

# Übung: Software-Qualität

Sommersemester 2016

[swq@se.uni-hannover.de](mailto:swq@se.uni-hannover.de)

1)

- Das Ziel beim Combinatorial Test Design (CTD) (dt. auch **kombinatorisches Testen** oder **paarweises Testen**)
  - Eine möglichst kleine Menge an Testfällen soll 100% aller möglichen Interaktionen eines bestimmten Levels abdecken

2)

- **Wie erzeugen wir diese Menge von Tests?**
  - **Tests** meinen im Folgenden die **Kombinationen von Eingabe-Parametern** – zum eigentlichen Test fehlen dann die Sollwerte

3)

- **IPOG-Algorithmus** (IPOG = In-Parameter-Order-General)
- Sei t das abzudeckende Interaktionslevel (z.B. Level-3 : t=3)
- **Idee:**
  1. Betrachte die ersten t Parameter und erstelle eine Testmenge mit 100%iger Level-t-Abdeckung
  2. Dann erweitere diese Menge sukzessive:
    - **Horizontale Ausdehnung:** Jeder existierende Test wird für den nächsten noch nicht betrachteten Parameter erweitert
    - **Vertikale Ausdehnung:** Weitere Tests werden hinzugefügt, wenn notwendig um weitere t-Tupel abzudecken

*Auch als T-Way Testing bezeichnet*

**Gegeben sind drei Parameter:  $P1$ ,  $P2$  u.  $P3$**

**Wertebereich  $P1, P2 = \{0,1\}$**

**Wertebereich  $P3 = \{2, 3\}$**

**Interaktionslevel  $t = 2$**

**Das Ziel - Erstellung einer Testmenge  
mit 100%iger Level-t-Abdeckung!**

*Step1 - Testerweiterung durch horizontale Ausdehnung*

*Step 2- Testerweiterung durch vertikale Ausdehnung*

Initiale Eingabe-Tupel

P1	P2
0	0
0	1
1	0
1	1



Step 1 – Horizontale Ausdehnung

P1	P2	P3
0	0	?
0	1	?
1	0	?
1	1	?

Im nachfolgenden finden Sie zwei Vorgehensweisen:

- Operationsweise ohne extendPlusOne() Methode – Beispielhaft
- **Operationsweise mit extendPlusOne() Methode**
  - **maximale Reduzierung der Testmenge**

Sie sollten immer versuchen das Verfahren mit extendPlusOne() zu berücksichtigen!

# IPOG-Algorithmus – ohne ExtendPlusOne()

Bilden Sie nun alle 2er-Tupel Möglichkeiten um alle Parameter abzudecken:

P1 P3

0	2
0	3
1	2
1	3

P2 P3

0	2
0	3
1	2
1	3

P1 P2 P3

0	0	?
0	1	?
1	0	?
1	1	?

Wertebereich  
von P3 einsetzen

Step 1 – Horizontale Ausdehnung

$$\pi = P1 \times P3 \cup P2 \times P3 \\ = (2 \times 2 \cup 2 \times 2)$$

# IPOG-Algorithmus – ohne ExtendPlusOne()

Bilden Sie nun alle 2er-Tupel Möglichkeiten  
um alle Parameter abzudecken:

$$\pi = P1 \times P3 \cup P2 \times P3$$

P1 P3

<del>0</del>	<del>2</del>
0	3
1	2
1	3

P2 P3

<del>0</del>	<del>2</del>
0	3
1	2
1	3

P1 P2 P3

0	0	<b>2</b>
0	1	?
1	0	?
1	1	?

Wertebereich  
von P3 einsetzen

Step 1 – Horizontale Ausdehnung

# IPOG-Algorithmus – ohne ExtendPlusOne()

Bilden Sie nun alle 2er-Tupel Möglichkeiten  
um alle Parameter abzudecken:

$$\pi = P1 \times P3 \cup P2 \times P3$$

P1 P3

<del>0</del>	<del>2</del>
<del>0</del>	<del>3</del>
1	2
1	3

P2 P3

<del>0</del>	<del>2</del>
0	3
1	2
<del>1</del>	<del>3</del>

P1 P2 P3

0	0	<div>2</div>
0	1	<div>3</div>
1	0	?
1	1	?

Wertebereich  
von P3 einsetzen

Step 1 – Horizontale Ausdehnung

Bilden Sie nun alle 2er-Tupel Möglichkeiten  
um alle Parameter abzudecken:

$$\pi = P1 \times P3 \cup P2 \times P3$$

P1 P3

<del>0</del>	<del>2</del>
<del>0</del>	<del>3</del>
<del>1</del>	<del>2</del>
1	3

P2 P3

<del>0</del>	<del>2</del>
0	3
1	2
<del>1</del>	<del>3</del>

P1 P2 P3

0	0	<b>2</b>
0	1	3
1	0	<b>2</b>
1	1	?

Wertebereich  
von P3 einsetzen

Step 1 – Horizontale Ausdehnung



Bilden Sie nun alle 2er-Tupel Möglichkeiten  
um alle Parameter abzudecken:

$$\pi = P1 \times P3 \cup P2 \times P3$$

P1 P3

0	2
0	3
1	2
1	3

P2 P3

0	2
0	3
1	2
1	3

P1 P2 P3

0	0	2
0	1	3
1	0	2
1	1	3

Wertebereich  
von P3 einsetzen

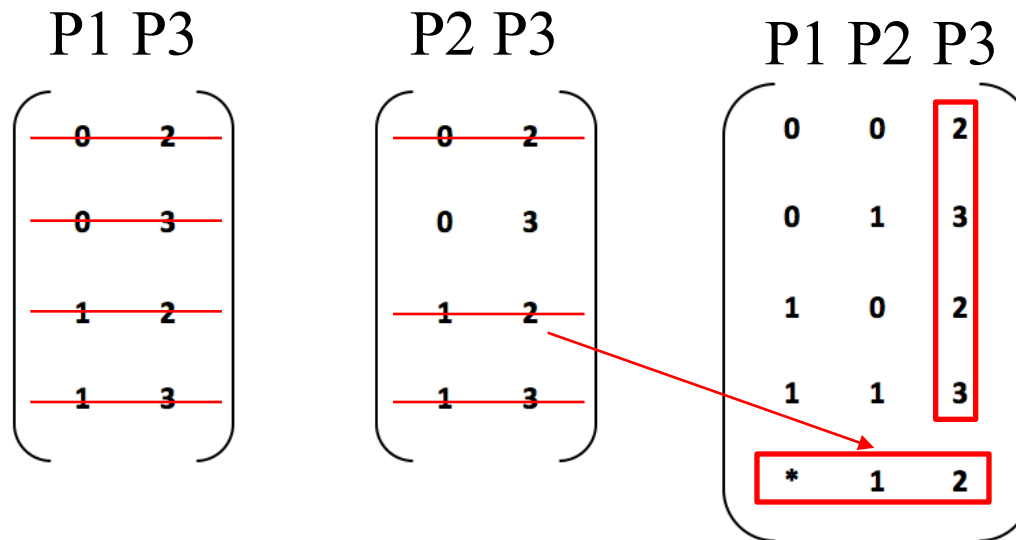
Verbleiben für Step 2

Step 1 – Horizontale Ausdehnung

# IPOG-Algorithmus – ohne ExtendPlusOne()

Bilden Sie nun alle 2er-Tupel Möglichkeiten  
um alle Parameter abzudecken:

$$\pi = P1 \times P3 \cup P2 \times P3$$

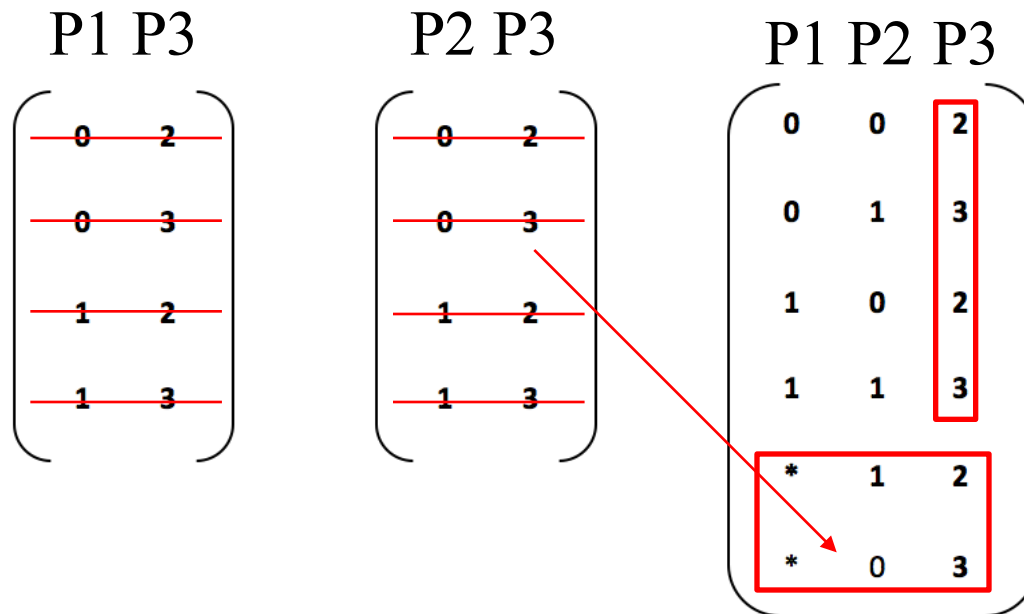


Step 2 – Vertikale Ausdehnung

# IPOG-Algorithmus – ohne ExtendPlusOne()

Bilden Sie nun alle 2er-Tupel Möglichkeiten  
um alle Parameter abzudecken:

$$\pi = P1 \times P3 \cup P2 \times P3$$

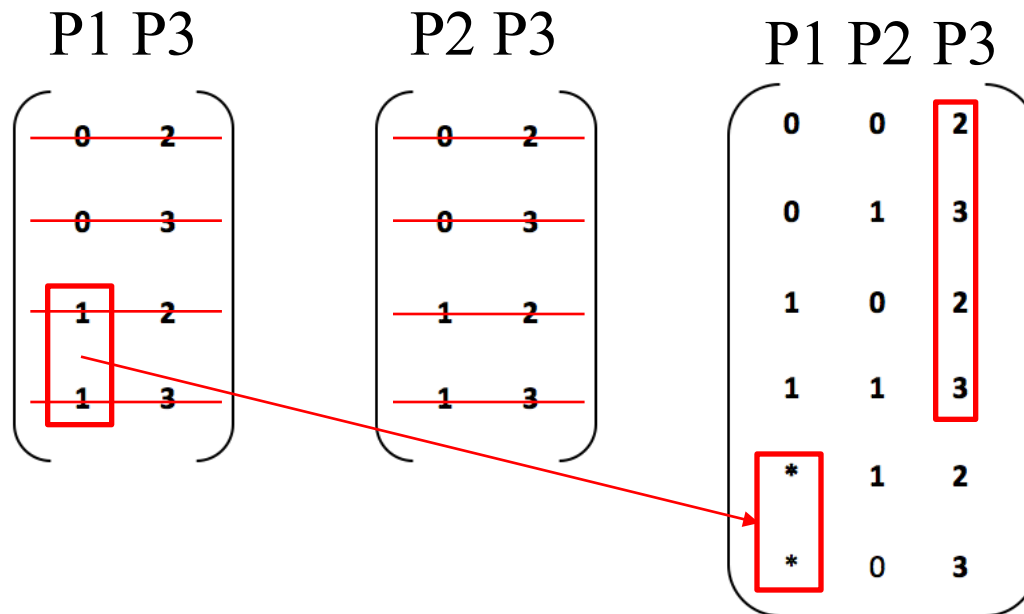


Step 2 – Vertikale Ausdehnung

# IPOG-Algorithmus – ohne ExtendPlusOne()

Bilden Sie nun alle 2er-Tupel Möglichkeiten  
um alle Parameter abzudecken:

$$\pi = P1 \times P3 \cup P2 \times P3$$



Step 2 – Vertikale Ausdehnung

Sinnig füllen, sodass auch  
P1 abgedeckt werden kann

# IPOG-Algorithmus – ohne ExtendPlusOne()

Testmenge nach horizontaler u.  
vertikaler Ausdehnung mit  
Interaktionslevel  $t = 2$

P1	P2	P3
0	0	2
0	1	3
1	0	2
1	1	3
1	1	2
1	0	3

VS

Testmenge ohne Anwendung  
des IPOG-Algorithmus

P1	P2	P3
0	0	2
0	1	2
1	0	2
1	1	2
0	0	3
0	1	3
1	0	3
1	1	3

Reduzierung um 2 Test Tuple  
-> nicht das Maximal mögliche!

# IPOG-Algorithmus – mit ExtendPlusOne()

Bilden Sie nun alle 2er-Tupel Möglichkeiten  
um alle Parameter abzudecken:

Wert aus Wertebereich von P3  
einsetzen, welches das  
Maximum an Testfälle abdeckt!

P1 P3		P2 P3		P1 P2 P3		
0	2	0	2	0	0	?
0	3	0	3	0	1	?
1	2	1	2	1	0	?
1	3	1	3	1	1	?

Step 1 – Horizontale Ausdehnung

$$\pi = P1 \times P3 \cup P2 \times P3 \\ = (2 \times 2 \cup 2 \times 2)$$

# IPOG-Algorithmus – mit ExtendPlusOne()

Bilden Sie nun alle 2er-Tupel Möglichkeiten  
um alle Parameter abzudecken:

Max Kombinationsmöglichkeit

P1 P3		P2 P3		P1 P2 P3		
<del>0</del>	<del>2</del>	<del>0</del>	<del>2</del>	0	0	2
0	3	0	3	0	1	?
1	2	1	2	1	0	?
1	3	1	3	1	1	?

Step 1 – Horizontale Ausdehnung

# IPOG-Algorithmus – mit ExtendPlusOne()

Max Kombinationsmöglichkeit

P1 P3		P2 P3		P1 P2 P3		
<del>0</del>	<del>2</del>	<del>0</del>	<del>2</del>	0	0	2
<del>0</del>	<del>3</del>	0	3	0	1	3
1	2	1	2	1	0	?
1	3	<del>1</del>	<del>3</del>	1	1	?

Step 1 – Horizontale Ausdehnung



# IPOG-Algorithmus – mit ExtendPlusOne()

Max Kombinationsmöglichkeit

P1 P3		P2 P3		P1 P2 P3		
<del>0</del>	<del>2</del>	<del>0</del>	<del>2</del>	0	0	2
<del>0</del>	<del>3</del>	<del>0</del>	<del>3</del>	0	1	3
1	2	1	2	1	0	3
<del>1</del>	<del>3</del>	<del>1</del>	<del>3</del>	1	1	?

Step 1 – Horizontale Ausdehnung

Max Kombinationsmöglichkeit

P1 P3		P2 P3		P1 P2 P3		
0	2	0	2	0	0	2
0	3	0	3	0	1	3
1	2	1	2	1	0	3
1	3	1	3	1	1	2

Step 1 – Horizontale Ausdehnung

Step 2 – Vertikale Ausdehnung entfällt da bereits 100% der Testmenge abgedeckt werden konnte über die Horizontale Ausdehnung!

# IPOG-Algorithmus – mit ExtendPlusOne()

Testmenge IPOG-Algorithmus  
und ExtendedPlusOne()  
nach horizontaler mit  
Interaktionslevel  $t = 2$

P1	P2	P3
0	0	2
0	1	3
1	0	3
1	1	2

Vs.

Testmenge ohne Anwendung  
des IPOG-Algorithmus

P1	P2	P3
0	0	2
0	1	2
1	0	2
1	1	2
0	0	3
0	1	3
1	0	3
1	1	3

Maximum an Test Tuple eingespart