BSSE System and Software Engineering

Generating Random Telecommand Test Data Using Genetic Algorithms

Ralf Gerlich, Rainer Gerlich

Data Systems in Aerospace DASIA 2018

May 29th-31st, 2018, Oxford, UK

Dr. Rainer Gerlich System and Software Engineering BSSE
Immenstaad, Germany
E-Mail: Rainer.Gerlich@bsse.biz
Ralf.Gerlich@bsse.biz

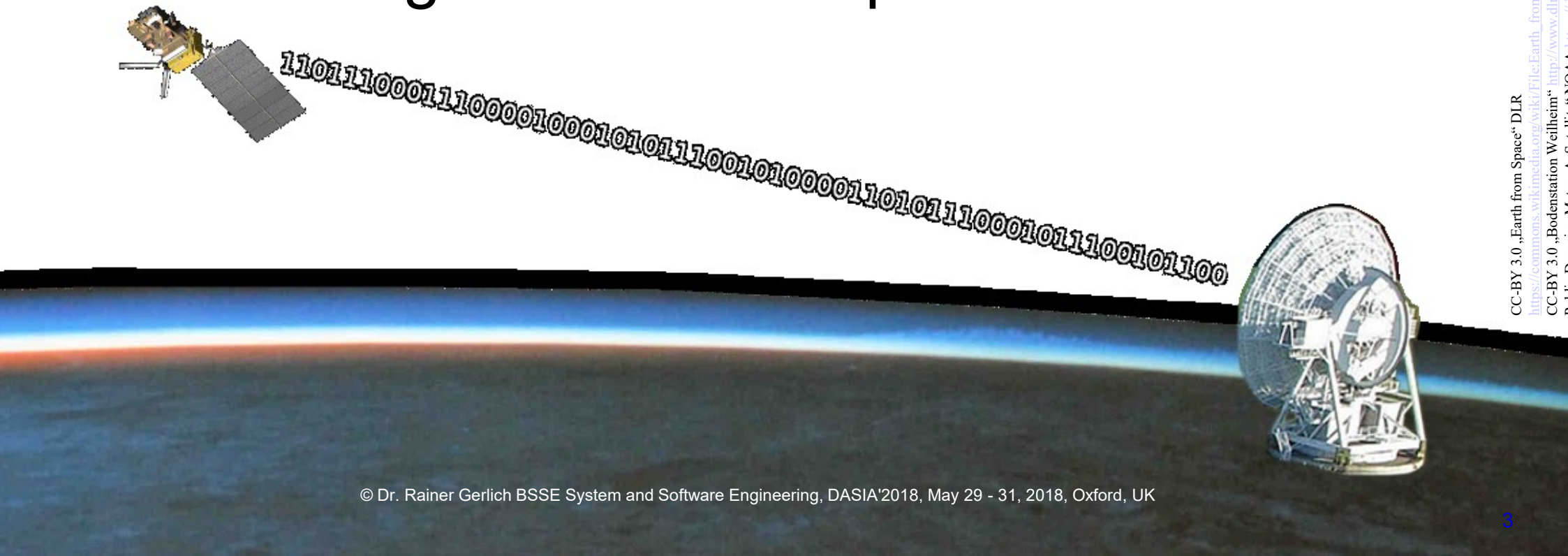
This work is supported by a grant from the
German Federal Ministry for Economic Affairs and Energy,
based on a decision of the German Bundestag,
grant No. 50PS1601.

Contents

- Introduction to Genetic Algorithms
- Chosen Approach
- Evaluation
- Conclusions and Outlook

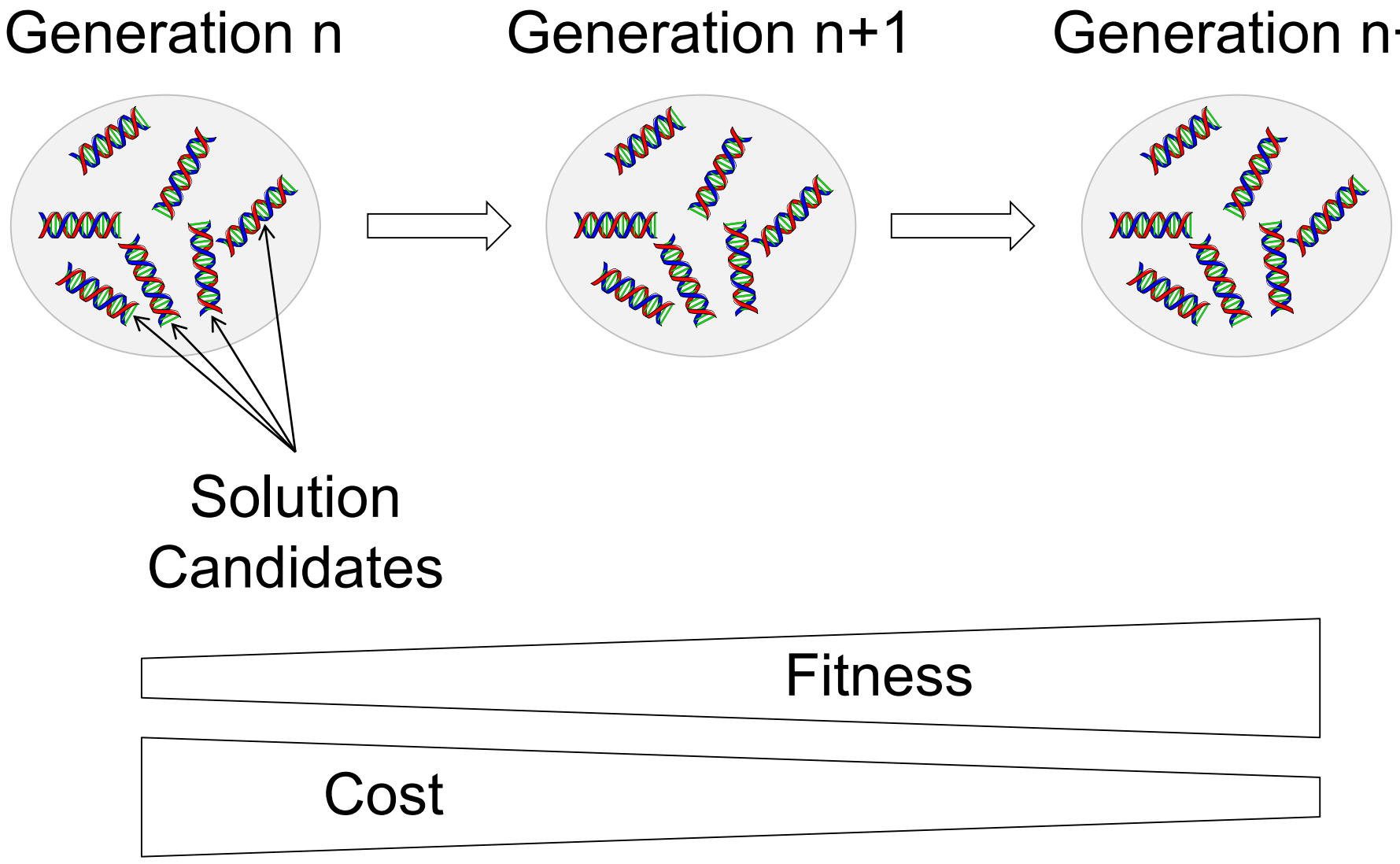
Motivation

- Spacecraft processes Telecommands from Ground Station, rejects invalid commands
- Commands arrive as untyped bytestreams
- How to generate test inputs?

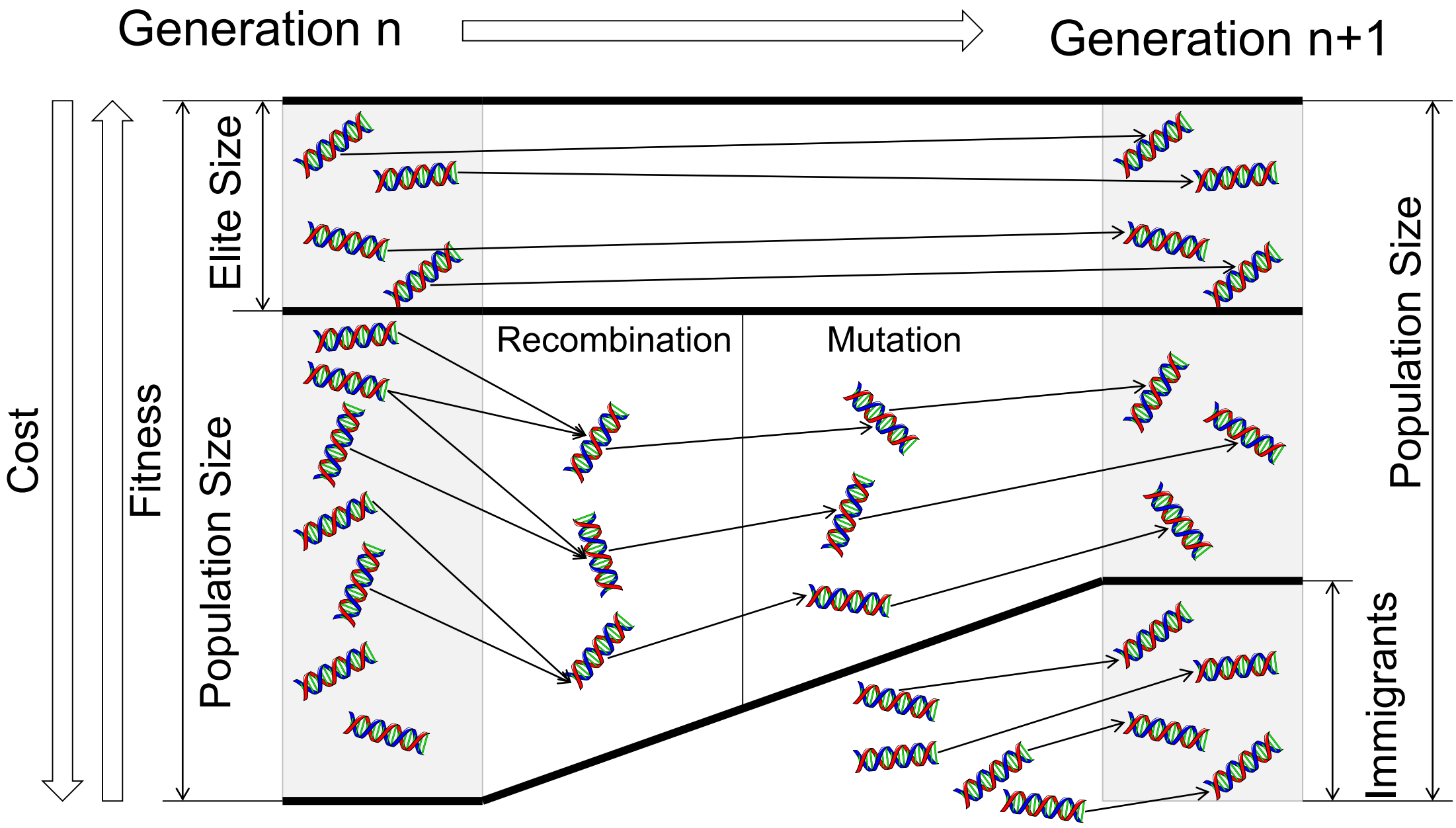


Introduction to Genetic Algorithms

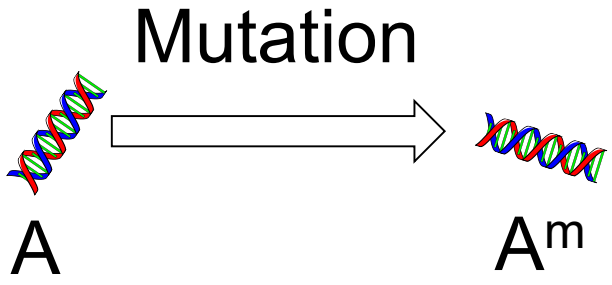
Evolution

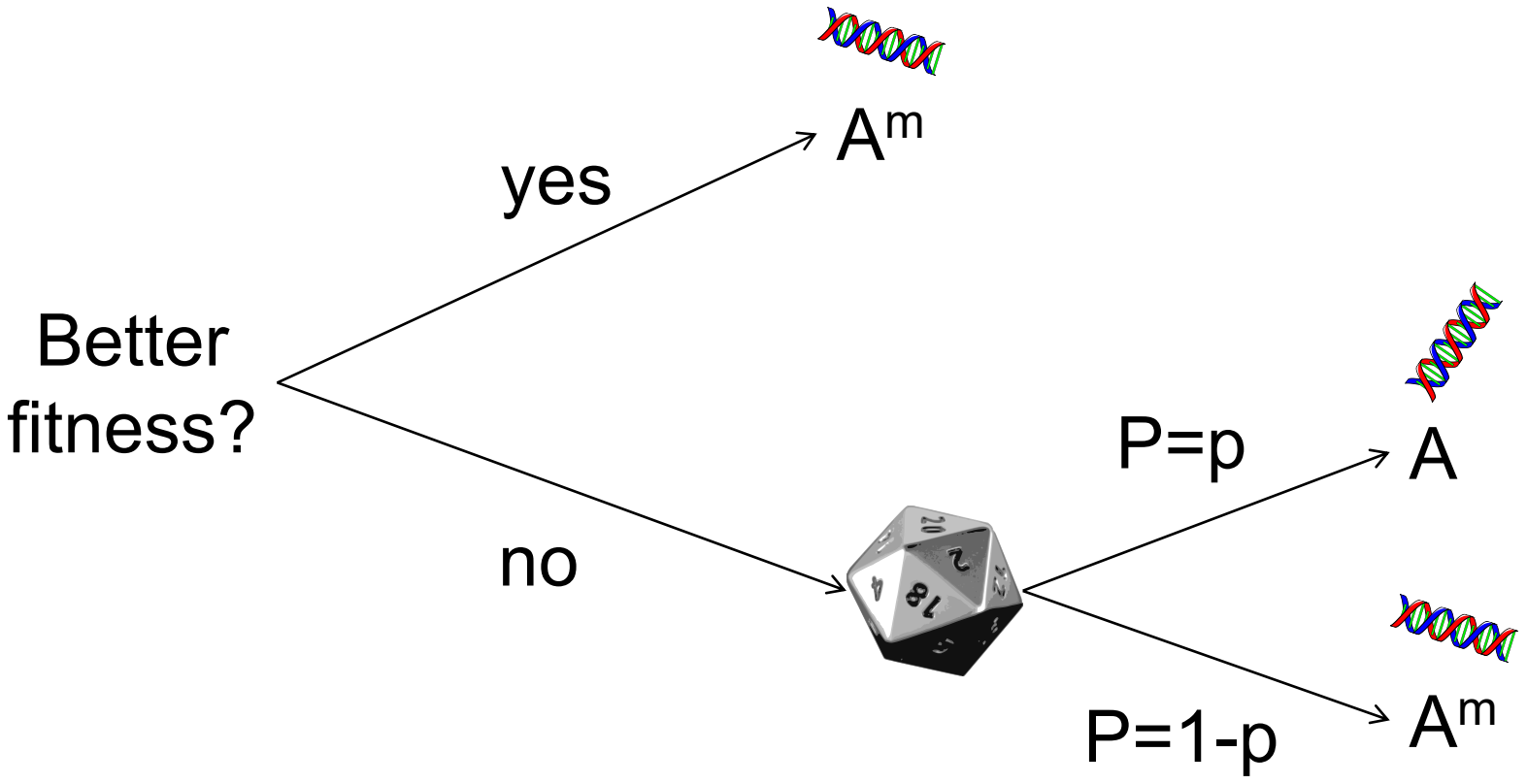

Generation n
Generation n+1
Generation n+2
Solution Candidates
Fitness
Cost

Progression of Generations



Probabilistic Mutation Reversal



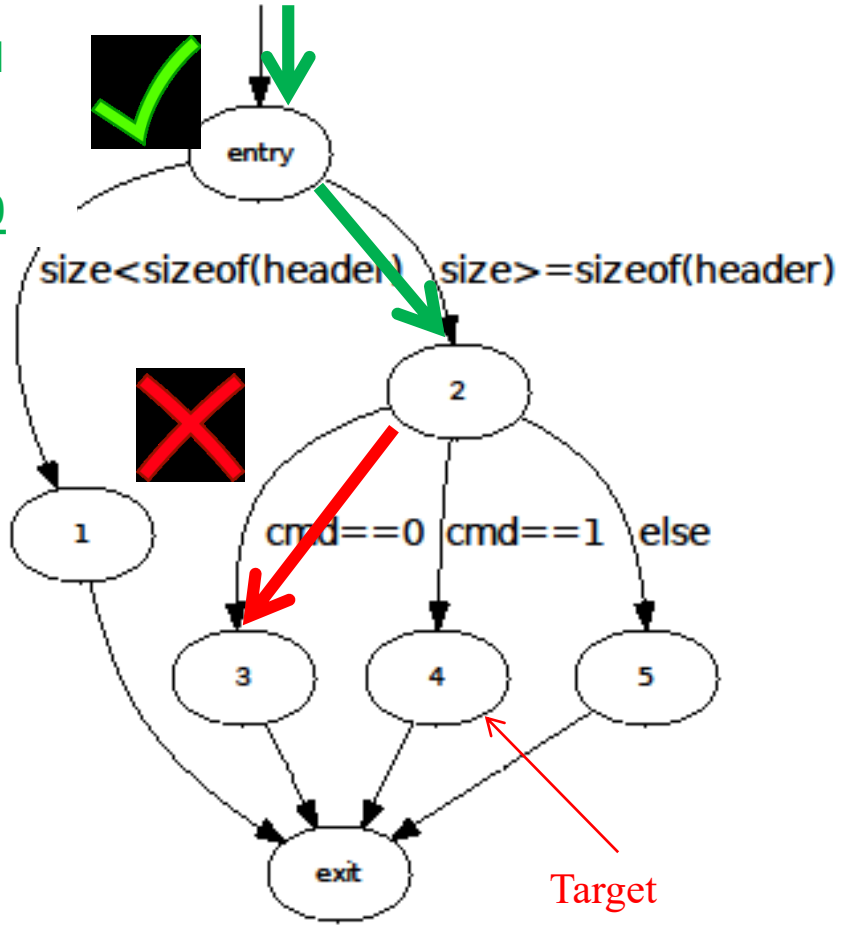


Chosen Approach

Cost-Function

Can still reach target
Cost = 0

Cannot reach target anymore
Cost > 0

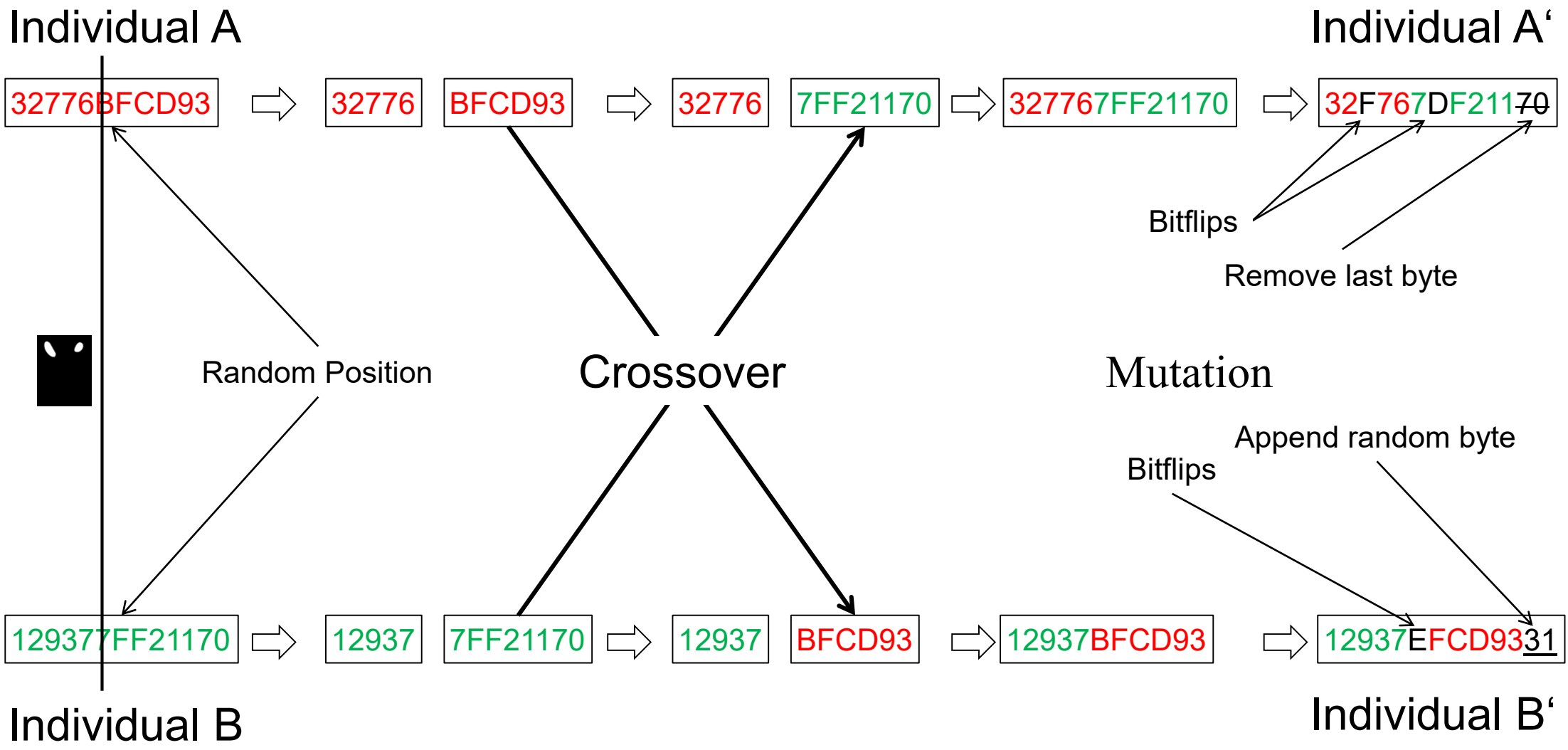


Cost Function describes how far individual is off from reaching target

Desired Condition	Cost Function
E op F (op ∈ {=,<,>,≤,≥})	$\begin{cases} 0 & \text{if } E \text{ op } F \\ \lvert F - E \rvert + 1 & \text{otherwise} \end{cases}$
E != F	$\begin{cases} 1 & \text{if } E = F \\ 0 & \text{otherwise} \end{cases}$

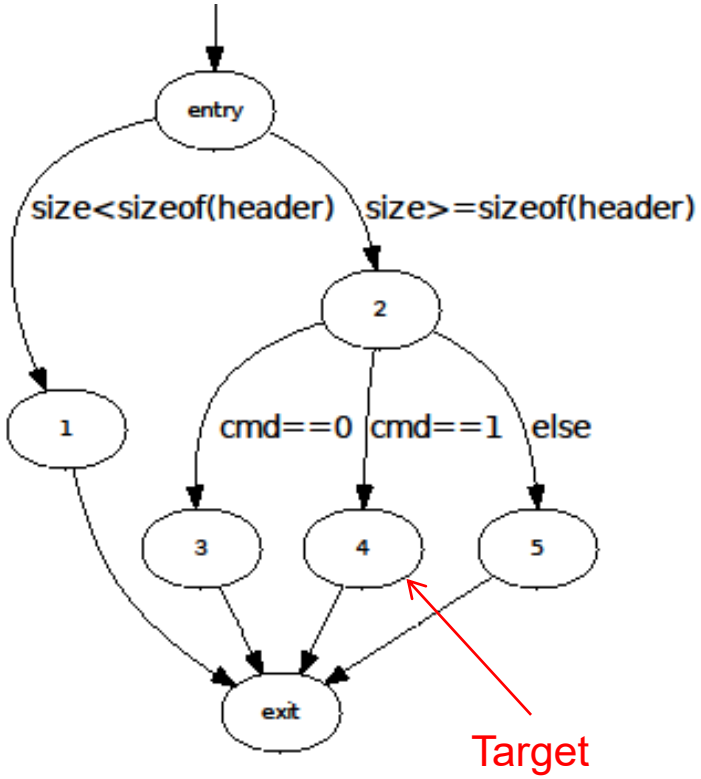
Singularity: No gradient!
Issue for boolean variables as well!

Crossover and Mutation

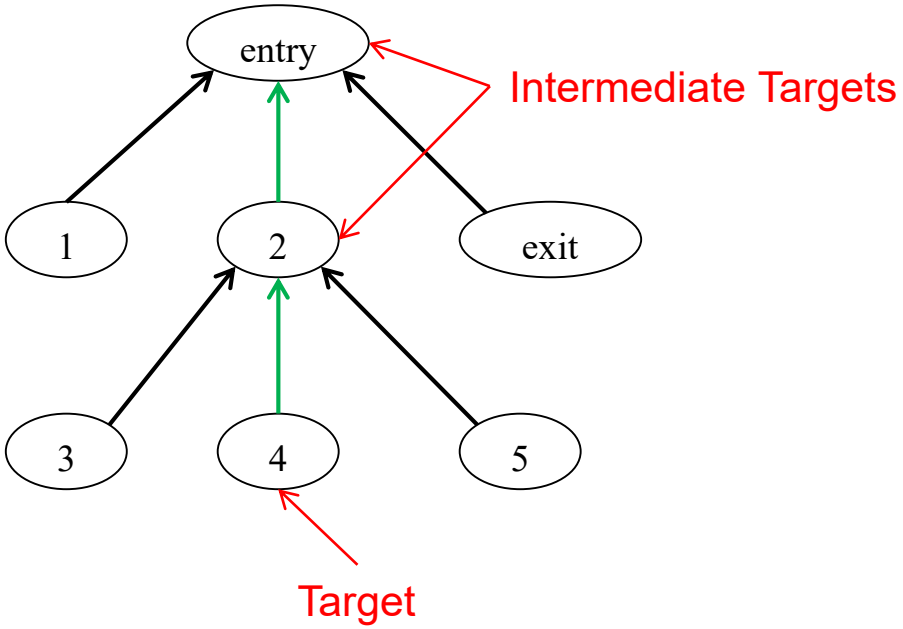


Sequential Approach: Intermediate Target

Control-Flow-Graph



Dominator Graph



- Single Step Approach: Use final target only
- Sequential Approach: Use intermediate targets

Evaluation

Preliminary Evaluation

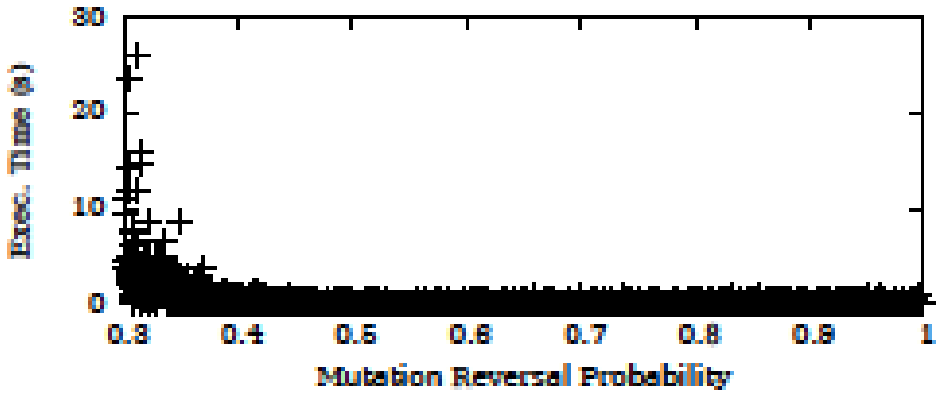
- Analyse Feasibility
- Compare runtime of algorithm variants
- Determine impact of parameters on runtime
  - Population Size
  - Elite Proportion
  - Immigrant Proportion
  - Mutation Probabilities
  - Mutation Reversal Probability
- So far only on simple example

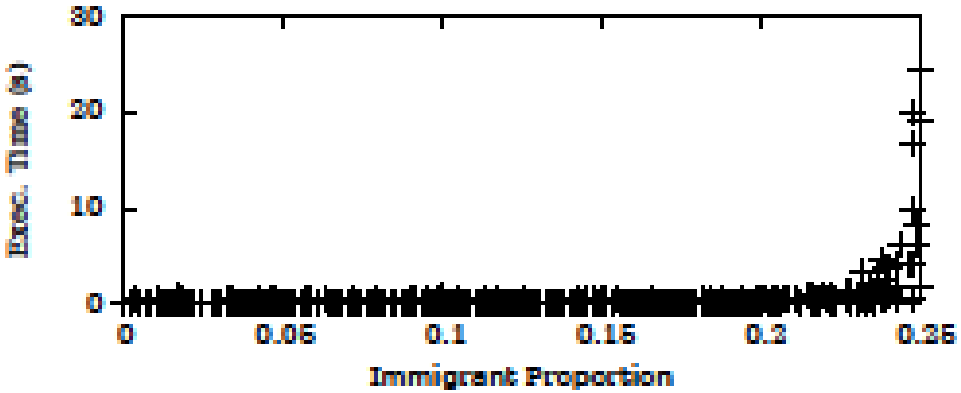
Runtime Measurements: Algorithm Variants

Variant	Min (s)	Mean (s)	Max (s)
Sequential	0.161	2.595	15.931
Single-Step	0.268	15.553	146.180

- Sequential is way faster than Single-Step
  - Mean runtime: Factor 6
  - Maximum runtime: Factor 9

Runtime Measurements: Parameters (1/2)





- Mutation Reversal has positive effect on runtime
- Does not seem to extend past $p \gtrsim 0.4$

- Immigrant Proportion has no noticeable effect for $p \lesssim 0.2$
- Negative effect for $p \gtrsim 0.2$

Runtime Measurements: Parameters (2/2)

BSSE System and Software Engineering

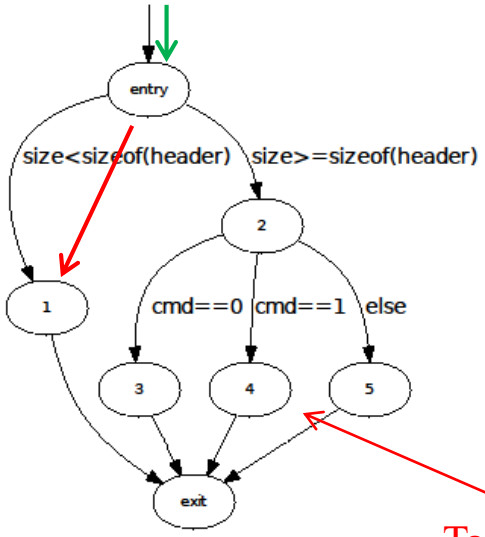
Byte Extension Prob.	0.00	0.25	0.50	0.75	1.00
Mean Execution Time (s)	0.46	0.13	0.12	0.12	0.13

- Optimum Value in the range $0.5 \lesssim p \lesssim 0.75$
- Decrease of Factor 3 from p=0 to p=0.25
- No further noticeable change for $p \rightarrow 1$

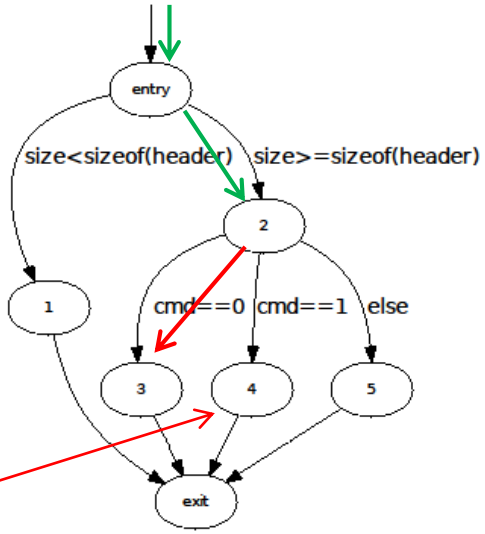
Conclusions and Outlook

Is Cross-Over Necessary?

Individual A



Individual B



Target

- How could A improve B by cross-over?
- Cost-Function does not consider branches not taken!

- Utility of Cross-Over is based on compositionality
- Test Candidates might not be compositional regarding reaching the target
- Multi-candidate gradient descent better?

Conclusions

- Sequential Approach is superior
- Significant positive impact of mutation reversal
- No positive effect from immigration, negative effect for more than 20% of population
- This genetic algorithm seems promising for generating telecommand test data
- Possibility for simplification: drop crossover

Future Work

- Integration with random test tool (DCRTT) is underway…
- Application to industry-grade code
- Comparison to random testing performance
- More detailed analysis of parameter impacts
  - Multivariate
  - Measurements on more realistic code

Important Question

Is there a general „good enough“ set of parameters for all applications?

Or do we need to „babysit“ the search?

Thank you for your attention!

Questions?

This work is supported by a grant from the German Federal Ministry for Economic Affairs and Energy, based on a decision of the German Bundestag, grant No. 50PS1601.

Backup