

Towards an Artificial DNA for the Use in Dynamic Environments

Mathias Pacher and Uwe Brinkschulte

ISORC 2019, Valencia

May 9, 2019

1. Motivation

Current ICT systems:

- Increasingly complex
- Distributed
- Interconnected
- Dynamic environments

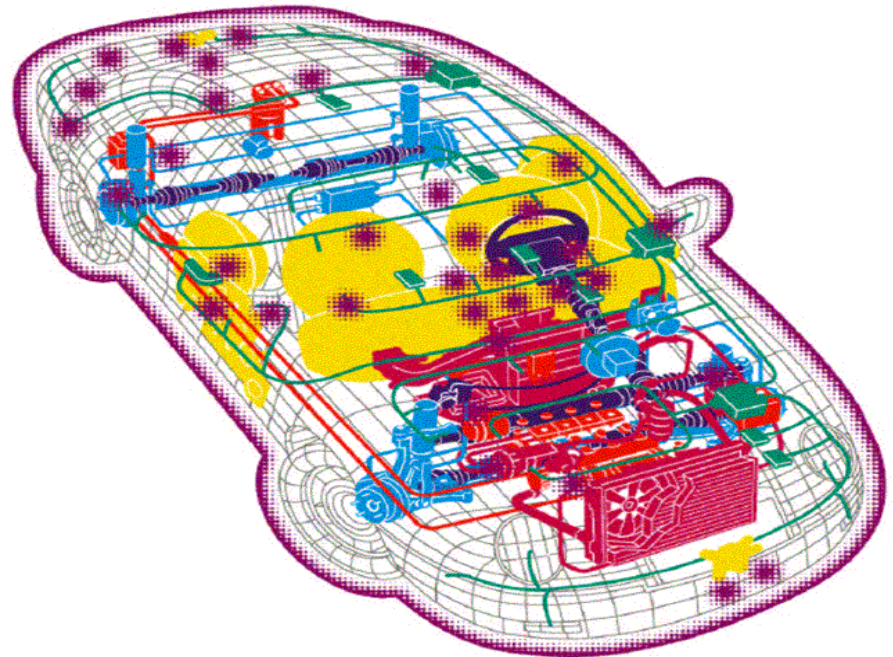
→ Thus,

- Development and maintenance are hard
- Failures at run-time

Idee of *Organic Computing*:

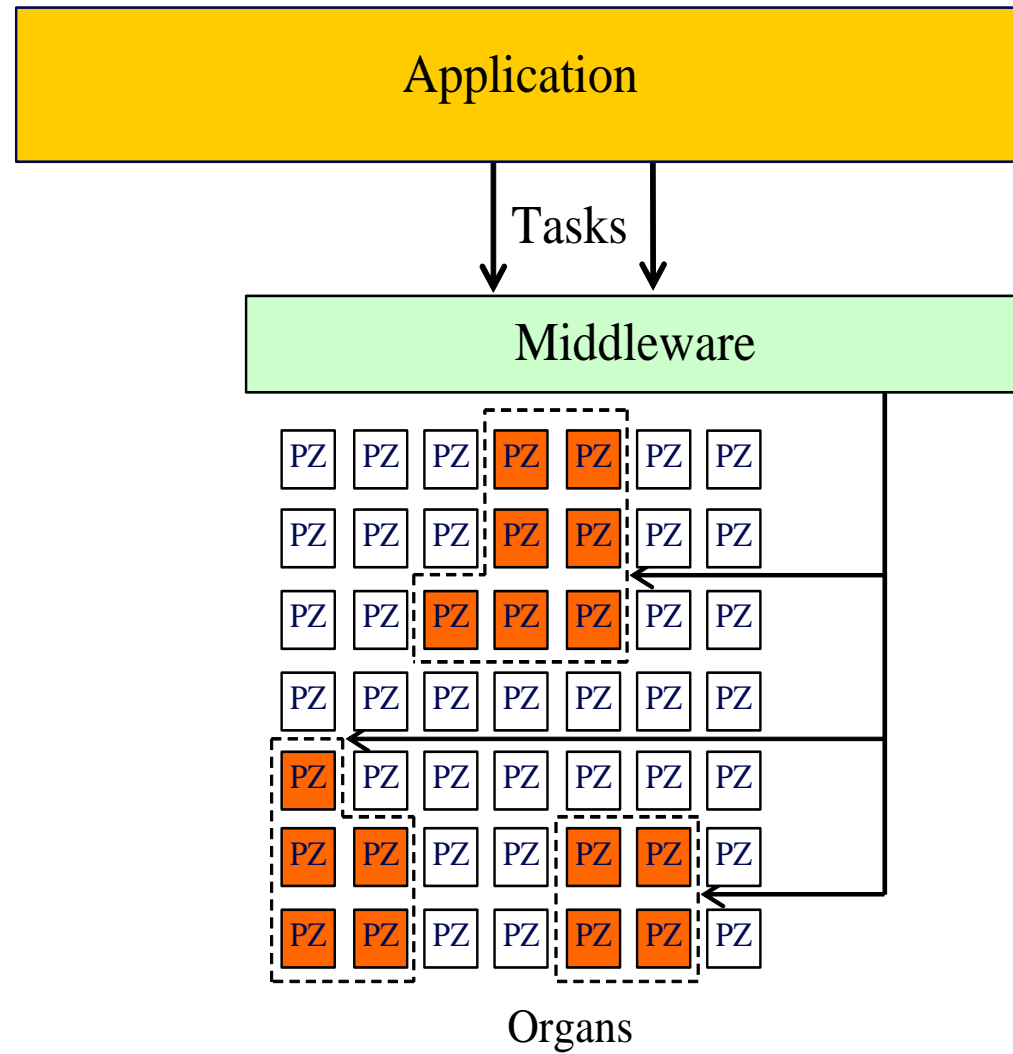
- System adapts autonomously and dynamically to environment

(Tomforde et al., „Organic Computing in the Spotlight“, 2017)



2. Artificial Hormone System

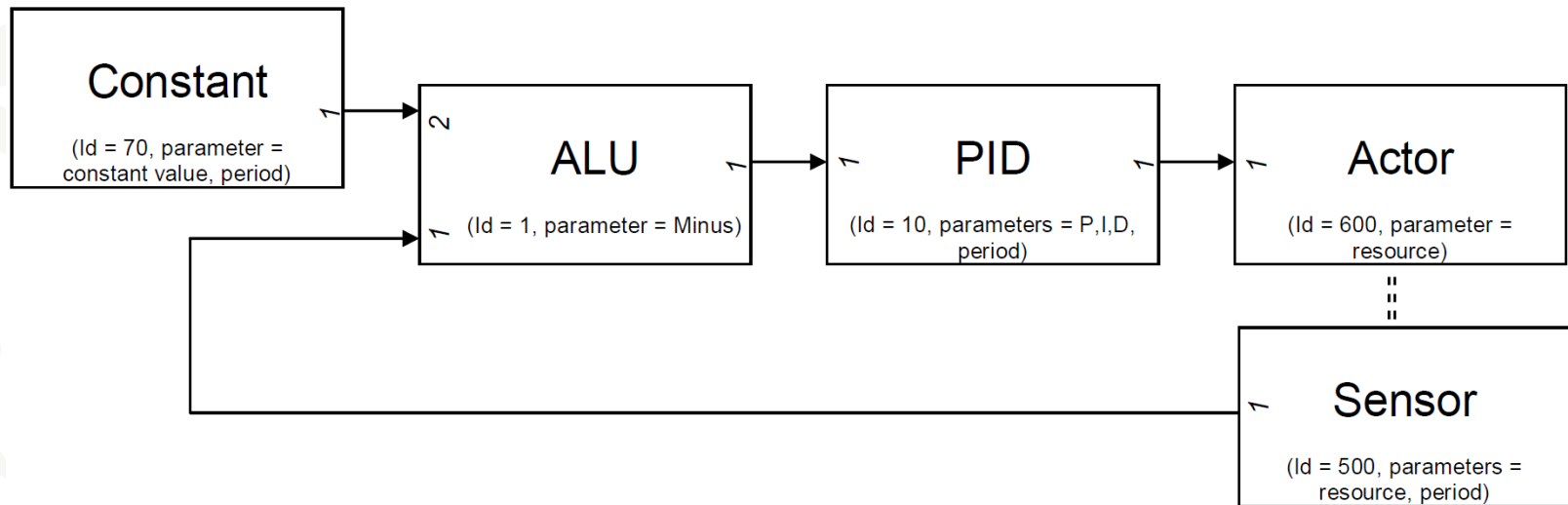
- Assignment of tasks to processors
- Hormone-based control loops
 - Self-configuration
 - Self-improvement
 - Self-healing
- Create virtual organs



3. Artificial DNA

Idea:

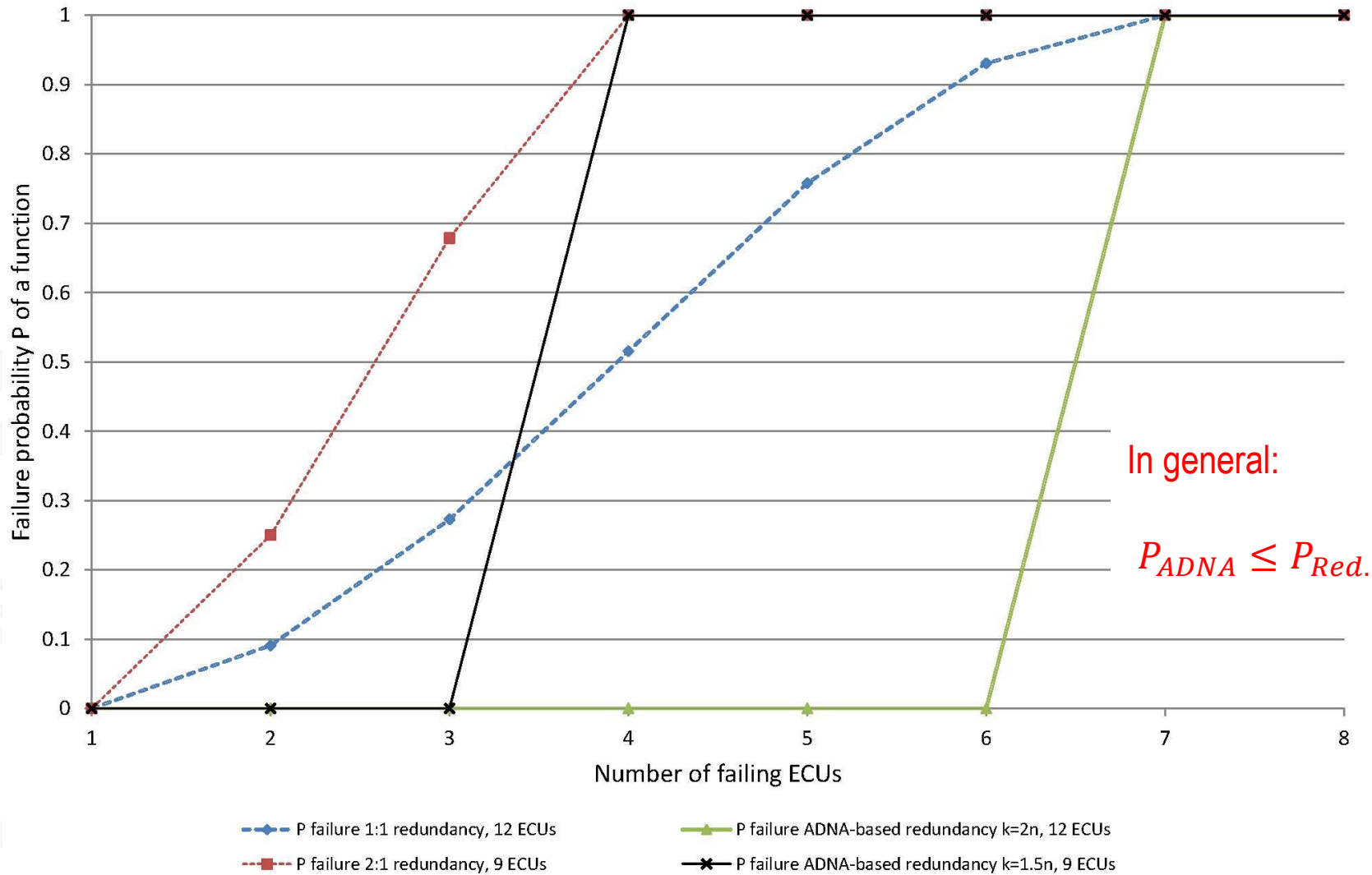
- Most embedded systems consist of standard components
- ➔ Describe components and interconnection as a text file ➔ **Artificial DNA**
- ➔ No programming, only parametrization
- ➔ Automatically determine tasks and hormone strength



```

1 = 70 (1:2.2) 100 25 // constant setpoint value, period 25 msec
2 = 1 (1:3.1) -        // ALU, control deviation (minus)
3 = 10 (1:4.1) 4 5 6 25 // PID (4, 5, 6), period 25 msec
4 = 600 1              // actor, resource id = 1
5 = 500 (1:2.1) 2 25   // sensor, resource id = 2, period 25 msec
  
```

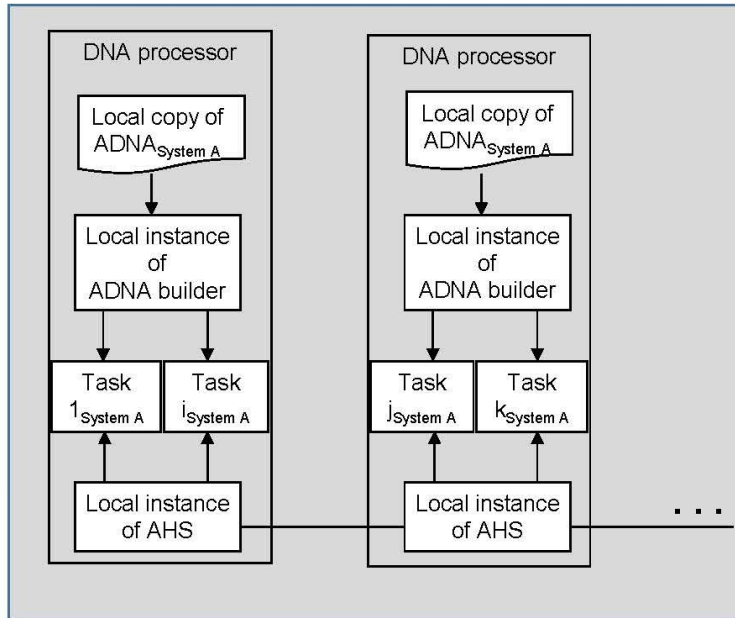
4. Dependability



➔ Interesting approach for automotive applications

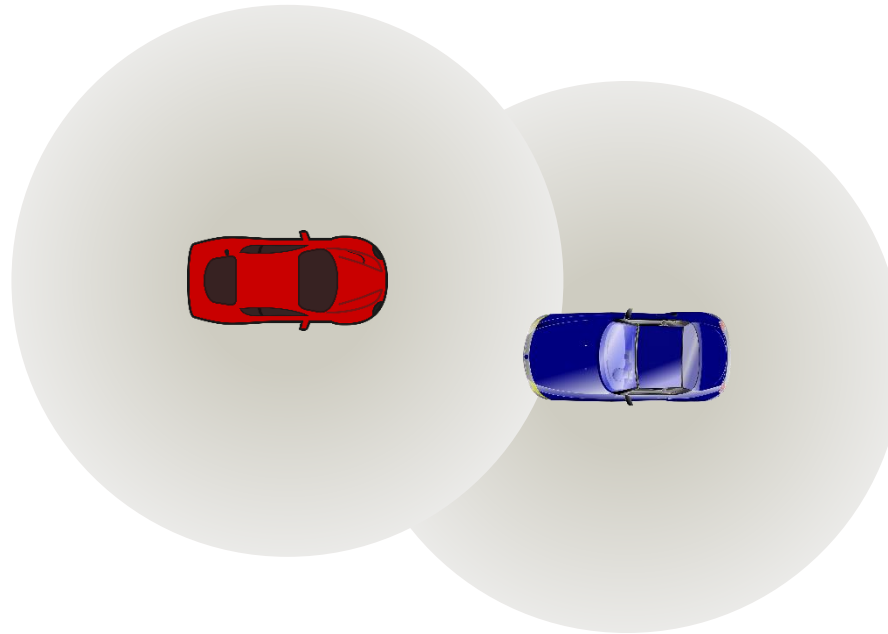
5. Artificial DNA for dynamic environments

System A



- Self-building system at run-time
- Easy to configure at run-time
- Scalable
- ADNAs of different systems may merge and separate

Example

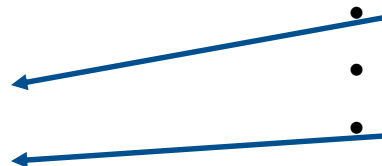


Red car:

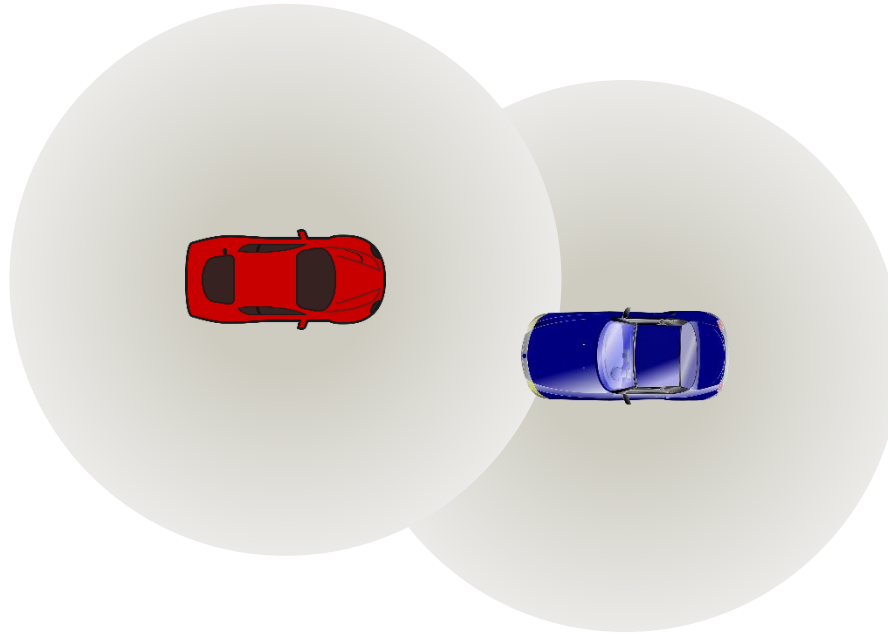
- ABS 1
- ESP 1
- Motor control 1
- Entertainment 1
- ...

Blue car (less computing power):

- ABS 2
- ESP 2
- Motor control 2
- Entertainment 2
- ...



Example

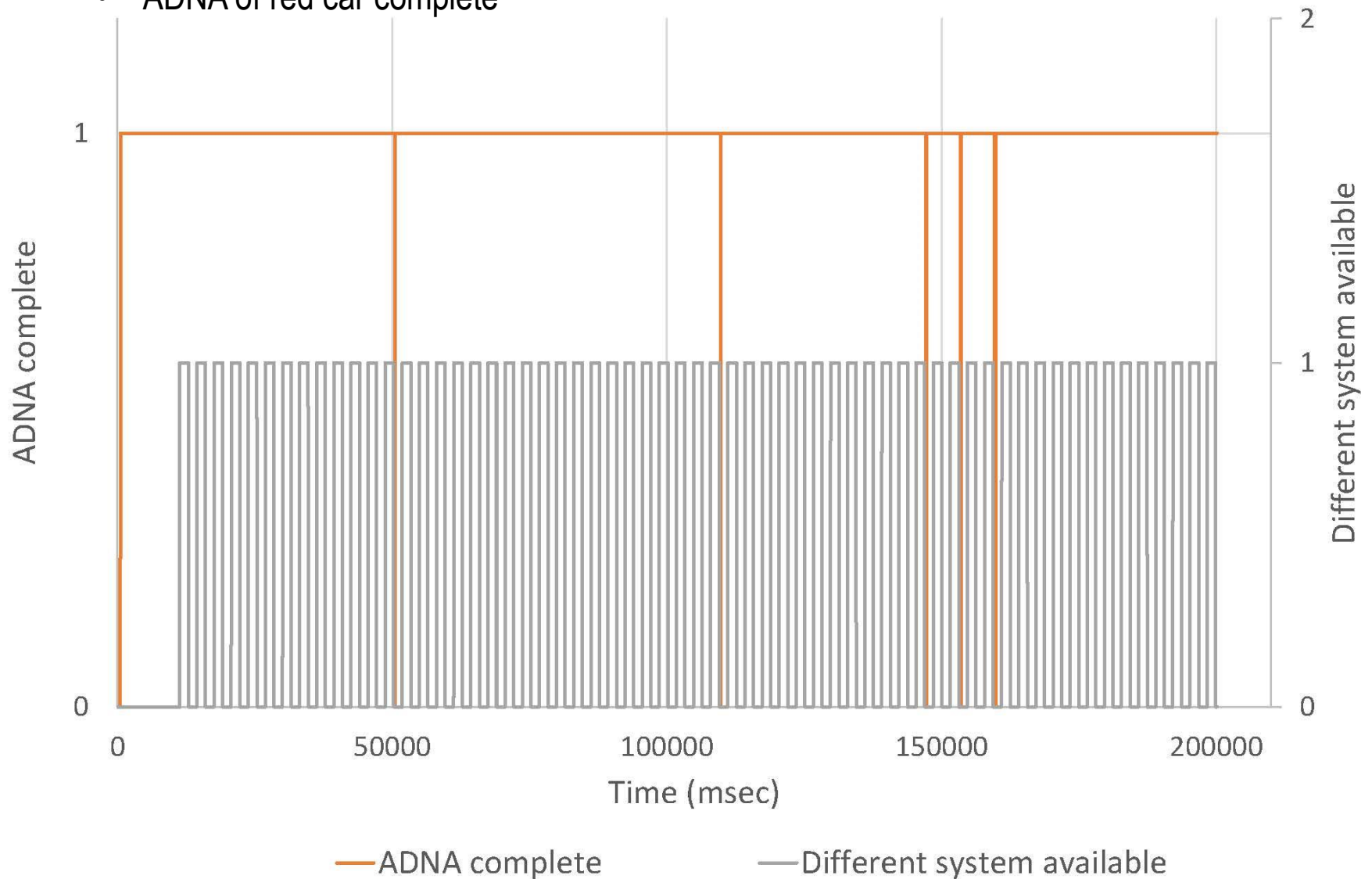


Scenarios:

1. Stress test → Different car is in range each 1.5 seconds for 1.5 seconds
2. Replacement for failing processors → Different car is in range each 6 seconds for 4.5 seconds

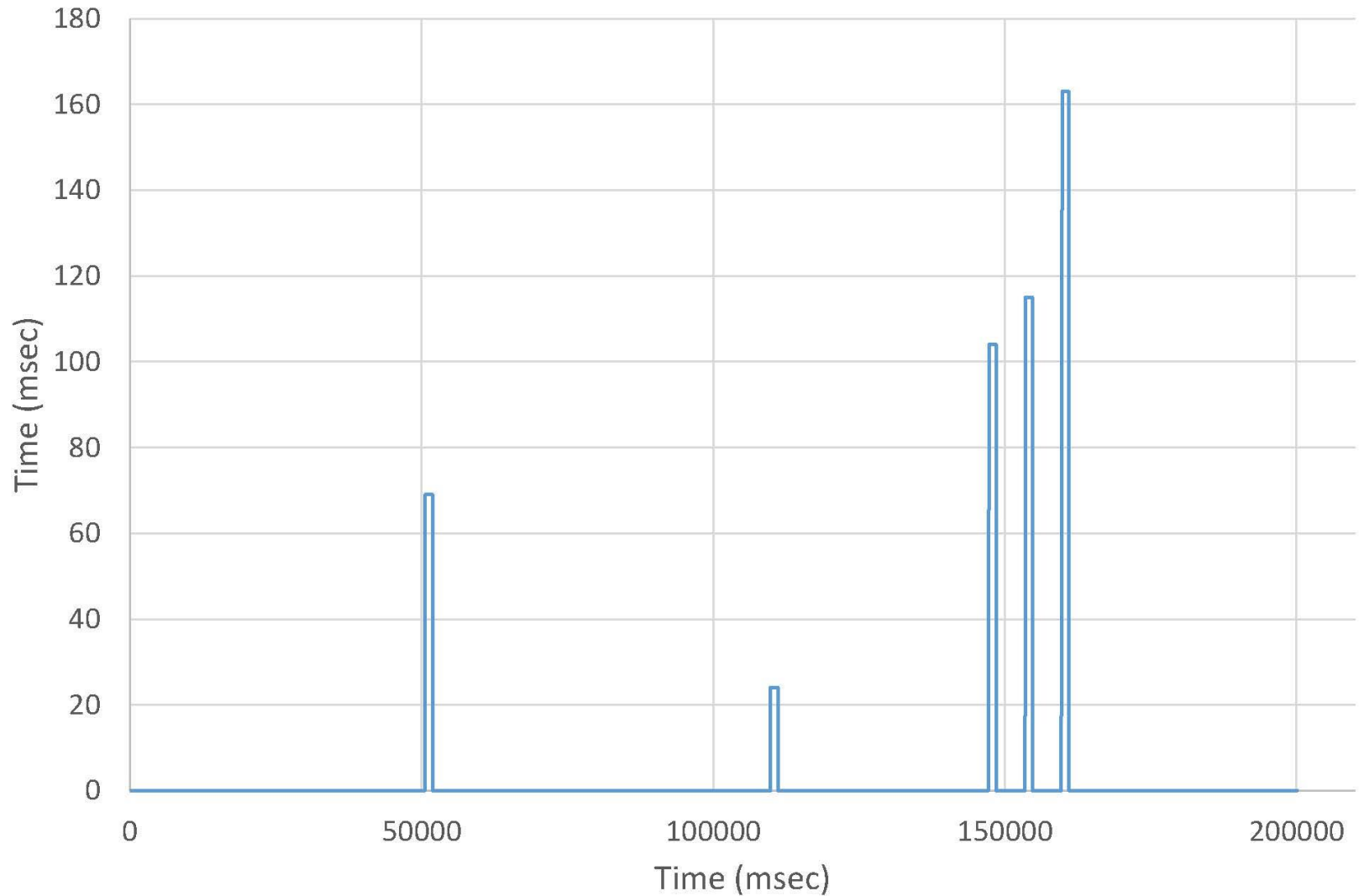
Evaluation scenario 1 (stress test)

- ADNA of red car complete



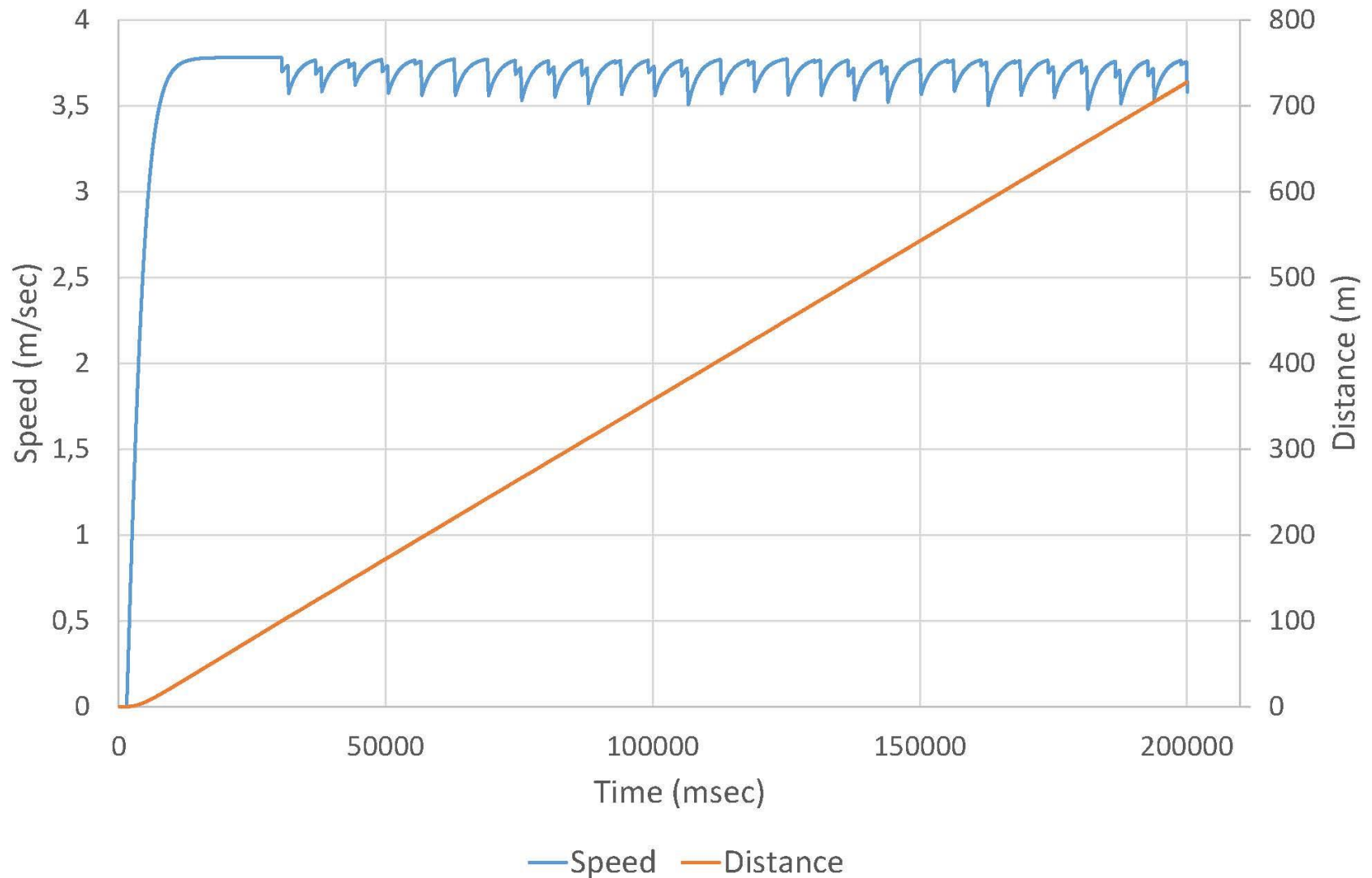
Evaluation scenario 1 (stress test)

- Reaction time until ADNA is complete



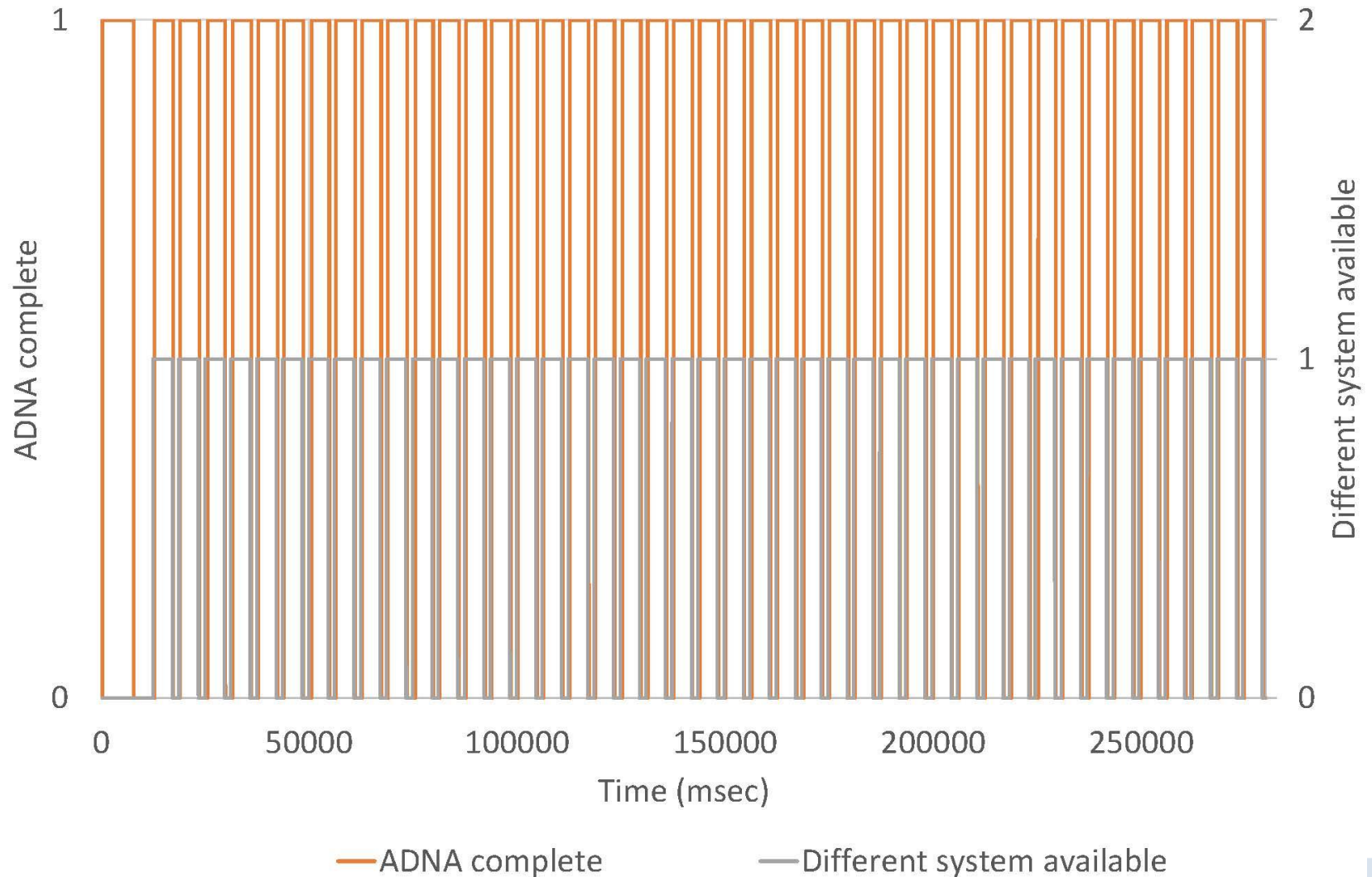
Evaluation scenario 1 (stress test)

- Speed and distance



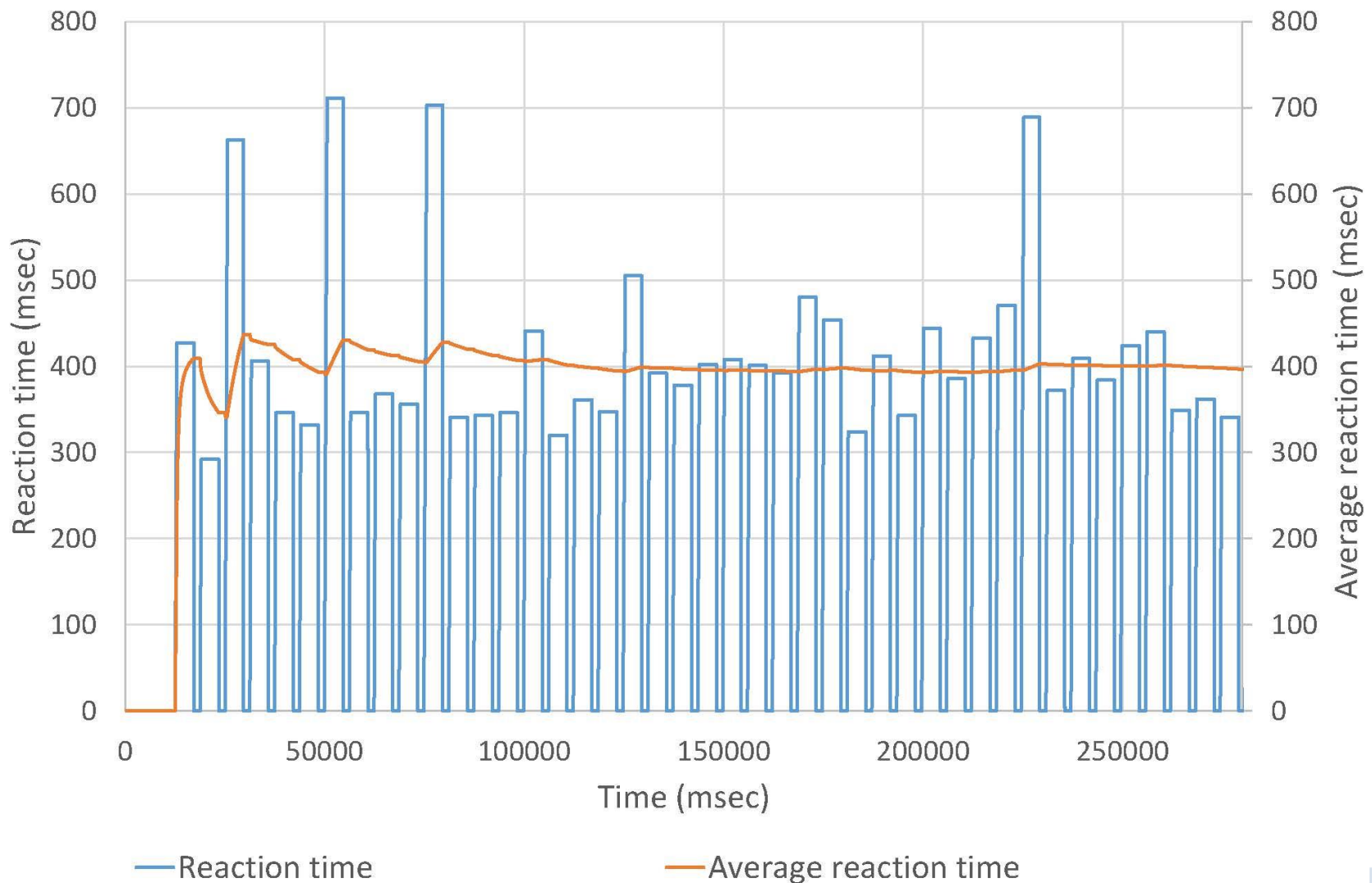
Evaluation scenario 2 (Replacement for failing processors)

- ADNA of red car complete



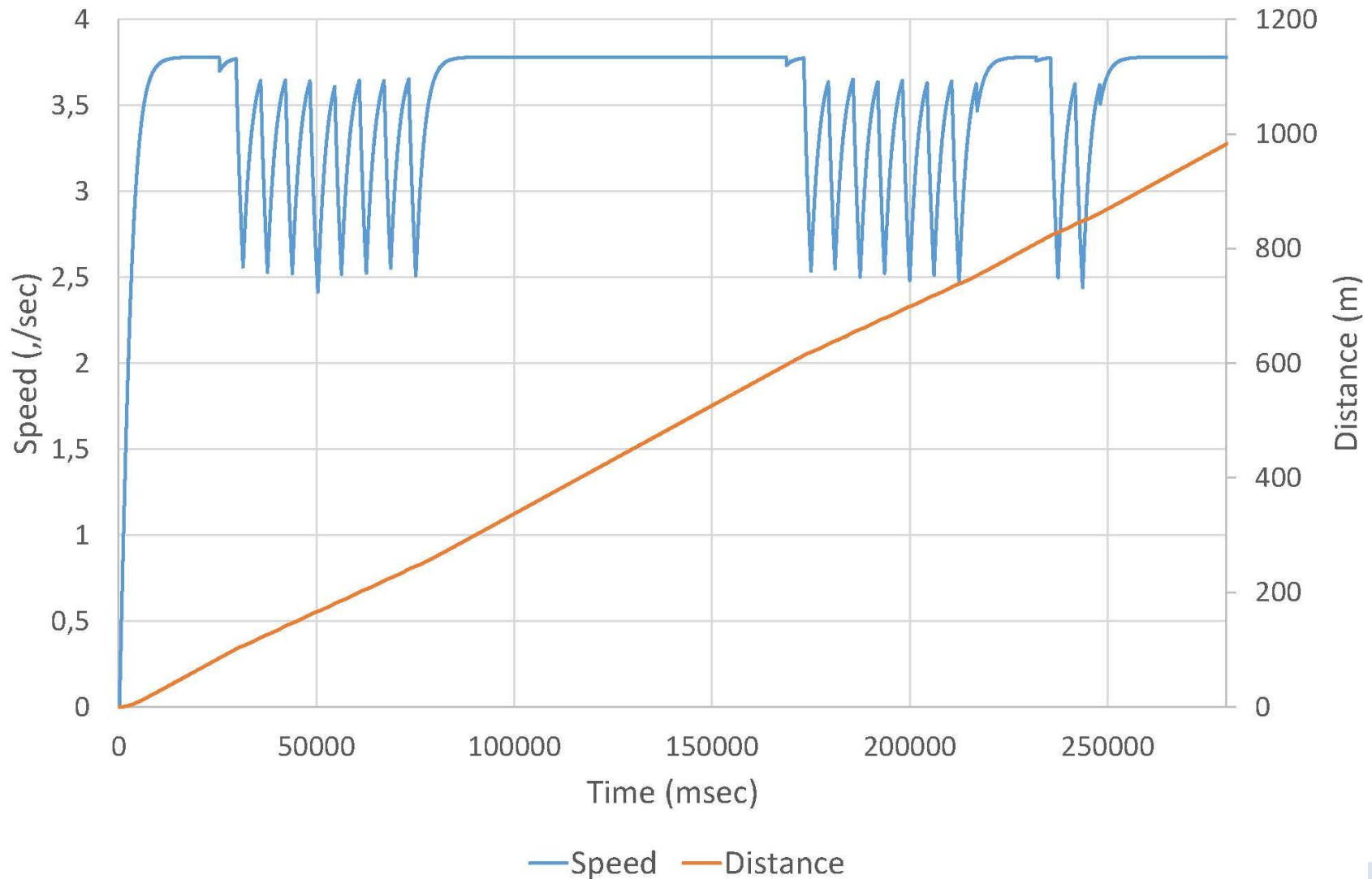
Evaluation scenario 2 (Replacement for failing processors)

- Reaction time until ADNA is complete



Evaluation scenario 2 (Replacement for failing processors)

- Speed and distance

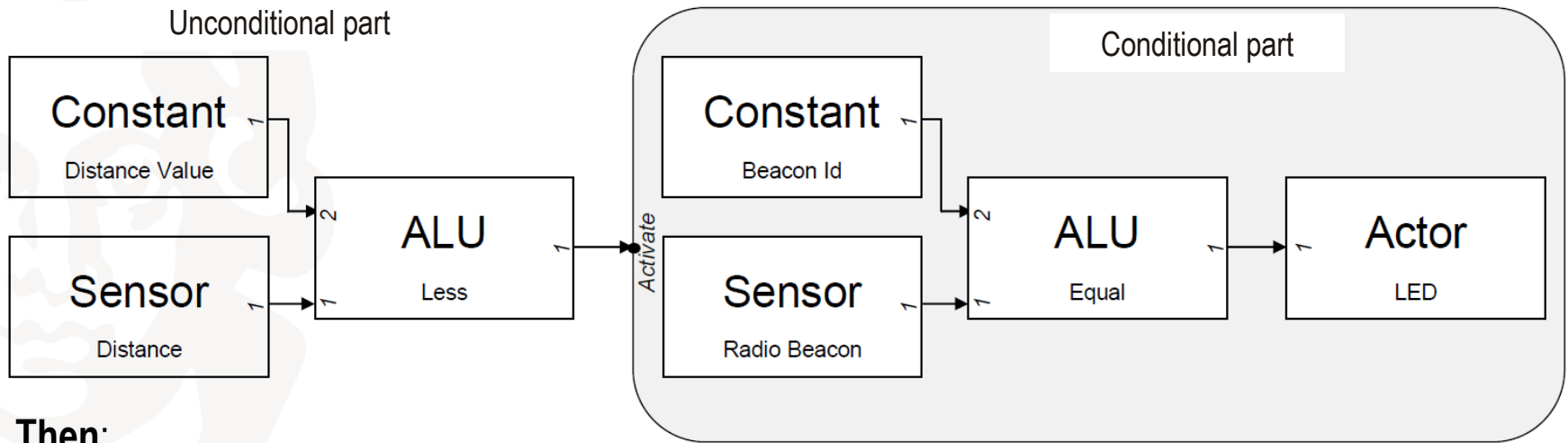


Conclusion

- First experiments with extended ADNA for dynamic systems
 - Stress test
 - Compensate failing processors

Future work:

- ADNA assignment priorities
- Conditional ADNA



Then:

- Paywall for automotive applications

Thank you

Questions?

