Specification-based testing -

specification

↓

input conditions

↓

valid subdomains & invalid -11-

↓

test cases

Equivalence Testing.

1. Normal Eq. testing.

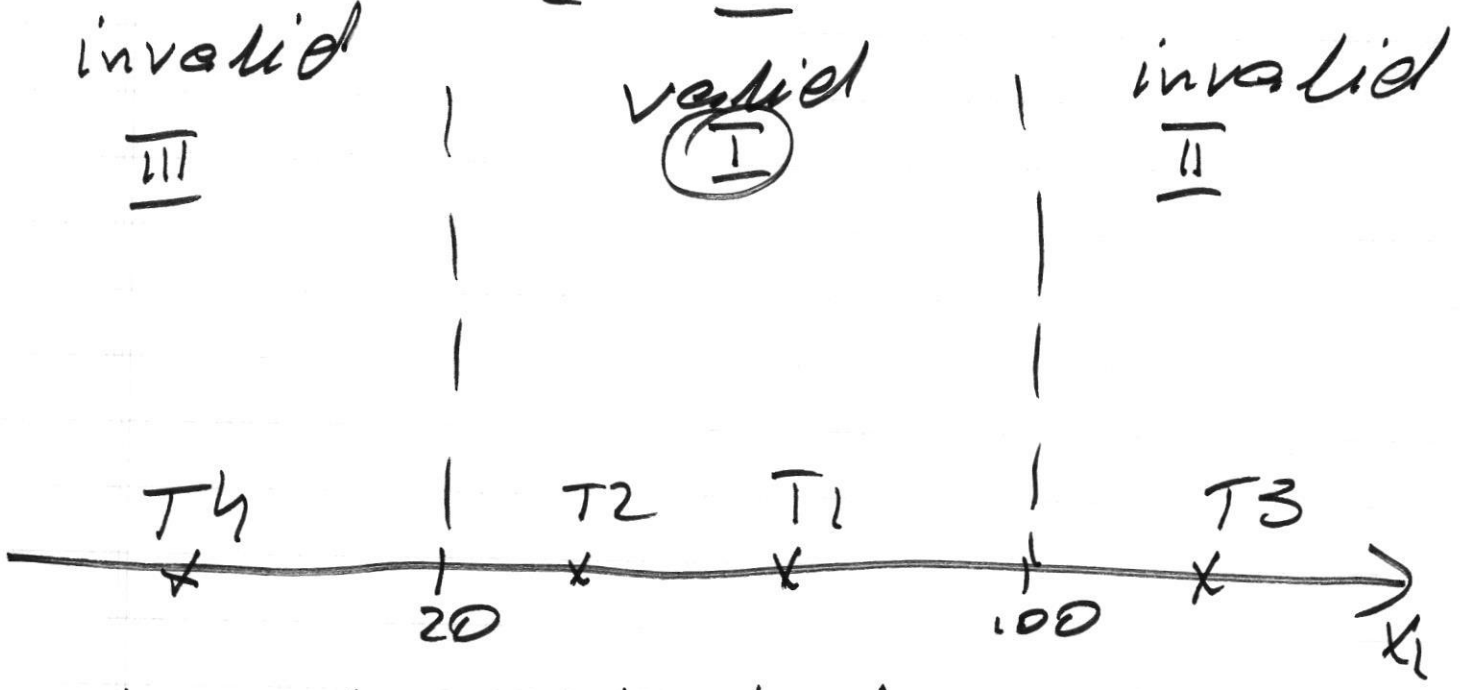
   several test cases from each valid subdomain

2. Robust Eq. testing.

   select test cases from invalid subdomains

1 input variable
1 input condition

$20 \leq X_1 \leq 100$



1. Normal Eq. tests (valid subdomain)

T1: $X_1 = 75$

T2: $X_1 = 35$

2. Robust Eq. tests (invalid subdomains)

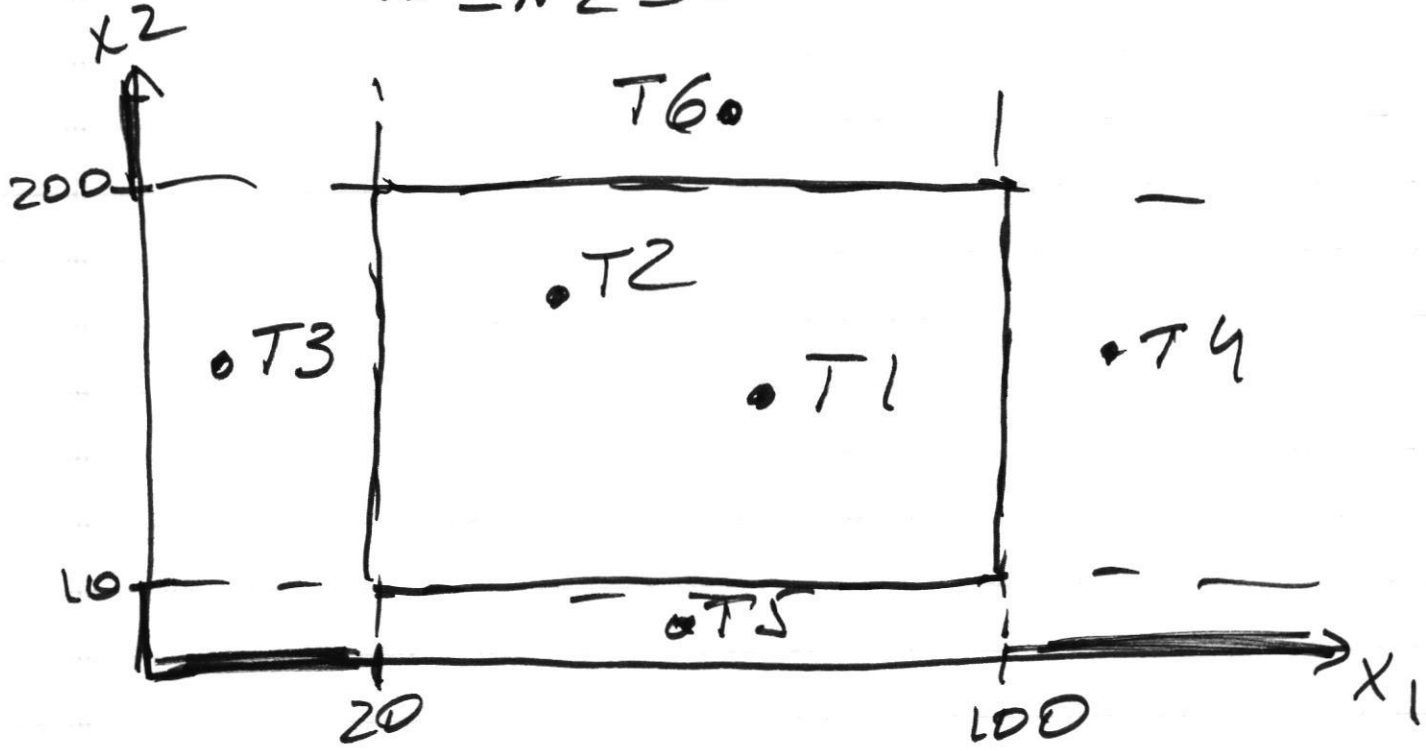
T3: $X_1 = 125$

T4: $X_1 = -10$

multiple input variables
multiple input conditions

2 input variables: $x1$, $x2$
2 input conditions

$$20 \leq x1 \leq 100$$
$$10 \leq x2 \leq 200$$



Normal Eq. tests

T1: $x1 = 85, x2 = 72$

T2: $x1 = 42, x2 = 120$

Robust Eq. tests

T3: $x1 = 10, x2 = 72$

T4: $x1 = 125, x2 = 72$

T5 $x1 = 64, x2 = 5$

T6: $x1 = 64, x2 = 210$

2 input variables

$20 \leq X1 \leq 100$

3 valid subdomains.

$20 \leq X1 \leq 40$

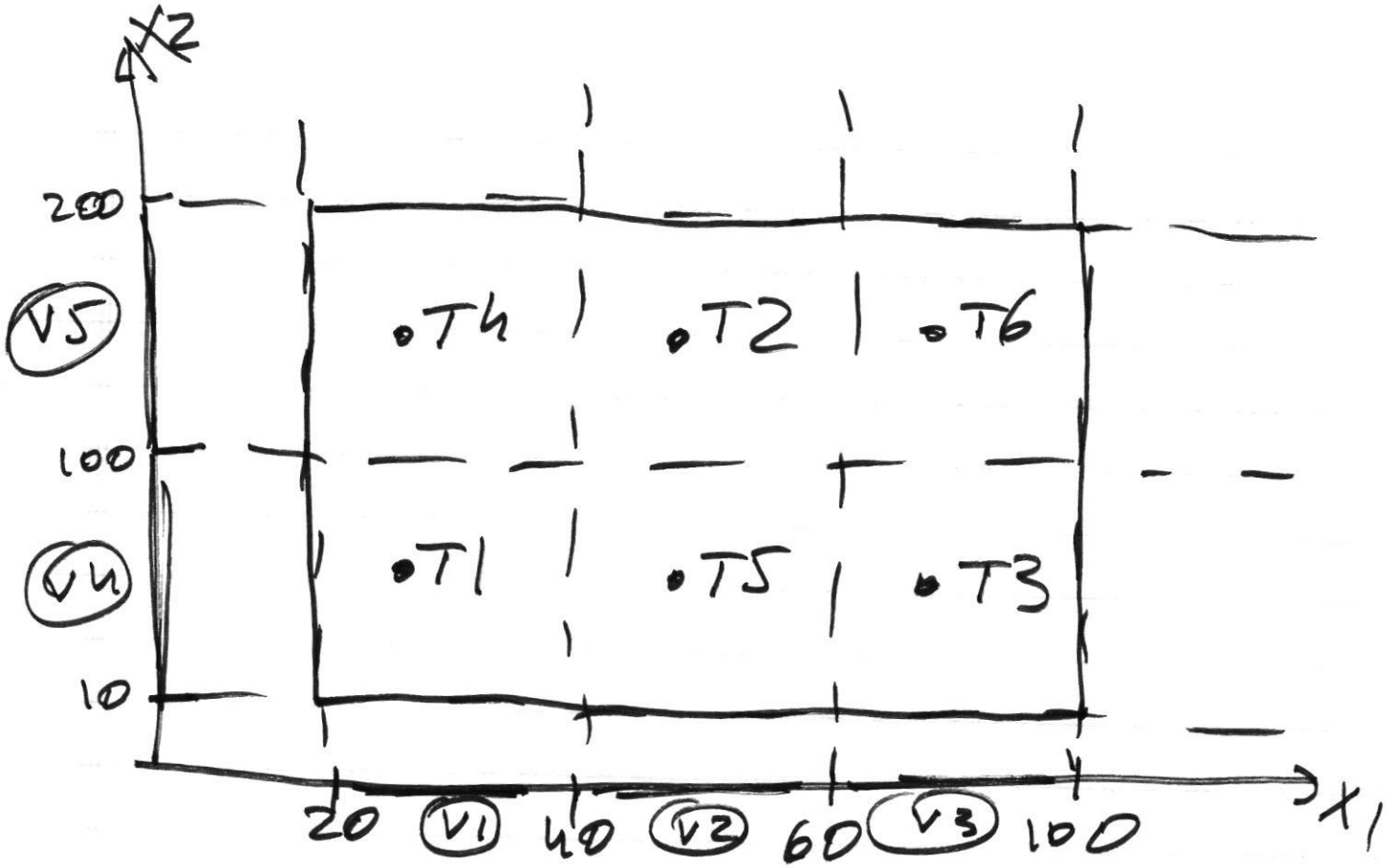
$40 < X1 \leq 60$

$60 < X1 \leq 100$

$10 \leq X2 \leq 200$

$10 \leq X2 \leq 100$

$100 < X2 \leq 200$



x_1: 3 valid subdomains
x_2: 2 valid —"—
3 valid subdomains

Weak Normal Eq. Testing	Strong Normal Eq. testing.
T1: $x_1=25$, $x_2=64$ (V1, V4)	3·2 = 6 tests
T2: $x_1=50$, $x_2=125$ (V2, V5)	T1 :
T3 : $x_1=25$, $x_2=64$ (V3)	T2 :
	T3 :
	T4 : $x_1=25$, $x_2=125$
	T5 : $x_1=50$, $x_2=75$
	T6 : $x_1=70$, $x_2=120$

Normal Eq. testing.

1. Weak Normal Eq. testing.
   1-dim subdomains.

2. strong Normal Eq. testing.
   multi-dim subdomains.

Weak Normal Eq. testing.

assumption:

input variables are not related

Tests are selected from 1-dim subdomains.

Strong Normal Eq. testing

1. input variables are related
2. multi-dim subdomains
   select tests from each
   multi-dim subdomain.

5 input variables

		valid subdomains
x_1:	4	— | | | —
x_2:	2	— | | —
x_3:	5	— | | | | —
x_4:	2	— | | —
x_5:	3	— | | | —

(I) Weak Normal Eq. tests

$4+2+5+2+3 = 16$ 1-dim subdomains

$5 \leq \#^{of}_{}\text{tests} \leq 16$

(II) Strong Normal Eq. tests

$4 \cdot 2 \cdot 5 \cdot 2 \cdot 3 = 240$ 5-dim subdomains

min # of tests = 240

Integer array dimension declaration

int n(d [,d]...)

n is the symbolic name of the array

d is a dimension declaration

Symbolic name can have 1-6 letters or digits, where the first character must be a letter.

The minimum # of dimensions = 1

The maximum # of dimensions = 7

The format of a dimension declaration

[lb:] ub

lb: lower bound; ub: upper bound

The bound may be in the range

-65,534 to +65,535

If lb is not specified, lb =1

ub ≥ lb

input conditions	valid sub-domain(s)	invalid sub-domain(s)
size of array name	1 - 6 (1) ✓	0; (2) > 6 (3)
C1 array name has letters/digits	has letters (4) ✓ has digit (5) ✓	something (6) else
array name starts with a letter	yes (7) ✓ .	no (8) .
# of dimensions	1 - 7 (9) ✓	0; (10) > 7 (11)
lower bound	-65,534 - +65,535 (12) ✓	<-65,534 - (13) >+65,535 (14)
upper bound	-65,534 - +65,535 (15) ✓	<-65,534 - (16) >+65,535 (17)
C2 lower bound specified	yes (18) ✓ no (19) ✓	
C3 upper bound to lower bound	ub > lb (20) ✓ ub = lb (21)	ub < lb (22)

Integer array declaration

Weak Normal Eq. tests

T1: int A(1:5)

T2: int A1(1:5)

T3: int A(5)

T4: int A(5:5)

---

min # of tests

T1: int A1(1:5)

T2: int A(1)

Strong Normal Eq. tests

C1, C2, C3

2 · 2 · 2 = 8 valid subdomains

8 tests

C1: array has digit
- no (A)
- yes (B)

C2: lower bound specified
- yes (A)
- no (B)

C3: upper-bound to lower bound
- ub > lb (A)
- ub = lb (B)

Assumption:

C1, C2, C3: are ~~=~~ related

Test #	C1	C2	C3	Tests
1	A	A	A	int A(1:5)
2	A	A	B	int A(5:5)
3	A	B	A	int A(5)
4	A	B	B	int A(1)
5	B	A	A	int A1(1:5)
6	B	B	A	int A1(5:5)
7	B	A	B	int A1(5)
8	B	B	B	int A1(1)

Observations

I: C1 is not related to C2

II C1 is not related to C3

III C2 is related to C3

Strong Normal Ep. tests.

We should consider

C2 and C3: 2-dim subdomain

and

C1: - 1-dim subdomain

C2: lower bound specified — yes (A)
                          — no (B)

C3: ub & lb — ub > lb (A)
            — ub = lb (B)

C4: has a digit — yes ✓
                — no ✓

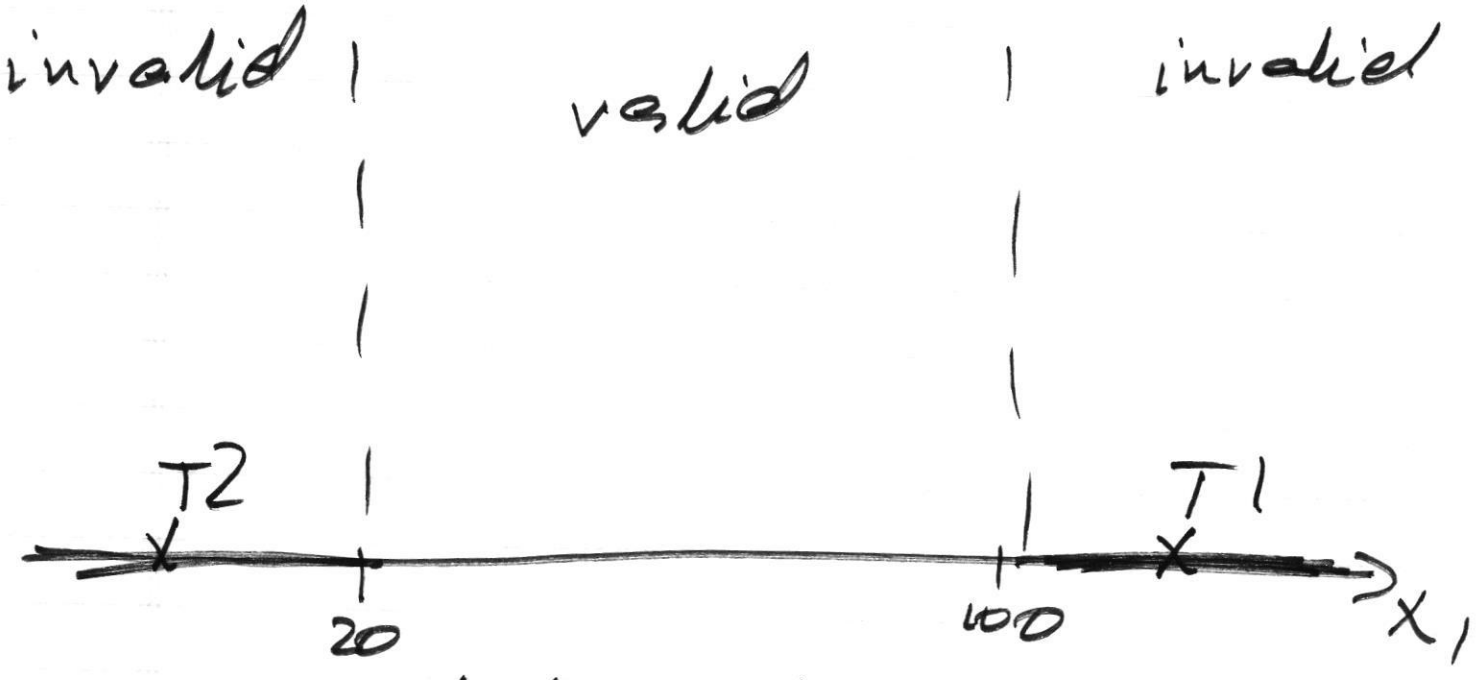
	C2	C3	Tests
T1	A	A	int A(1:5)
T2	A	B	int A1(5:5)
T3	B	A	int A(5)
T4	B	B	int A(1)

Robust Eq. testing.

1-dim subdomain

1 variable

$20 \leq x_1 \leq 100$



2 invalid subdomains

T1: $x_1 = 125$

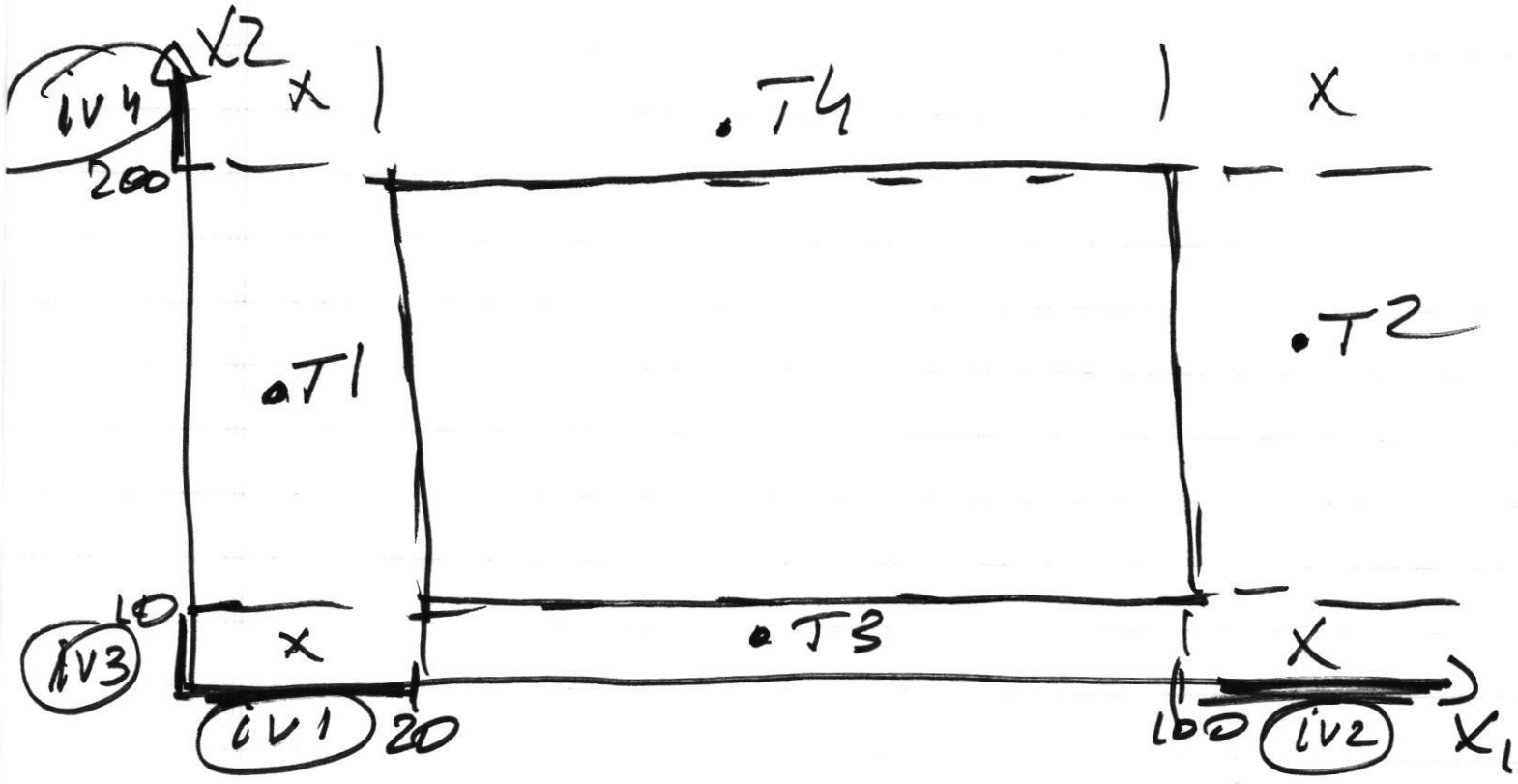
T2: $x_1 = 10$

multiple input variables
multiple input conditions

x_1, x_2

$20 \le x1 \le 100$

$10 \le x2 \le 200$



x1: invalid
x2: valid

iv1: T1: x1 = 10, x2 = 75
iv2: T2: x1 = 125, x2 = 75

x1: valid
x2: invalid

iv3: x1 = 65, x2 = 5
iv4: x1 = 65, x2 = 225

Robust Eq. testing.

1. Weak Robust Eq. tests.
   1-dim invalid sub-domains

2. Strong Robust Eq. tests
   multi-dim invalid subdomains.