Workshop
Entwicklung zuverlässiger Softwaresysteme

Softwarezuverlässigkeitsbewertung auf Basis von Betriebsprofiltreue und Schnittstellenüberdeckung

Matthias Meitner, Francesca Saglietti
Universität Erlangen-Nürnberg
Lehrstuhl für Software Engineering (Informatik 11)

Motivation

- Ziel:
  - Softwarezuverlässigkeitsbewertung für komponentenbasierte Systeme (in Zusammenarbeit mit Siemens Corporate Technology)
- Zuverlässigkeitsbewertung mit Hilfe der statistischen Stichprobentheorie
  - Black-Box Testen
  - Berücksichtigt nicht die interne Struktur eines Programms
- Integrationstest
  - Kopplungsbasierte Überdeckungskriterien
  - Ableitung eines Zuverlässigkeitsaussage nicht möglich
- Kombination der beiden Testperspektiven, so dass
  - Einerseits das Betriebsprofil berücksichtigt
  - Und andererseits hohe Interaktionsüberdeckung garantiert wird

Statistische Stichprobentheorie

- Anforderungen an Testdurchführung
  - Unabhängige Ausführung der Testfälle (z.B. durch reset-Mechanismen)
  - Jedes Versagen während des Testens wird entdeckt
  - Keine oder wenige Versagen werden beobachtet
- Anforderungen an Testdaten
  - Betriebsprofiltreue
  - Unabhängige Auswahl der Testfälle

3 Optimierungsziele

- Ziel 1: Betriebsprofiltreue
  - K.O.-Kriterium
  - Goodness-of-Fit-Tests
- Ziel 2: Unabhängige Auswahl
  - K.O.-Kriterium
  - Korrelationsmetriken
- Ziel 3: Hohe Interaktionsüberdeckung
  - Maximierung des Kriteriums
  - Kopplungsbasierte Überdeckungskriterien (prozedural / OO)

Statistische Stichprobentheorie

$$P\left(p \leq \tilde{p}\right) = \beta$$

Kein Versagen: $$\tilde{p} = 1 - \sqrt[n]{1-\beta}$$

Safety Integrity Levels Low Demand gemäß IEC 61508

SIL	β	$\tilde{p}$	n
1	0.99	$\geq 10^{-2}$ to $< 10^{-1}$	459
2	0.99	$\geq 10^{-3}$ to $< 10^{-2}$	4603
3	0.99	$\geq 10^{-4}$ to $< 10^{-3}$	46050
4	0.99	$\geq 10^{-5}$ to $< 10^{-4}$	460515

Korrelationen

- Abhängig von der Anwendung können Eingabeparameter semantisch korreliert sein
  - Durch physikalische Gesetze
  - Durch logische Muster
- Diese Abhängigkeiten
  - Werden durch das Betriebsprofil erfasst und
  - Können nicht modifiziert werden
- Abhängigkeiten, die sich durch die Instantiierung der Parameter ergeben, müssen hingegen durch Filter entfernt werden

Ziel 1: Betriebsprofiltreue

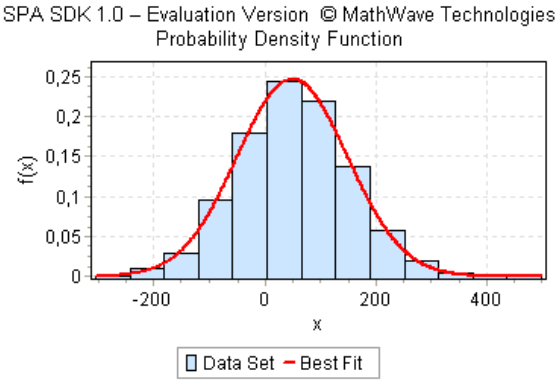
- Betriebsprofil wird auf Basis der Auftrittshäufigkeit der Eingabeparameter bestimmt
- Unabhängige Parameter können mit Hilfe von Zufallszahlengeneratoren erzeugt werden
- Abhängige Parameter werden gemäß ihrer funktionalen Abhängigkeiten instantiiert
- Goodness-of-Fit-Tests überprüfen, ob eine beobachtete Verteilung konsistent mit der im Betriebsprofil spezifizierten Verteilung ist
- Goodness-of-Fit-Tests, die in dieser Arbeit verwendet werden
  - χ^2 –Test
  - Kolmogorow-Smirnow-Test
  - Anderson-Darling-Test

Ziel 1: Betriebsprofiltreue

Beispiel: χ^2 -Test

χ^2– Statistik definiert als: $$\chi^2 = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}$$

- Nullhypothese (beobachtete Verteilung ist konsistent mit der erwarteten Verteilung)
  - Wird bei einem gegebenen Signifikanzniveau α verworfen, wenn der Wert der Teststatistik größer als der kritische Wert ist
  - Wird bei einem gegebenen Signifikanzniveau α akzeptiert, wenn der Wert der Teststatistik niedriger als der kritische Wert ist



Ziel 2: Unabhängige Testfallauswahl

- Arten der Korrelation
  - Autokorrelation
  - Kreuzkorrelation

- Kreuzkorrelationsmetrik: Pearsonscher Korrelationskoeffizient

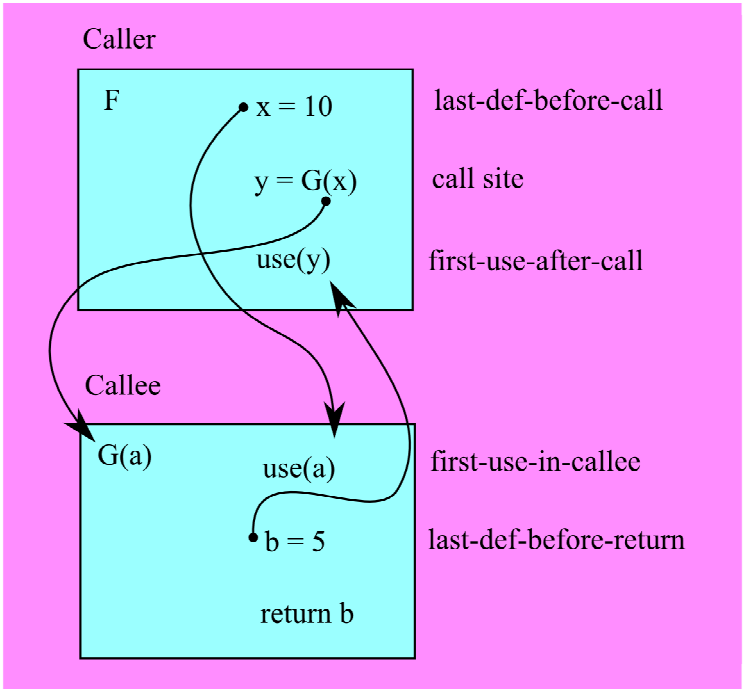
$$r_{xy} = \frac{\frac{1}{n}\sum_{i=1}^{n}\left(x_i - \bar{x}\right)\cdot\left(y_i - \bar{y}\right)}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(x_i - \bar{x}\right)^2}\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \bar{y}\right)^2}}$$

  - Misst nur lineare Abhängigkeiten
  - Datenmengen sollten normalverteilt sein
- Wegen dieser Einschränkungen wurden weitere Korrelationsmetriken verwendet, u.a.
  - Spearmans Rangkorrelationskoeffizient
  - Cramers V

Ziel 3: Interaktionsüberdeckung

Jin & Offutt

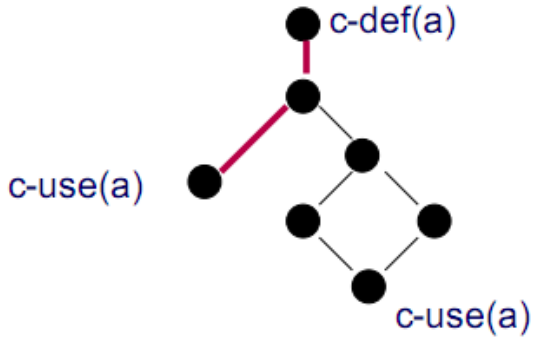
- Coupling-Def
  - Last-Def-Before-Call
  - Last-Def-Before-Return
  - Shared-Data-Def
- Coupling-Use
  - First-Use-In-Callee
  - First-Use-After-Call
  - Shared-Data-Use



- Aufrufstelle: Aufruf einer anderen Komponente
- Kopplungspfad: Definitionsfreier Pfad von einem coupling-def einer Variable zu einem coupling-use derselben Variable in einer anderen Komponente

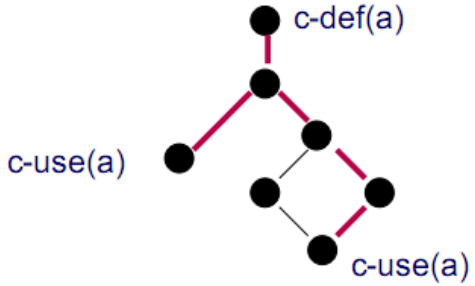
Ziel 3: Interaktionsüberdeckung

- Anwendung von Datenflusstesten auf Integrationstestebene
- Call coupling: alle Aufrufstellen müssen überdeckt werden
- All-coupling-defs: für jedes Coupling-Def einer Variable muss mindestens ein Kopplungspfad zu mindestens einem erreichbaren Coupling-Use überdeckt werden

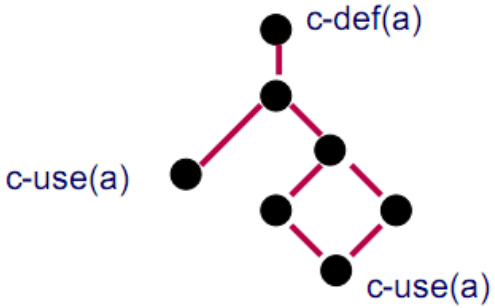


Ziel 3: Interaktionsüberdeckung

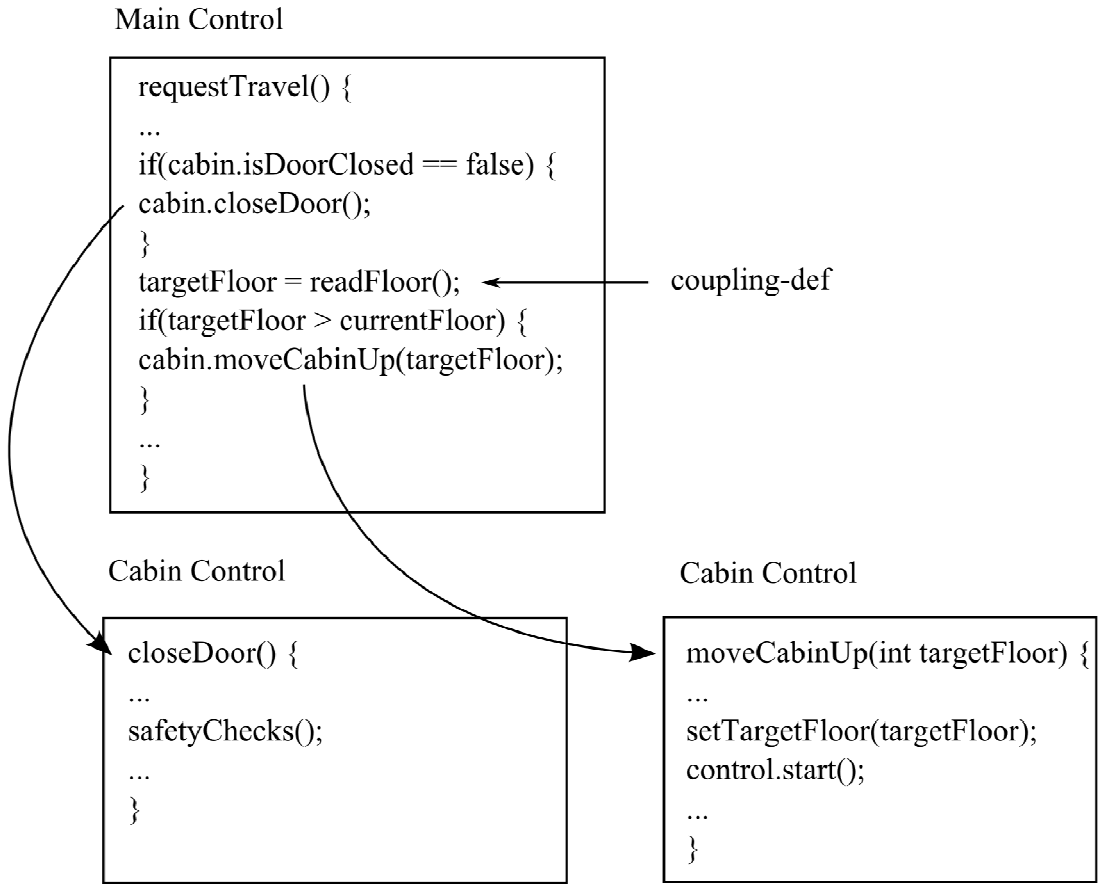
- **All-coupling-uses**: für jedes Coupling-Def einer Variable muss mindestens ein Kopplungspfad zu jedem erreichbaren Coupling-Use überdeckt werden



- **All-coupling-paths**: für jedes Coupling-Def müssen alle Kopplungspfade zu jedem erreichbaren Coupling-Use überdeckt werden



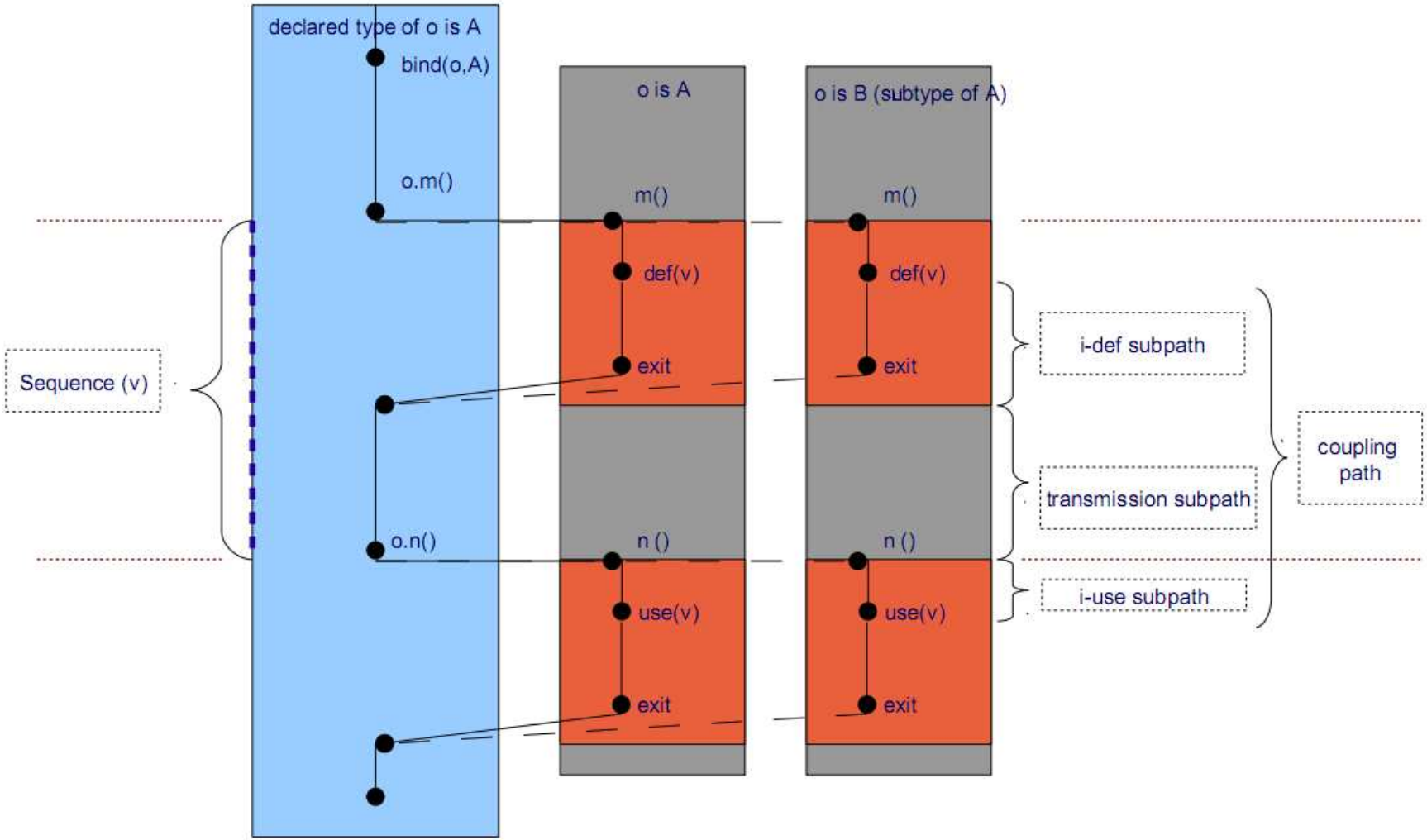
Erweiterung der kopplungsbasierten Kriterien



- Erweiterung um die Überdeckung aller Aufrufstellen, unabhängig davon, ob Parameter übergeben werden

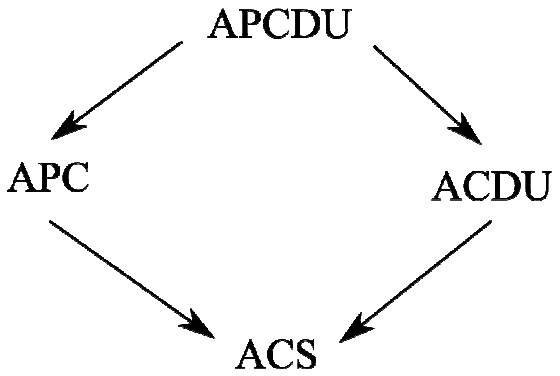
Ziel 3: Interaktionsüberdeckung für OO

Alexander & Offutt



Ziel 3: Interaktionsüberdeckung

Alexander & Offutt



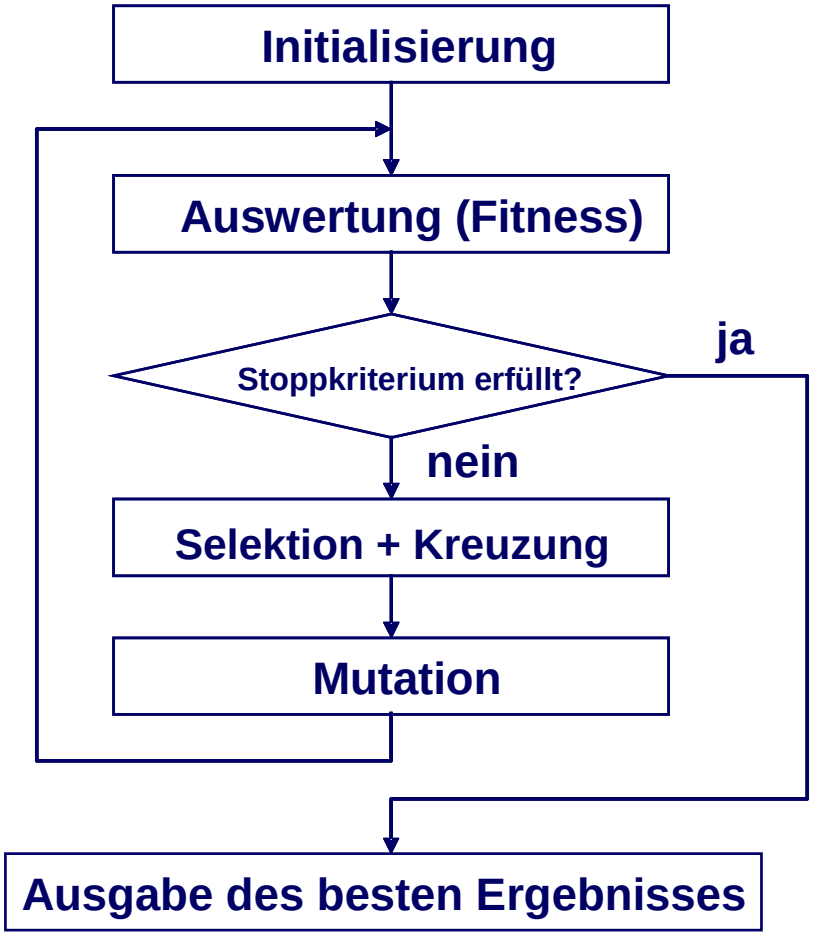
- All-Coupling-Sequences (ACS)
  - Jede Sequenz muss mindestens einmal überdeckt werden
- All-Poly-Classes (APC)
  - für jede Sequenz und jedes Mitglied einer Typfamilie im Kontext der Sequenz muss ein Kopplungspfad überdeckt werden
- All-Coupling-Defs-Uses (ACDU)
  - für jede Kopplungsvariable in jeder Sequenz müssen alle Kopplungspfade (Coupling-Def/Coupling-Use-Paare) überdeckt werden
- All-Poly-Coupling-Defs-Uses (APCDU)
  - für jede Kopplungsvariable jeder Sequenz und für jedes Mitglied der Typfamilie im Kontext der Sequenz müssen alle Kopplungspfade überdeckt werden

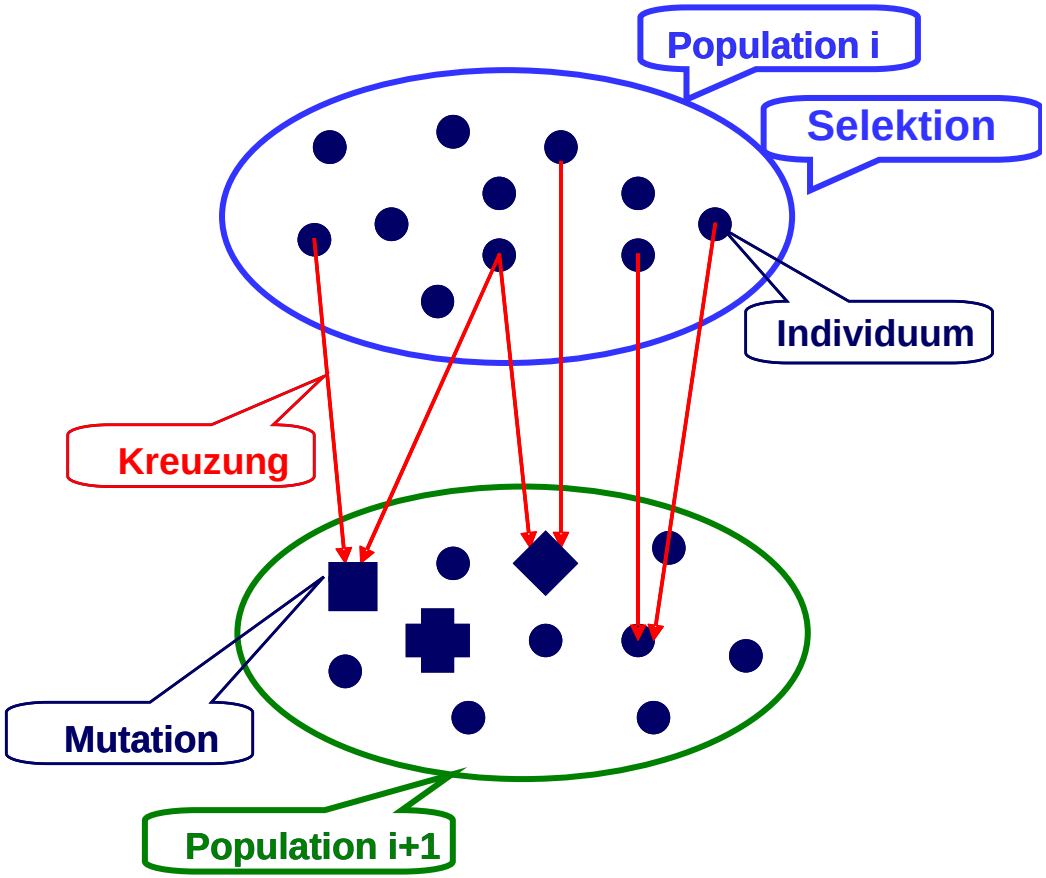
Kombination der Ziele

- Kombination von
  - Ziel 1: Betriebsprofiltreue (K.O.-Kriterium)
  - Ziel 2: Unabhängige Auswahl der Testfälle (K.O.-Kriterium)
  - Ziel 3: Hohe Interaktionsüberdeckung

- Multikriterielles Optimierungsproblem
  - Systematische Ansätze zur Bestimmung einer Testfallmenge sind ungeeignet
  - Deswegen werden Heuristiken (genetische Algorithmen) verwendet
  - Entscheidend ist dabei die Definition einer adäquaten Fitnessfunktion

Genetische Algorithmen



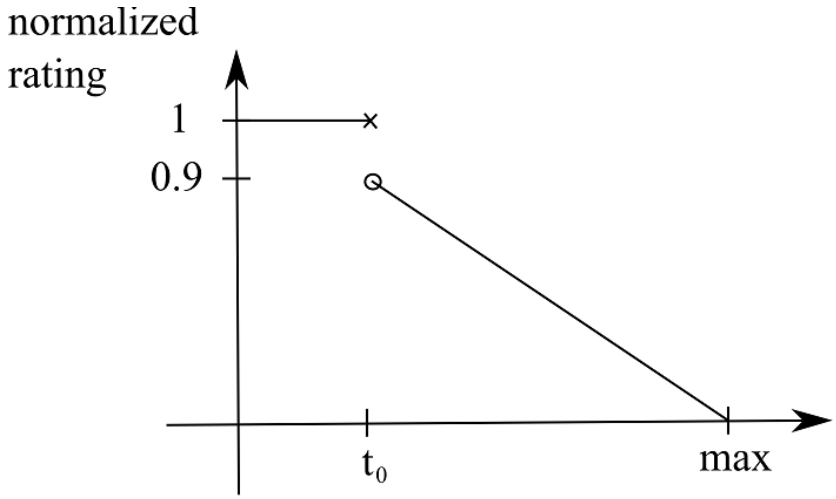


Normierung

- Der Grad der Erfüllung der einzelnen Ziele muss gemessen werden
- Deswegen werden die Werte der Goodness-of-Fit-Tests und der Korrelationsmetriken auf das Interval [0; 1] abgebildet
  - Wobei der Wert 1 die Erfüllung des Kriteriums und
  - Wobei der Wert 0 die größtmögliche Verletzung des Kritierums wiedergibt.
  - Dazwischen liegende Werte den relativen Erfüllungsgrad reflektieren
- Werte für die Interaktionsüberdeckung liegen bereits im Intervall [0; 1]

Normierung

- Wenn t_0 den maximal akzeptablen Grenzwert für die Verletzung eines Kriteriums bezeichnet, werden
  - Alle Werte von [0; t_0] als vollständige Erfüllung des Kritierums gewertet und erhalten deswegen den Wert 1
  - Alle Werte größer t_0 mit Werten < 0,9 bewertet



Fitnessbewertung

- Fitnessfunktion für genetische Optimierung:

  Fitnesswert =

  1,0 · Betriebsprofiltreue +

  1,0 · Unabhängigkeit +

  0,1 · Überdeckung

- Das Ziel der Schnittstellenüberdeckung kann dabei eine Verletzung eines der beiden K.O.-Kriterien nicht kompensieren
- Testfallmengen, die keines der beiden K.O.-Kriterien verletzen, haben Fitnesswerte ≥2
- Testfallmengen, die mindestens eines der K.O.-Kriterien verletzen, haben Fitnesswerte <2, unabhängig vom Grad der erreichten Überdeckung

Zusammenfassung & Ausblick

- Neuer Ansatz kombiniert
  - Zuverlässigkeitsbewertung mit
  - Hoher Interaktionsüberdeckung
- Durch die Verwendung genetischer Algorithmen
- Fitnessfunktion berücksichtigt
  - Betriebsprofiltreue
  - Unabhängige Auswahl der Testfälle
  - Kopplungsbasierte Überdeckungskriterien
- Ausblick
  - Stand der Implementierung
  - Anwendung auf ausgewählte Software-Projekte von Siemens Corporate Technology