.3

$CL_{HH}(w) = 10.68\ w$

1 VD HW(100) 24.8

$CL_{HH}(w) = 10.68\ w$

1 VP HW(100) : 18.1

$CL_{HS}(w) = 74.41\ w$

0 SG SW(700) : 6.3

$CL_{SH}(w) = 67.38\ w$

$CLD_{SS}(w) = 2.07\ w \times 10^3$

HW(100) : 9.8 IWT 1

2 AR SW(1200) : 0.022

2 SVM

SW(1200) : 1650

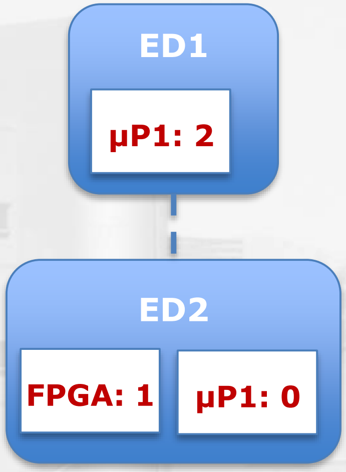
- Task Mapping

  {0, 2, 2, 1, 1, 1, 0, 1, 2, 2}

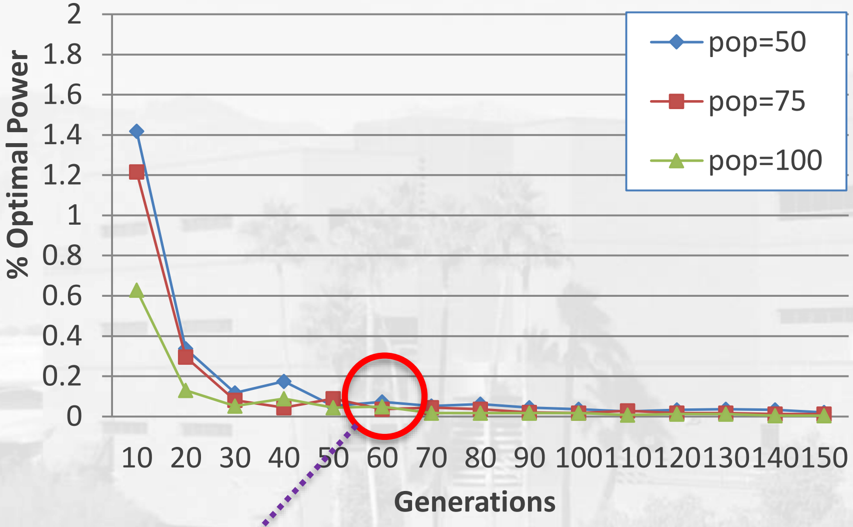
- Security Level Mapping

  {12, 0, 2, 4, 10, 8, 5, 0, 5}

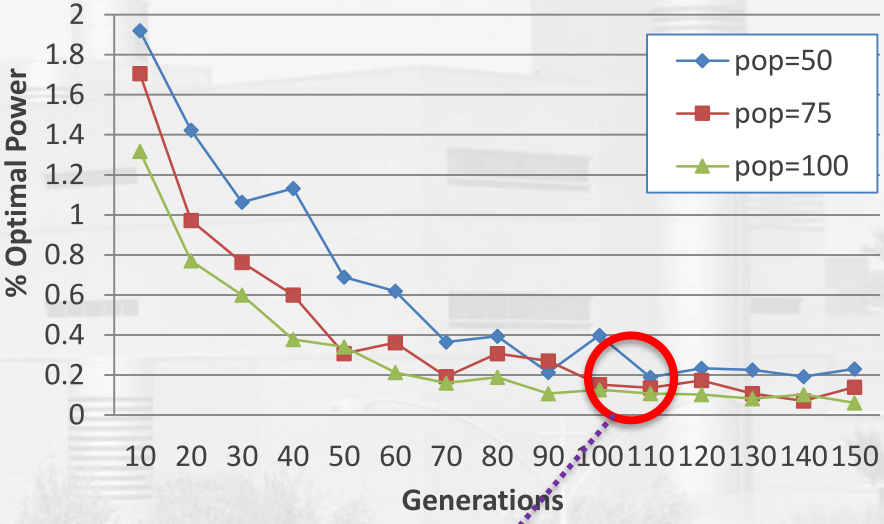
- Evaluation Metrics
  - End-to-end Latency: 11.15sec
  - Min Security Level: 0
  - Hardware Task: 4
  - Maximum Security Level:12
  - Average SL: 5.11
  - **Power: 3.98 [W]**



Experimental Results – Genetic Optimization Algorithm



Base Latency Constraint



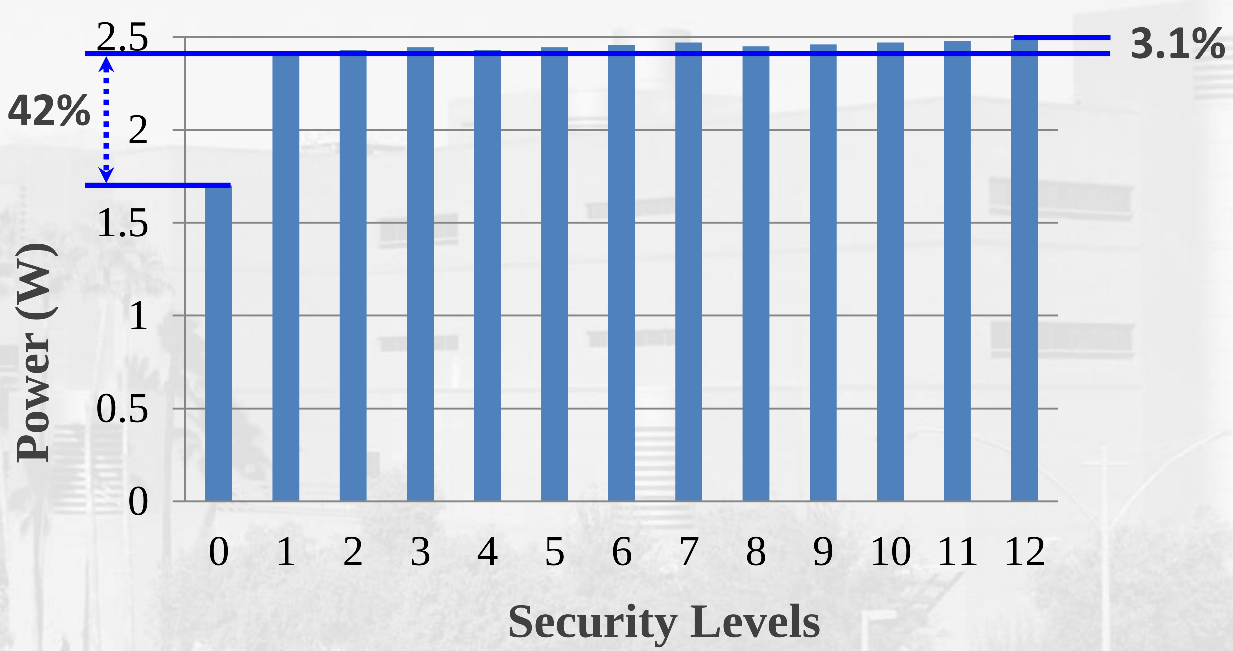
Relaxed Latency Constraint

For base constraint, all population size reach 0.1 % optimal after 50 generations

For relaxed constraint, population size 75 and 100 reach 0.1 % Optimal after 100

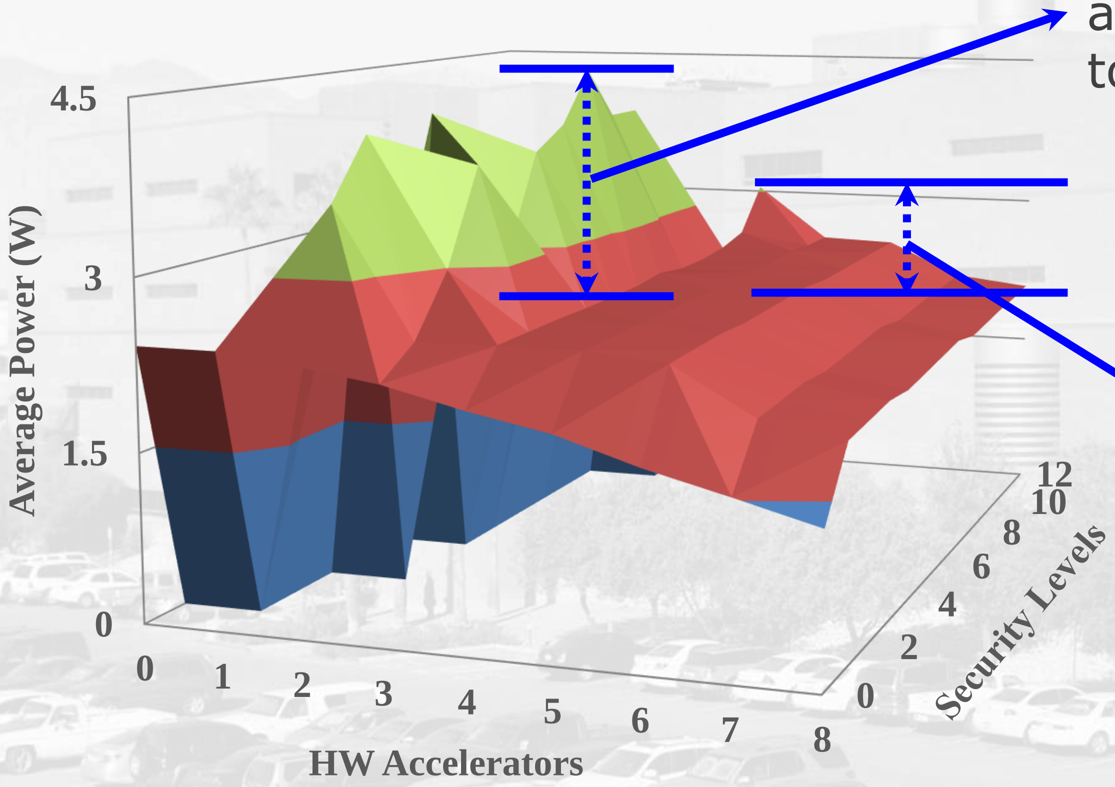
Configured genetic optimization algorithm to use population size of 75 and generations of 100

Experimental Results - Power vs. Security Level



- Each doubling key size (with same rounds) increases power by average of 0.2% (e.g., SL of 3 to SL 8)
- Each increase in number of round (with same key size) increases power by average of 1.6% (e.g., SL of 8 to SL 12)

Experimental Results – Hardware/Security/Power Tradeoffs

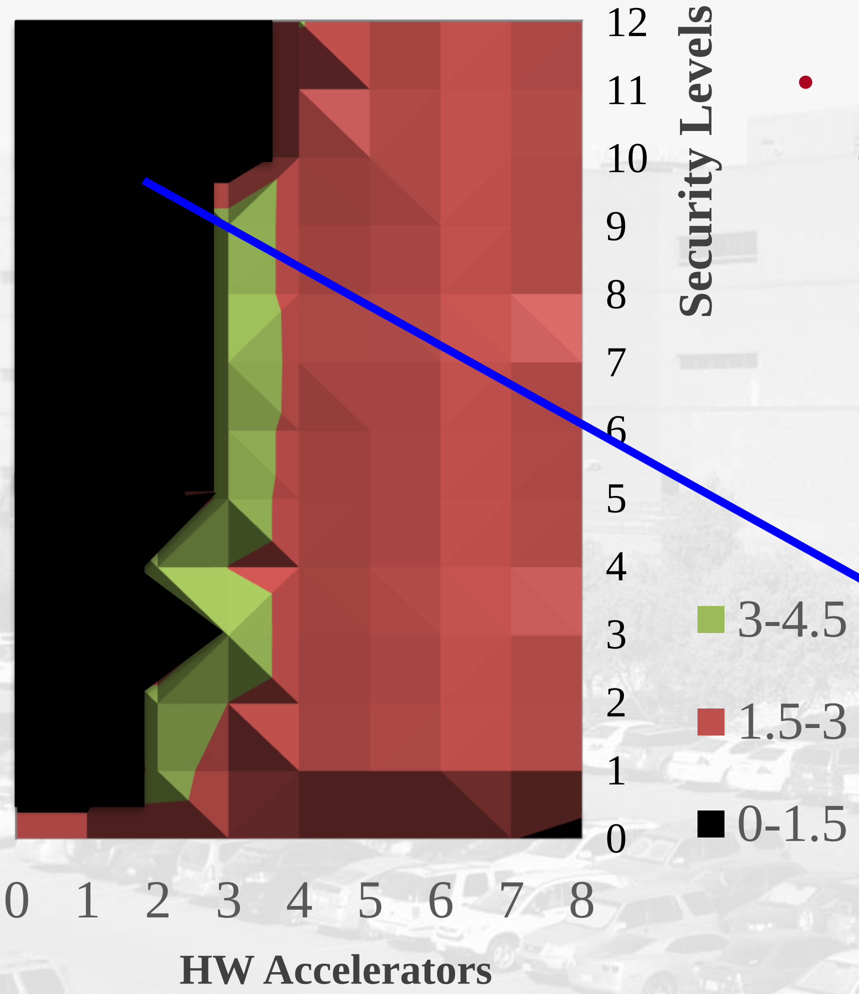


For SL 9, increasing accelerators in FPGA from 3 to 4 reduces power by 38%

For SL 12, increasing accelerators in FPGA from 5 to 8 reduces the power by 26%

Increasing number of hardware accelerator results in lower power

Experimental Results – Hardware/Security/Power Tradeoffs



- Numerous security level and hardware accelerator constraints lead to infeasible implementations
  - With 0 hardware accelerators, no security (i.e., SL 0) can be utilized

Dark region indicates infeasible constraints

Conclusions and Future Works

- **Conclusions**
  - Application modeling and optimization framework for dataflow applications
  - Different cryptographic configurations to achieve different security levels
  - Evaluated the security, hardware, and power tradeoffs, demonstrating the power reductions that can be achieved using reconfigurable hardware and in some cases using a higher security level
- **Future works**
  - Utilizing multi-objective optimization metrics
  - Integrating dynamic profiling and system observation methods to monitor system execution and detect deviations
  - Explore the effectiveness of the proposed approach both for different applications and different heterogeneous system architectures

Thank you

Questions?