Some Thoughts on Component Models

Tom Henzinger

3.14

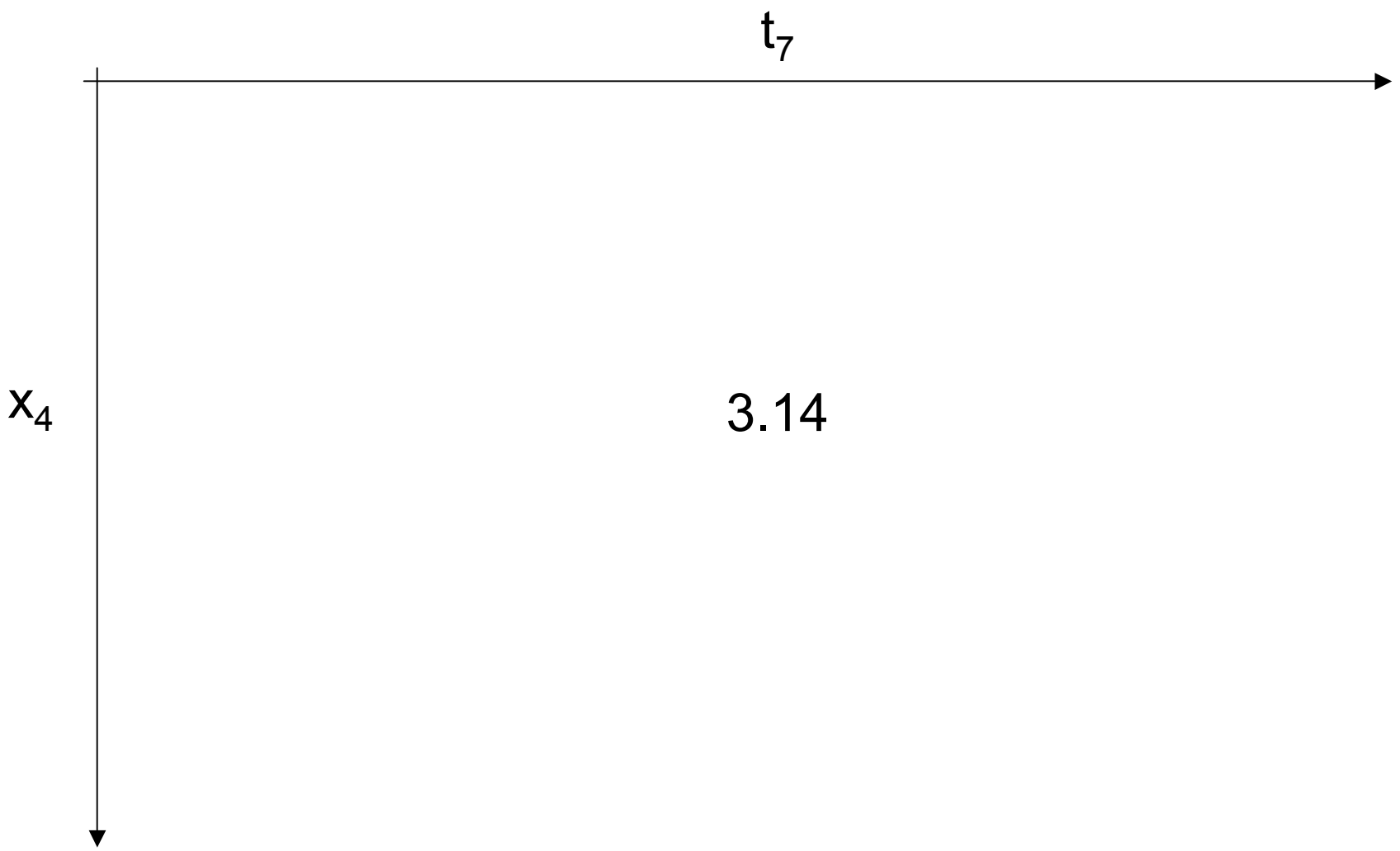
value

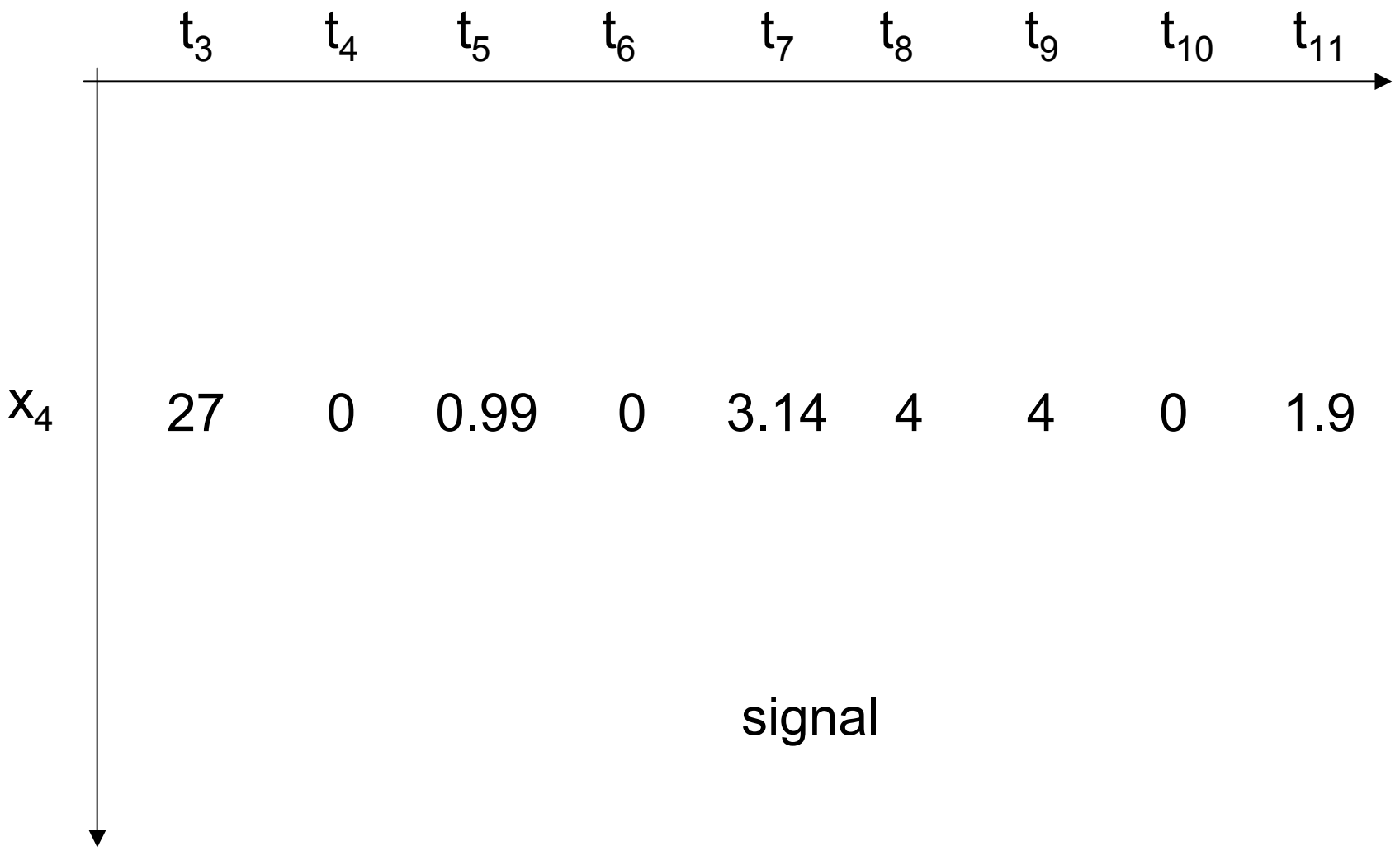
$(3.14, t_7)$

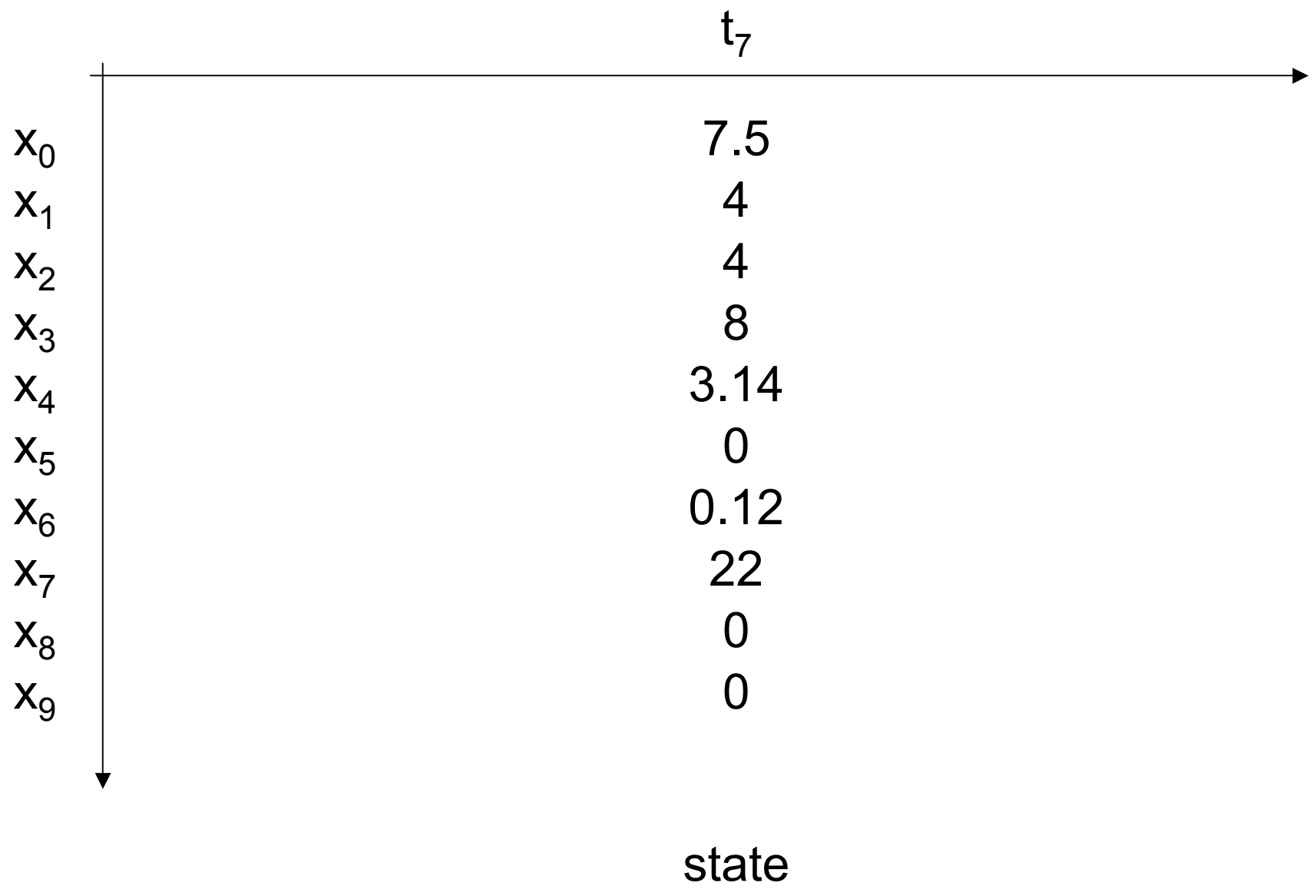
(value, time)

$(x_4, 3.14, t_7)$

(space, value, time)


t_7
x_4
3.14


t_3 t_4 t_5 t_6 t_7 t_8 t_9 t_{10} t_{11}
x_4
27 0 0.99 0 3.14 4 4 0 1.9
signal

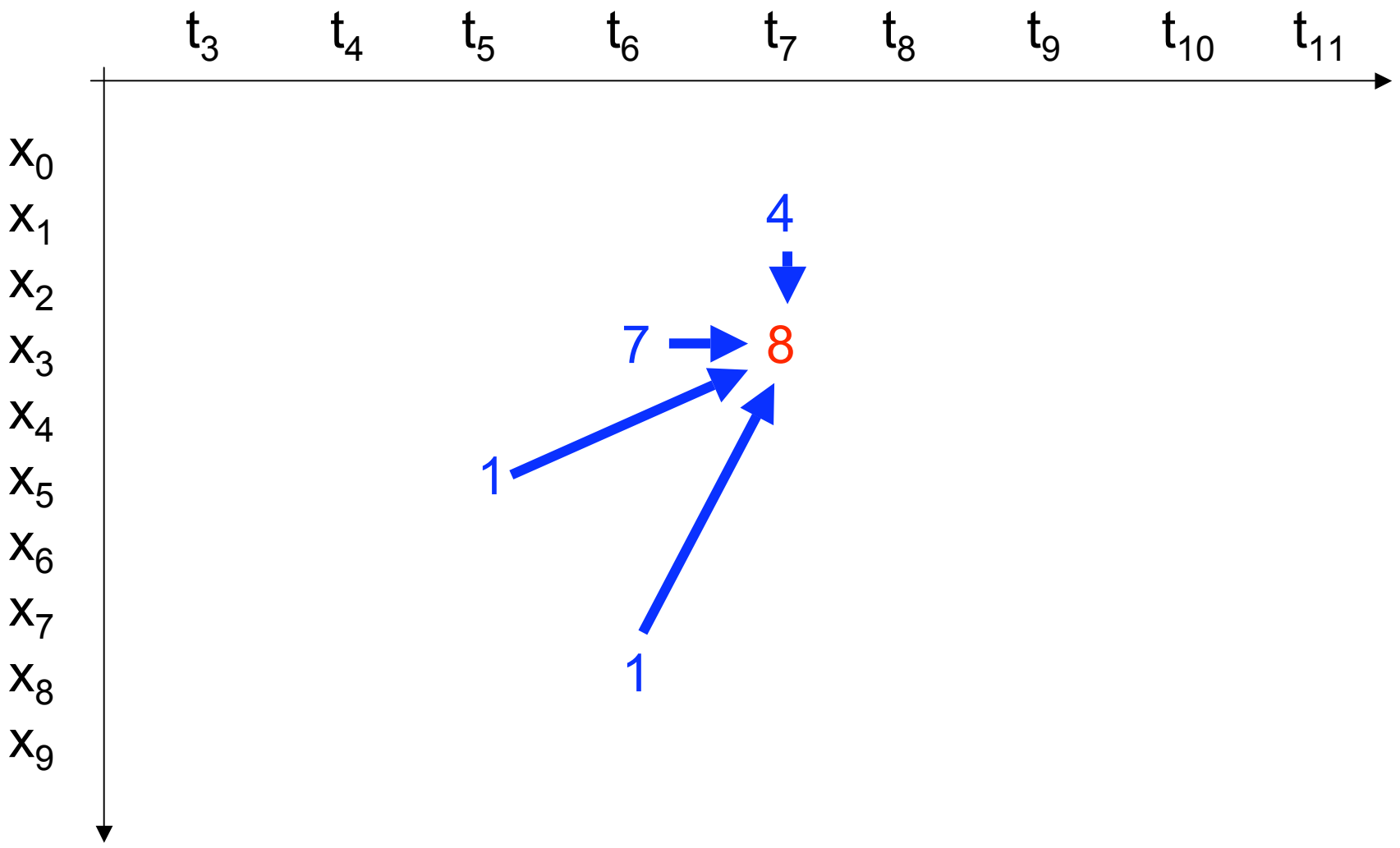

t_7
x_0
x_1
x_2
x_3
x_4
x_5
x_6
x_7
x_8
x_9
7.5
4
4
8
3.14
0
0.12
22
0
0
state

	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}
x_0	25	25	7.5	7.5	7.5	15	15	15	15
x_1	0	1	2	3	4	5	6	7	8
x_2	3	3	3	4	4	4	4	4	4
x_3	3	4	5	7	8	9	10	11	12
x_4	27	0	0.99	0	3.14	4	4	0	1.9
x_5	0	0	1	0	0	0	1	0	0
x_6	7	7.3	0	0.13	0.12	6.9	6.8	7	7
x_7	22	22	22	22	22	22	22	22	22
x_8	0	0	0	1	0	0	0	1	0
x_9	7	0	1	7.3	0	2	3	0.13	0.12

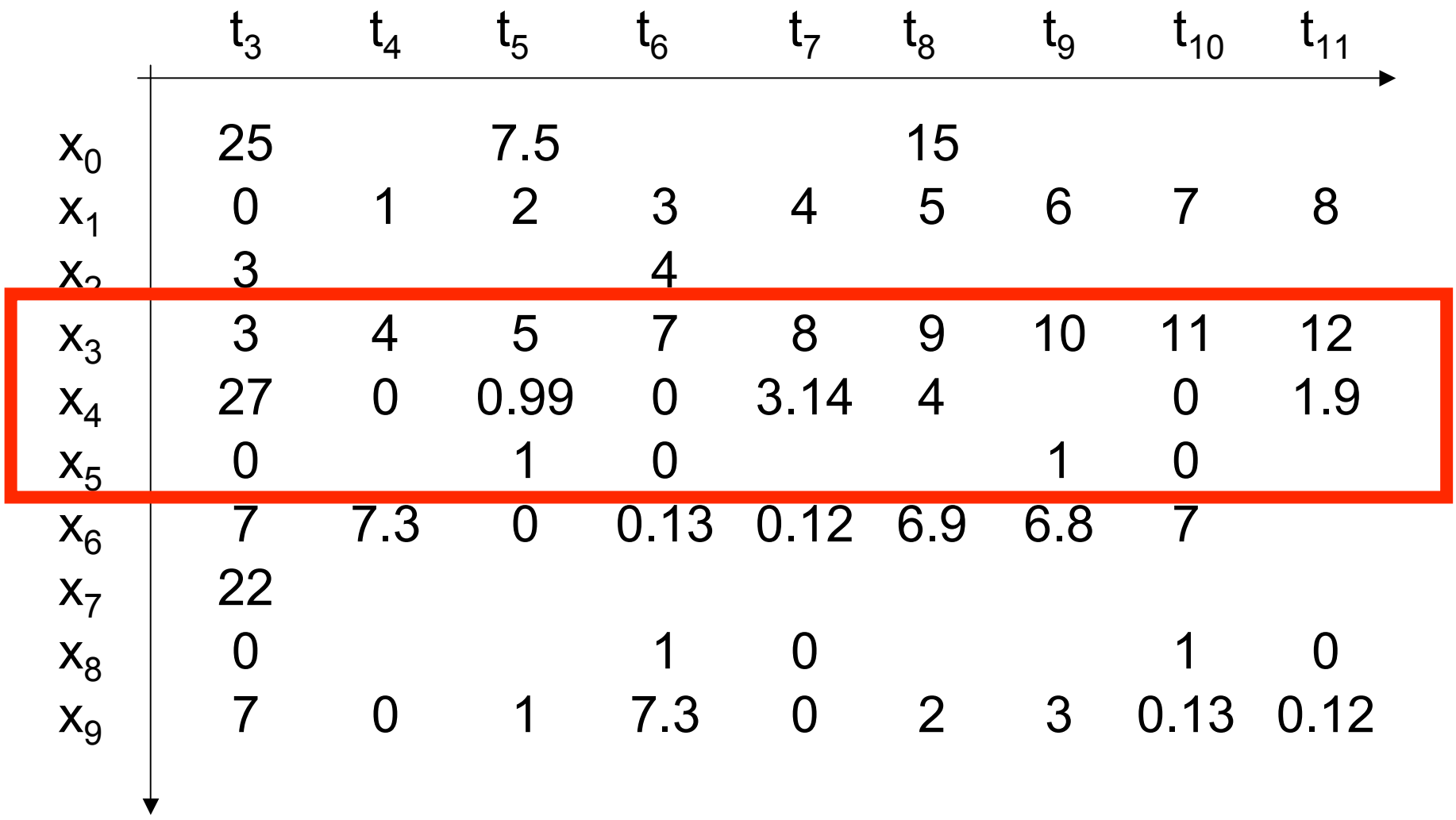
behavior

	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}
x_0	25		7.5			15			
x_1	0	1	2	3	4	5	6	7	8
x_2	3			4					
x_3	3	4	5	7	8	9	10	11	12
x_4	27	0	0.99	0	3.14	4		0	1.9
x_5	0		1	0			1	0	
x_6	7	7.3	0	0.13	0.12	6.9	6.8	7	
x_7	22								
x_8	0			1	0			1	0
x_9	7	0	1	7.3	0	2	3	0.13	0.12

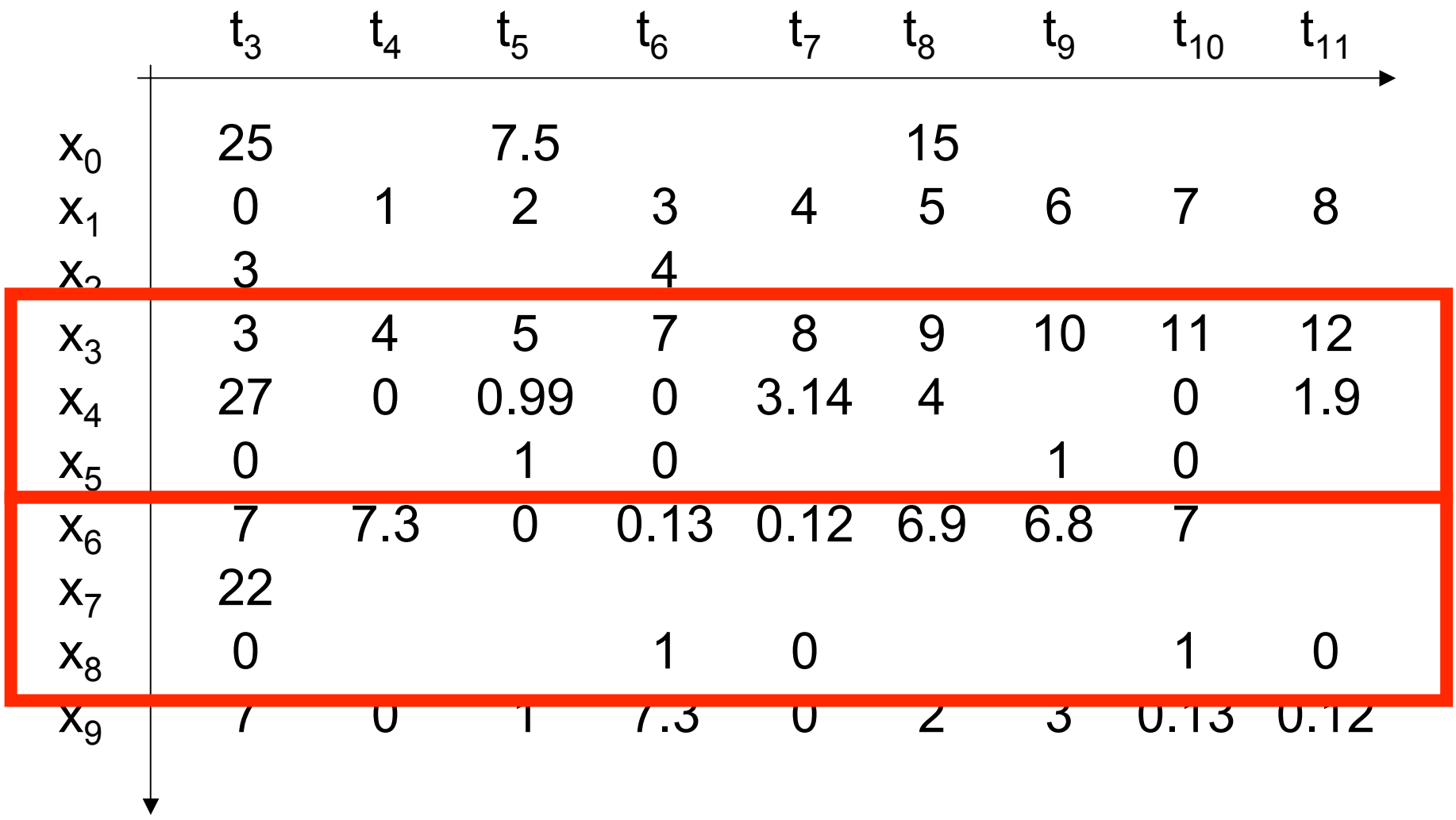
behavior



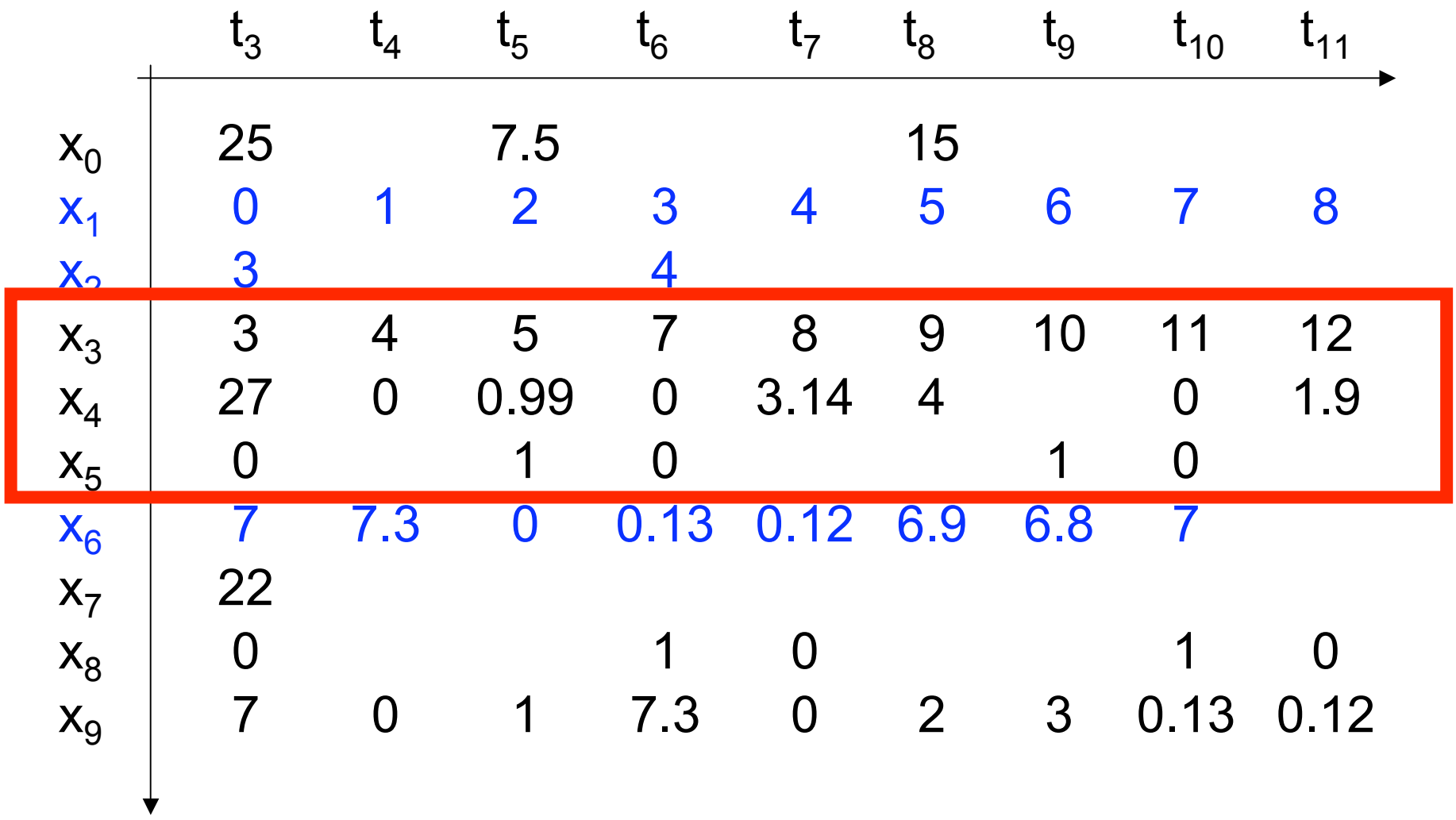
dependencies


t_3 t_4 t_5 t_6 t_7 t_8 t_9 t_{10} t_{11}
x_0 25 7.5 15
x_1 0 1 2 3 4 5 6 7 8
x_2 3 4
x_3 3 4 5 7 8 9 10 11 12
x_4 27 0 0.99 0 3.14 4 0 1.9
x_5 0 1 0 1 0
x_6 7 7.3 0 0.13 0.12 6.9 6.8 7
x_7 22
x_8 0 1 0 1 0
x_9 7 0 1 7.3 0 2 3 0.13 0.12

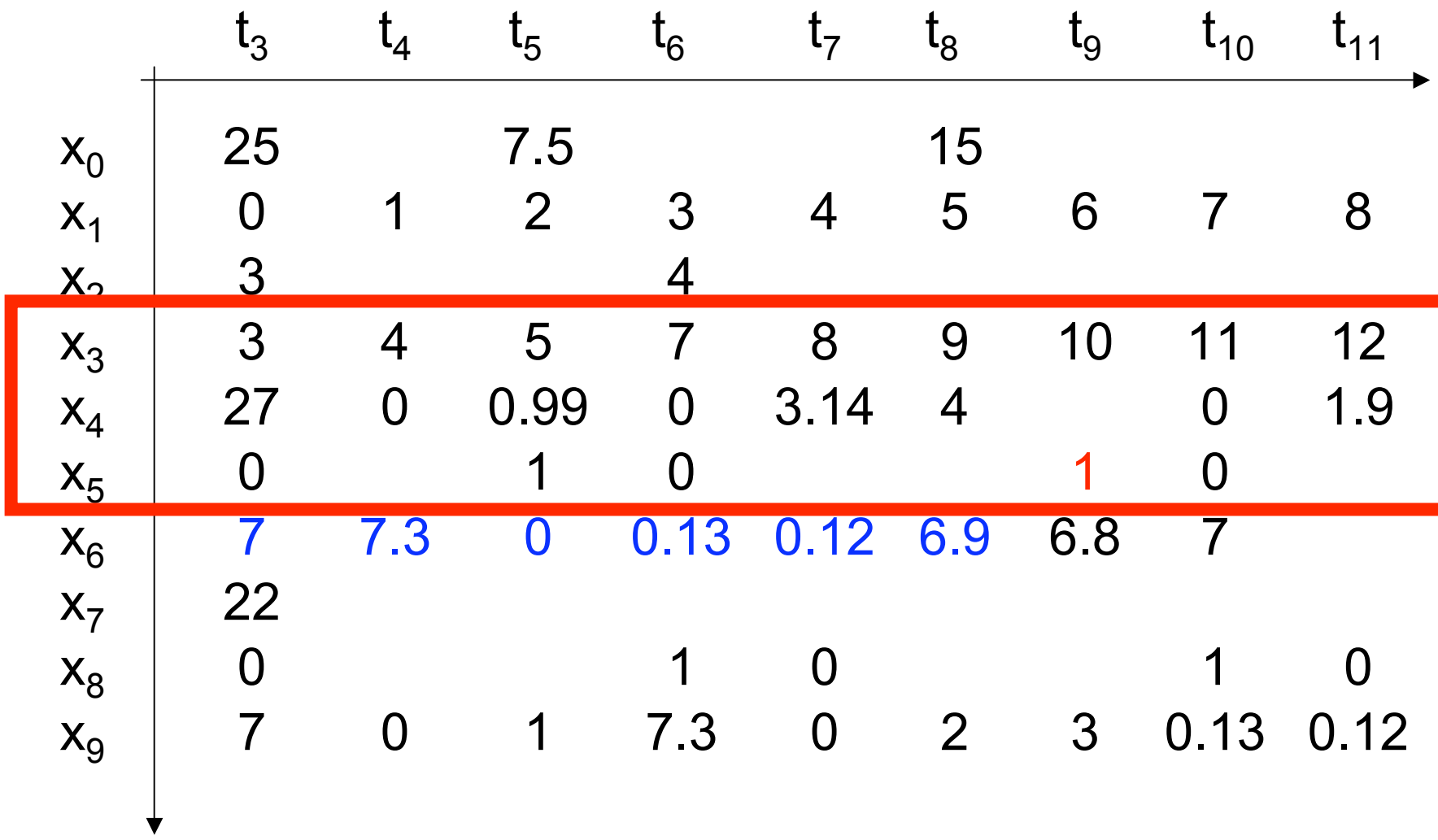
actor



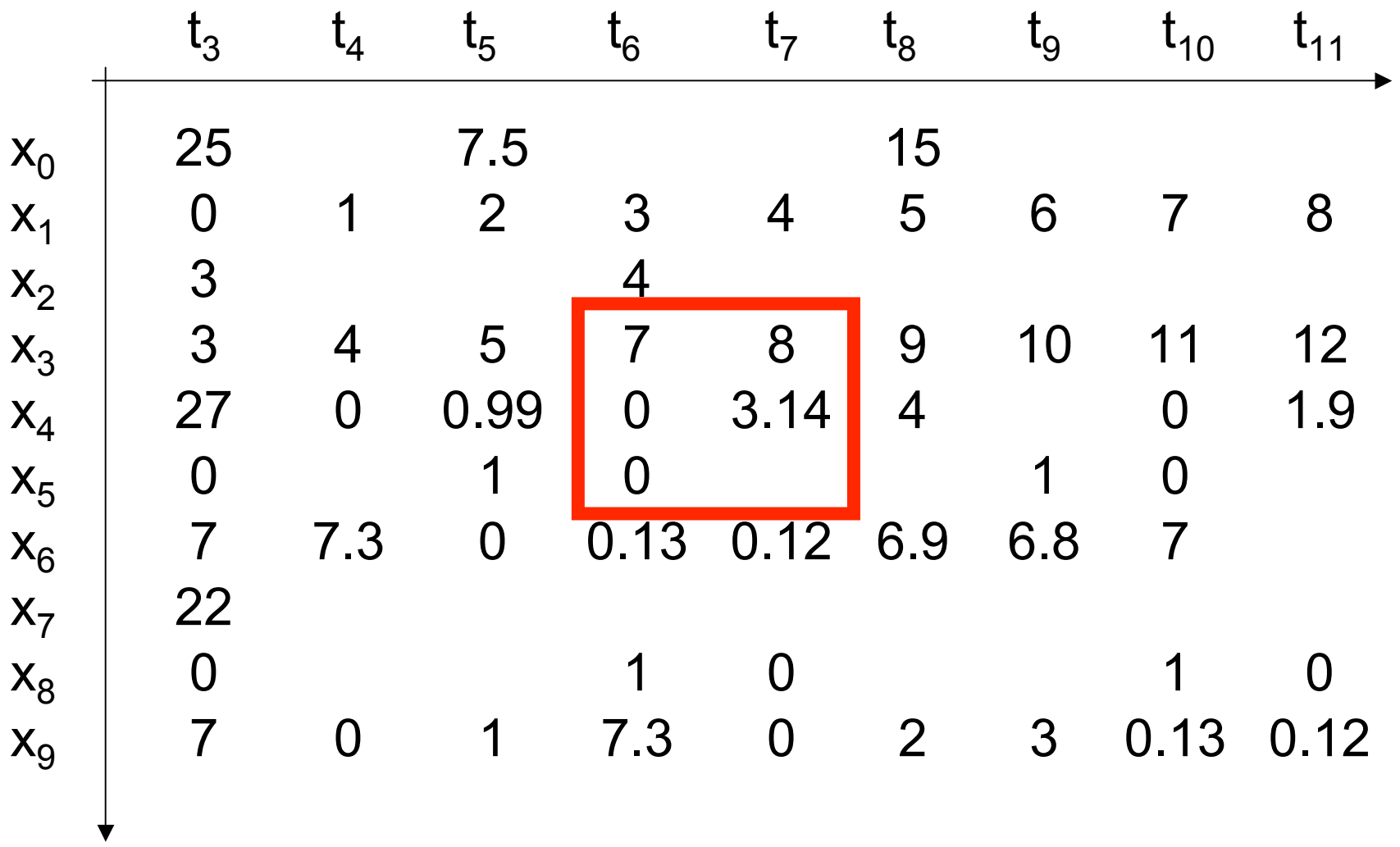
composition



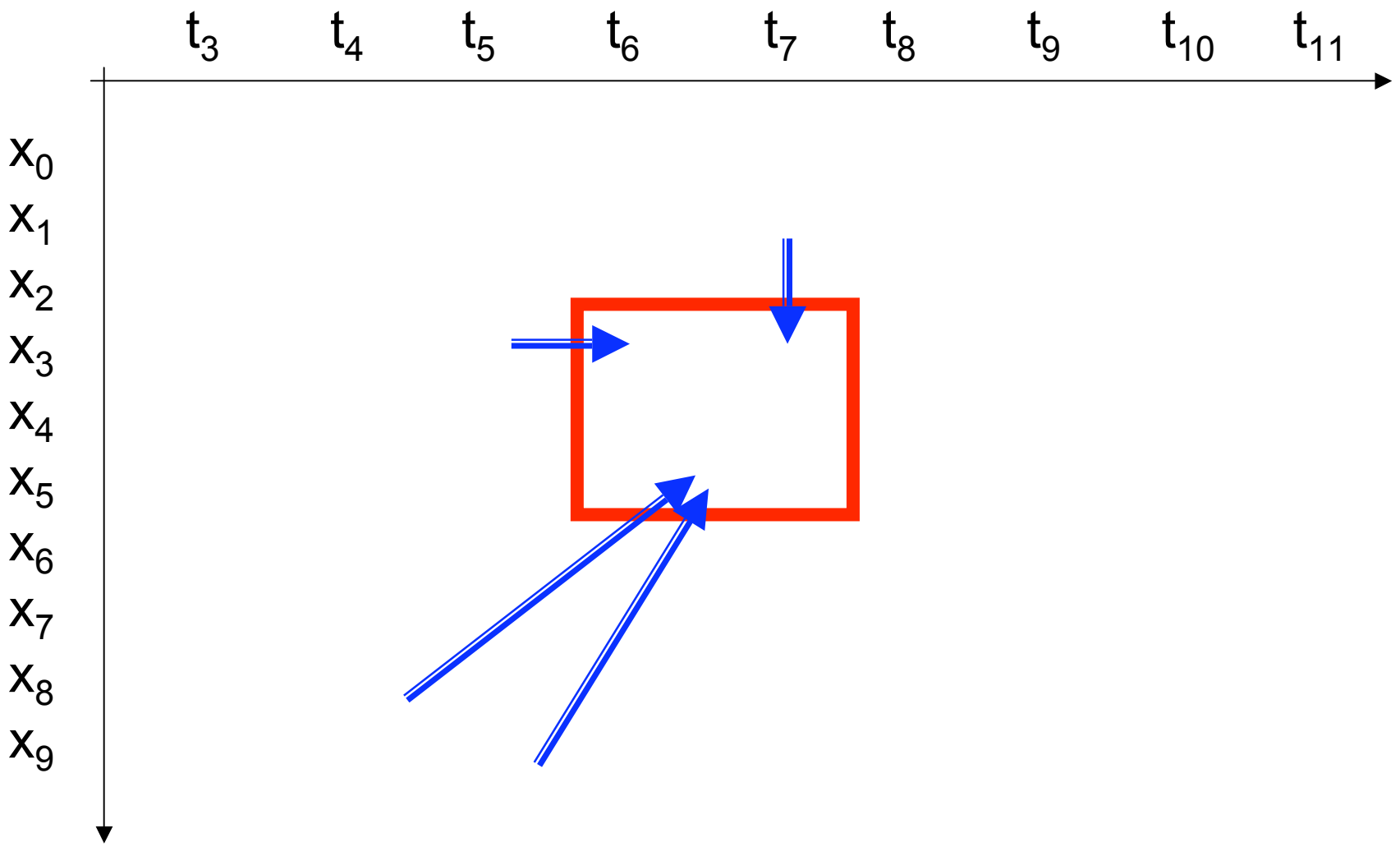
space dependencies static



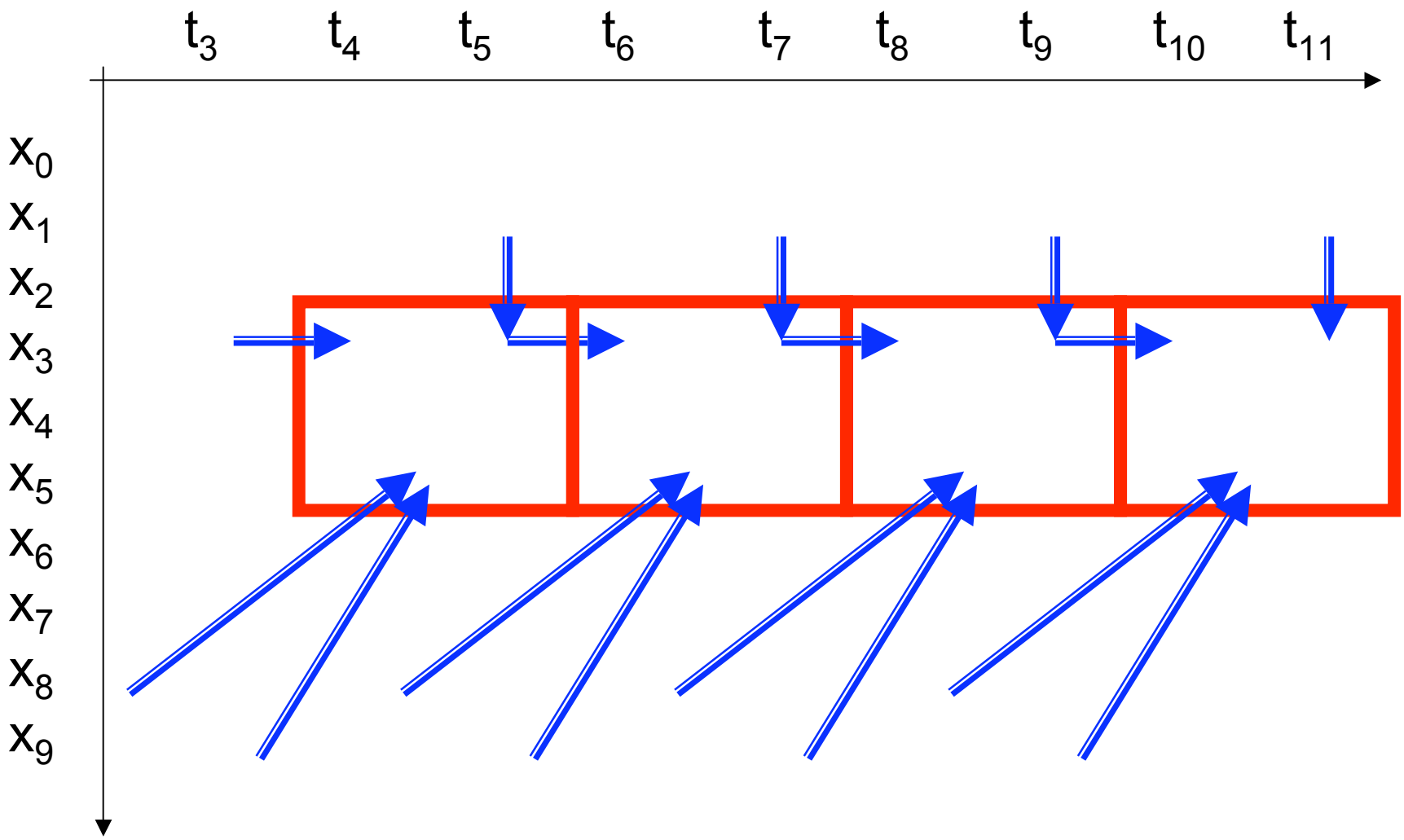
time dependencies unbounded

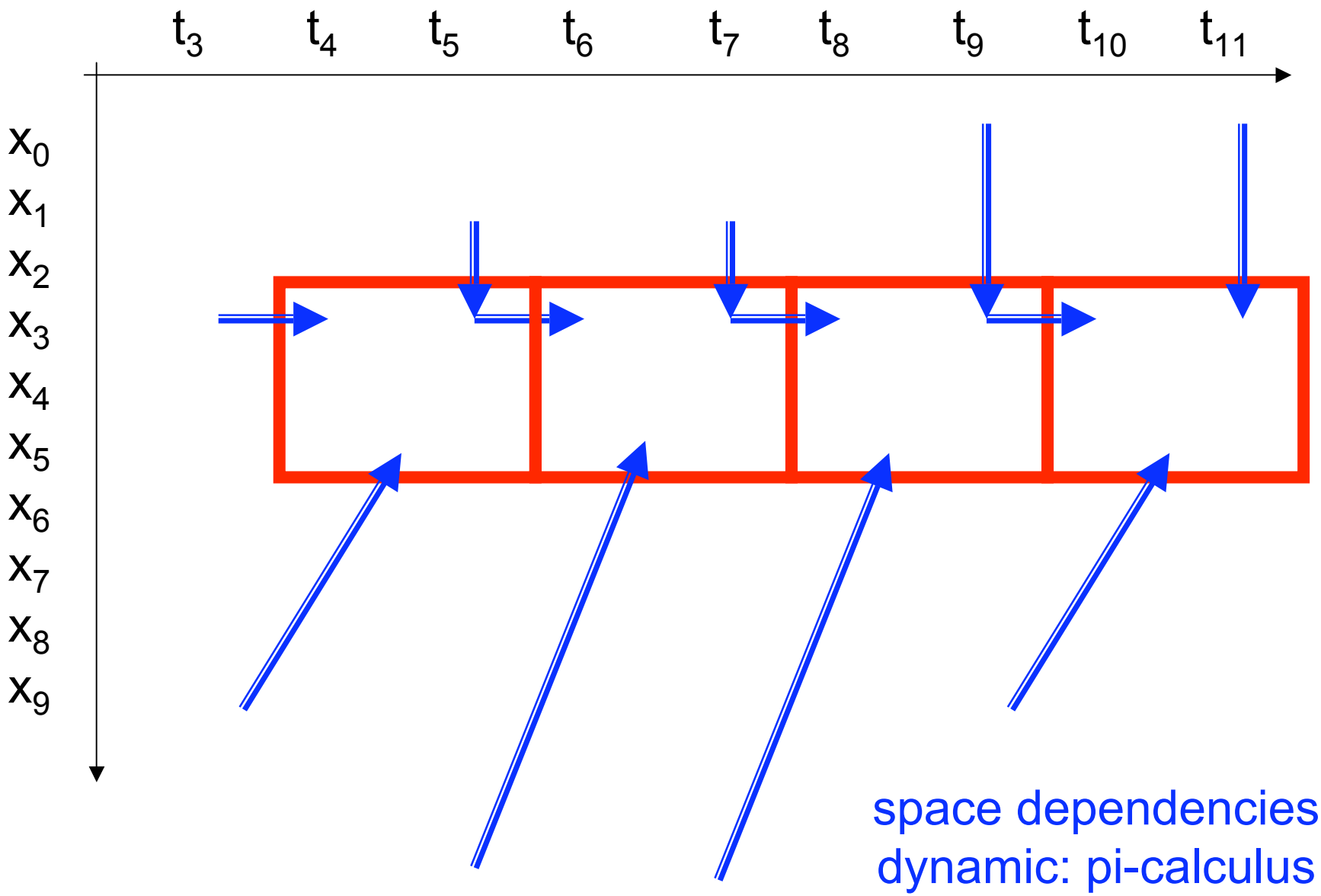


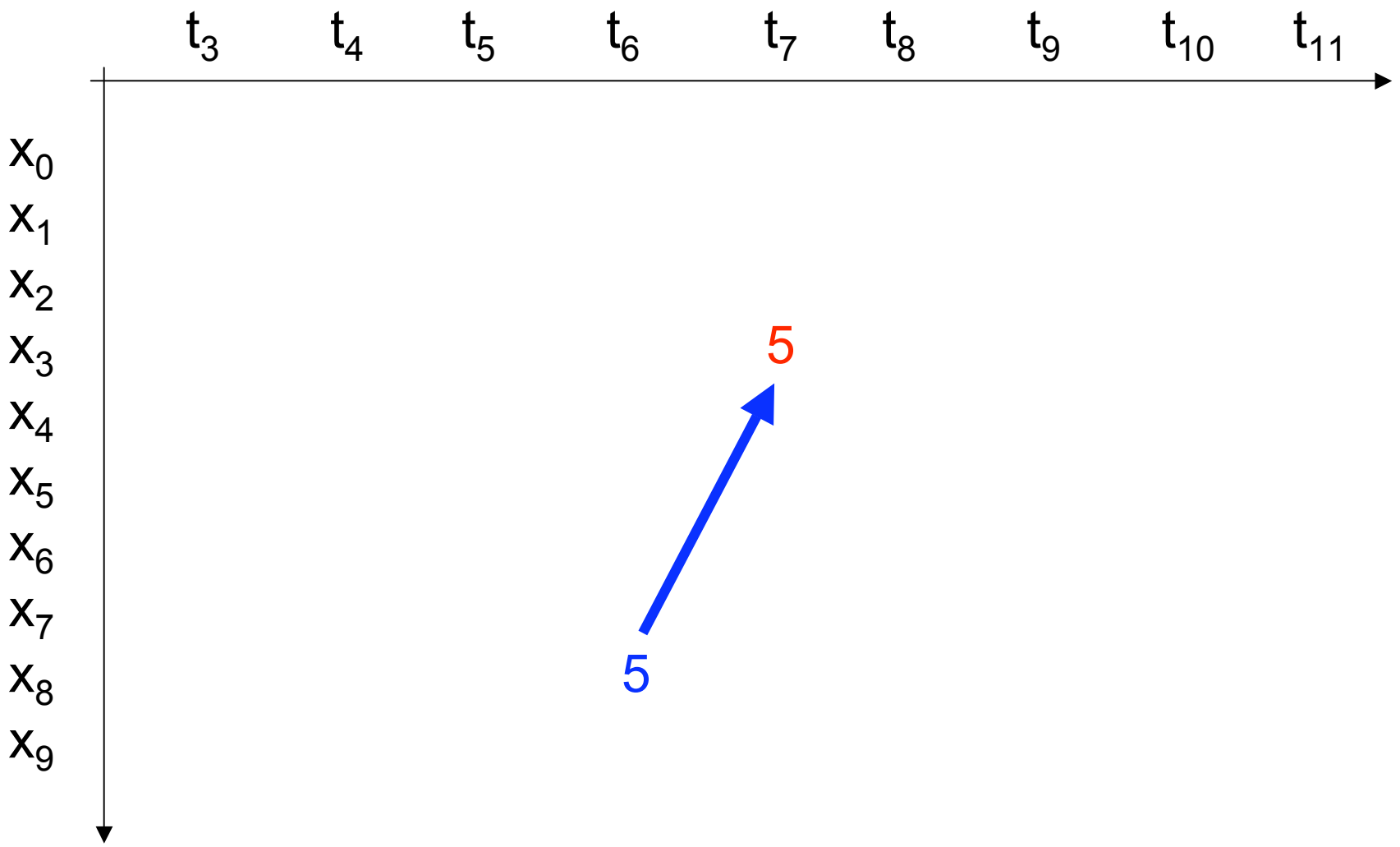
reactive module



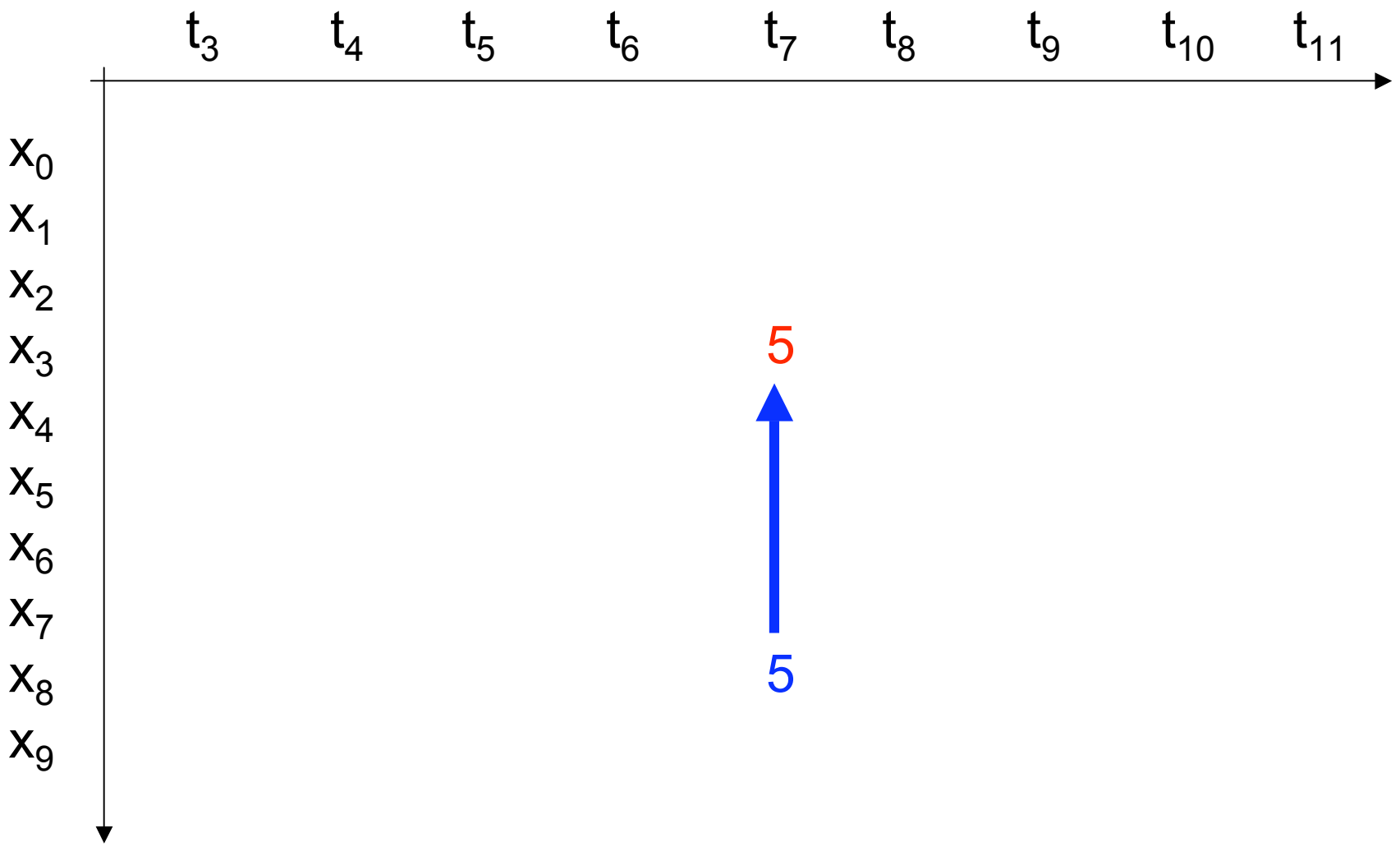
time dependencies bounded


t3
t4
t5
t6
t7
t8
t9
t10
t11
x0
x1
x2
x3
x4
x5
x6
x7
x8
x9

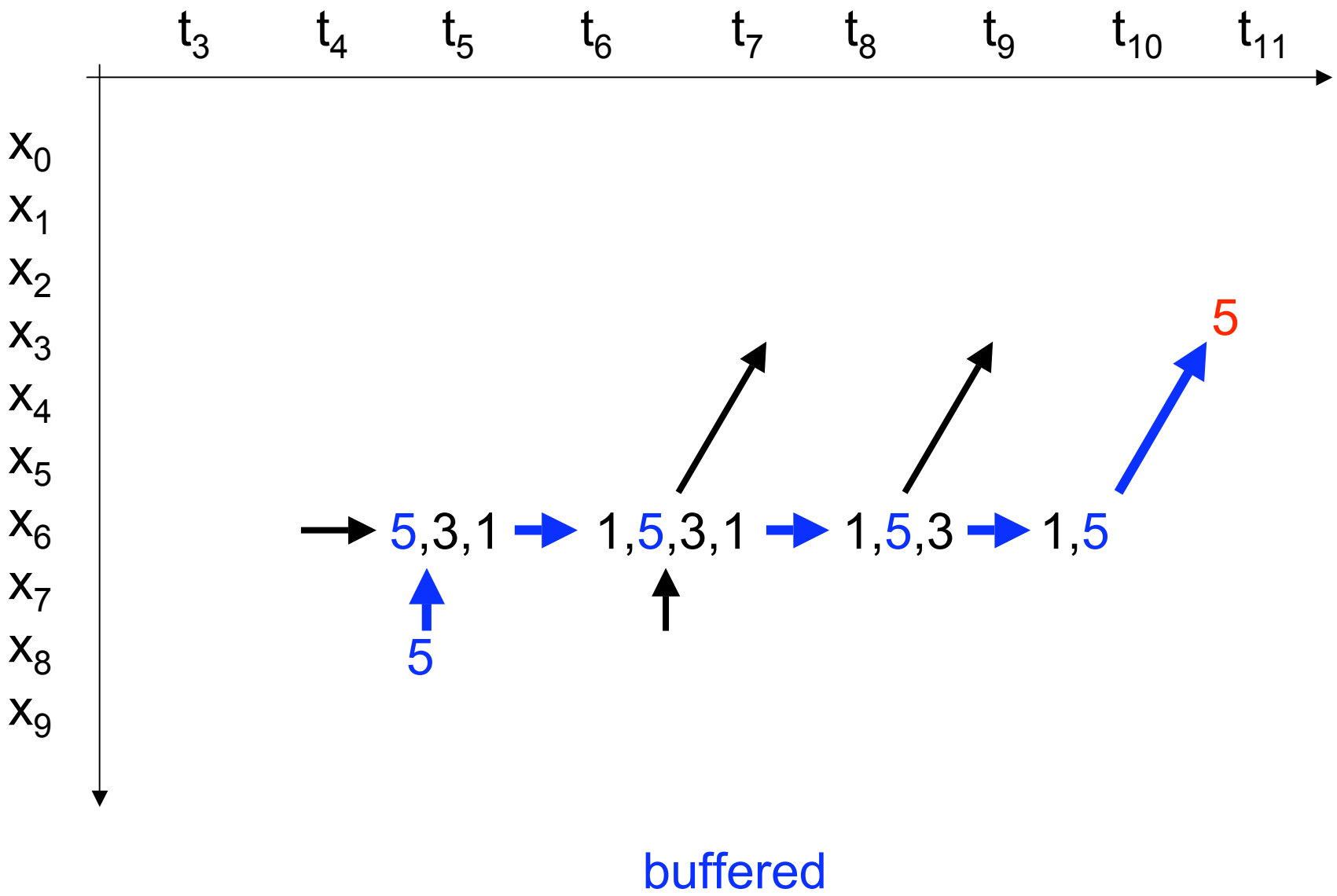

t3
t4
t5
t6
t7
t8
t9
t10
t11
x0
x1
x2
x3
x4
x5
x6
x7
x8
x9
space dependencies
dynamic: pi-calculus

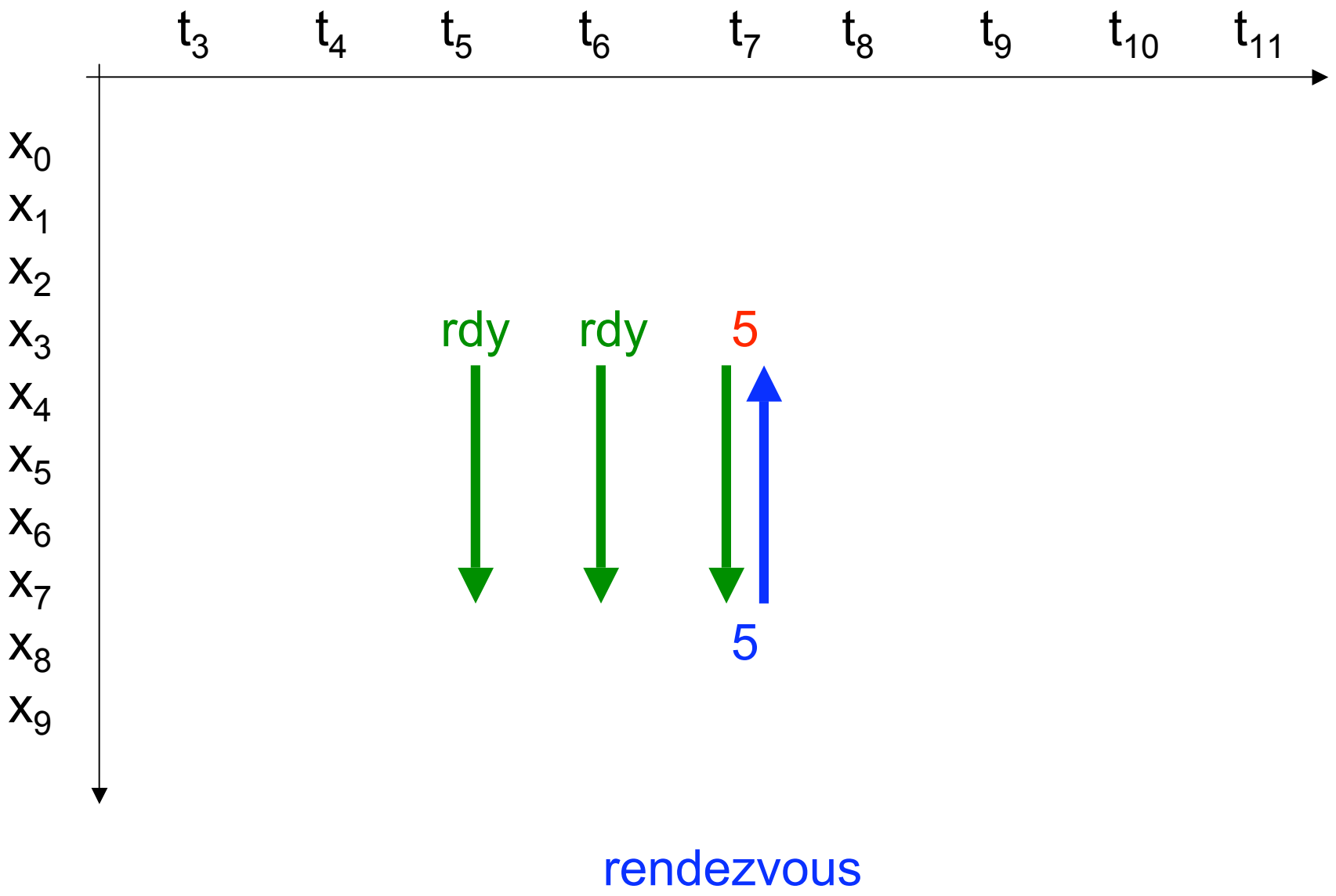


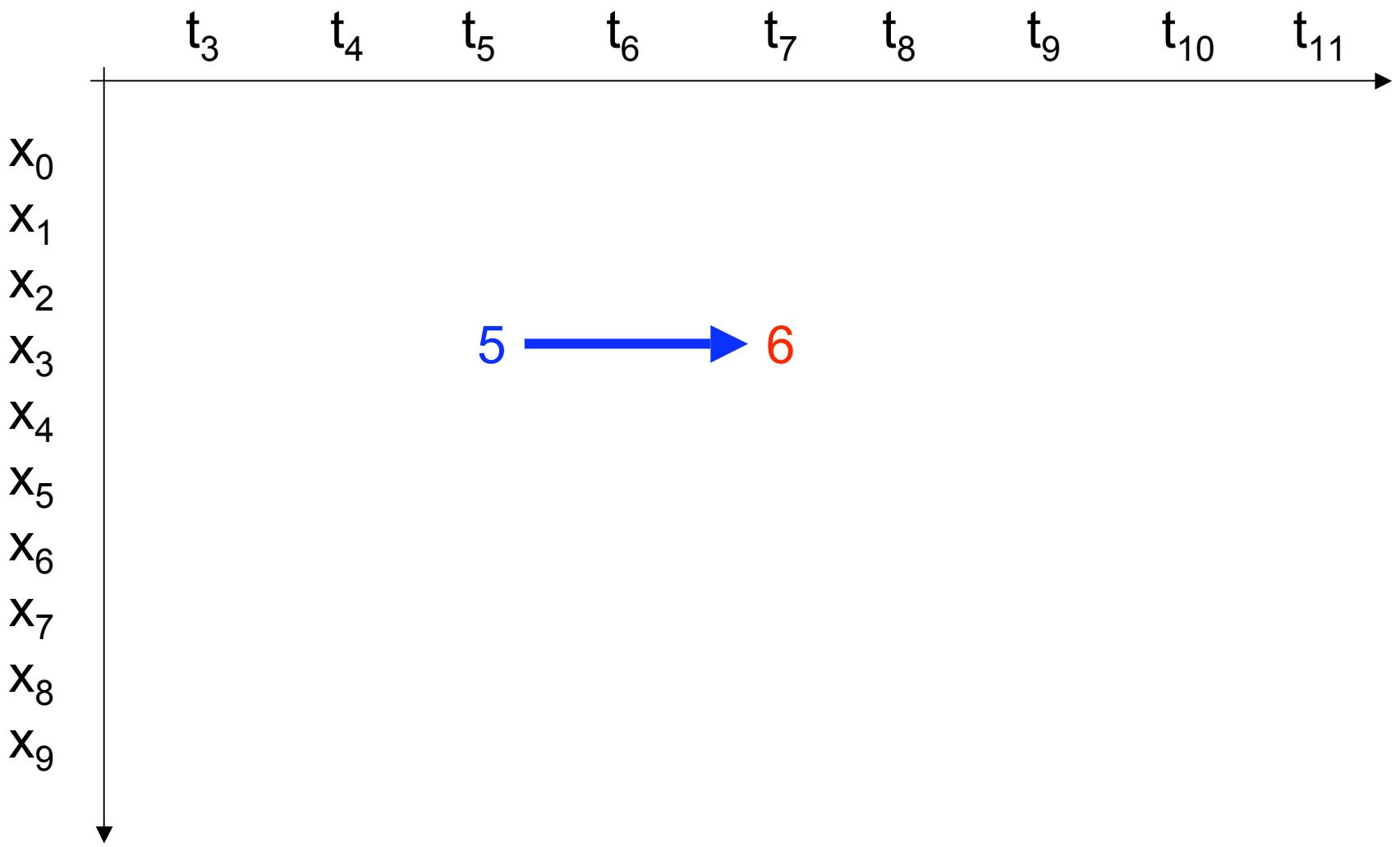
communication



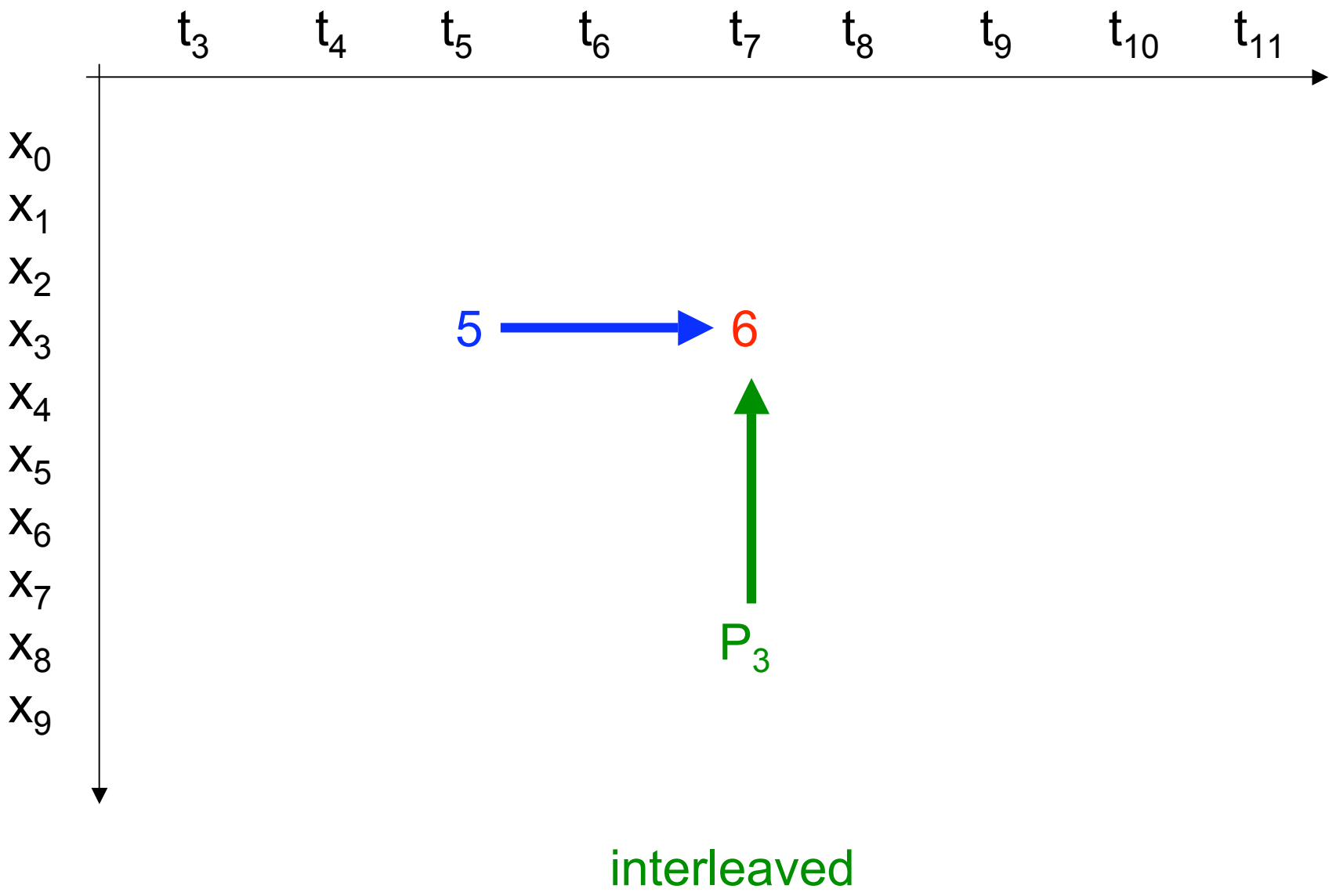
synchronous


t3
t4
t5
t6
t7
t8
t9
t10
t11
x0
x1
x2
x3
x4
x5
x6
x7
x8
x9
5,3,1
1,5,3,1
1,5,3
1,5
5
5
buffered

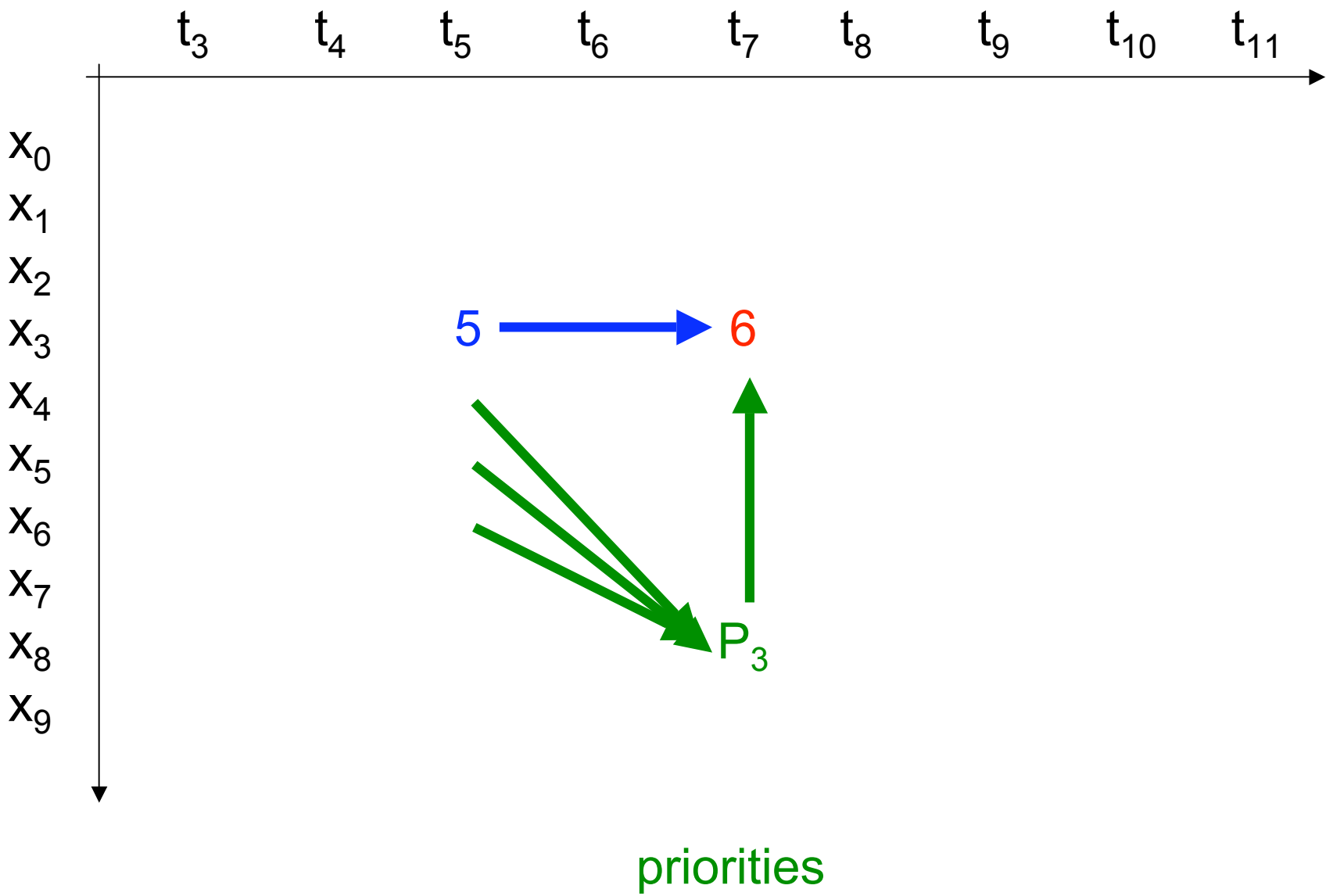

t3
t4
t5
t6
t7
t8
t9
t10
t11
x0
x1
x2
x3
x4
x5
x6
x7
x8
x9
rdy
rdy
5
5
rendezvous



computation



interleaved

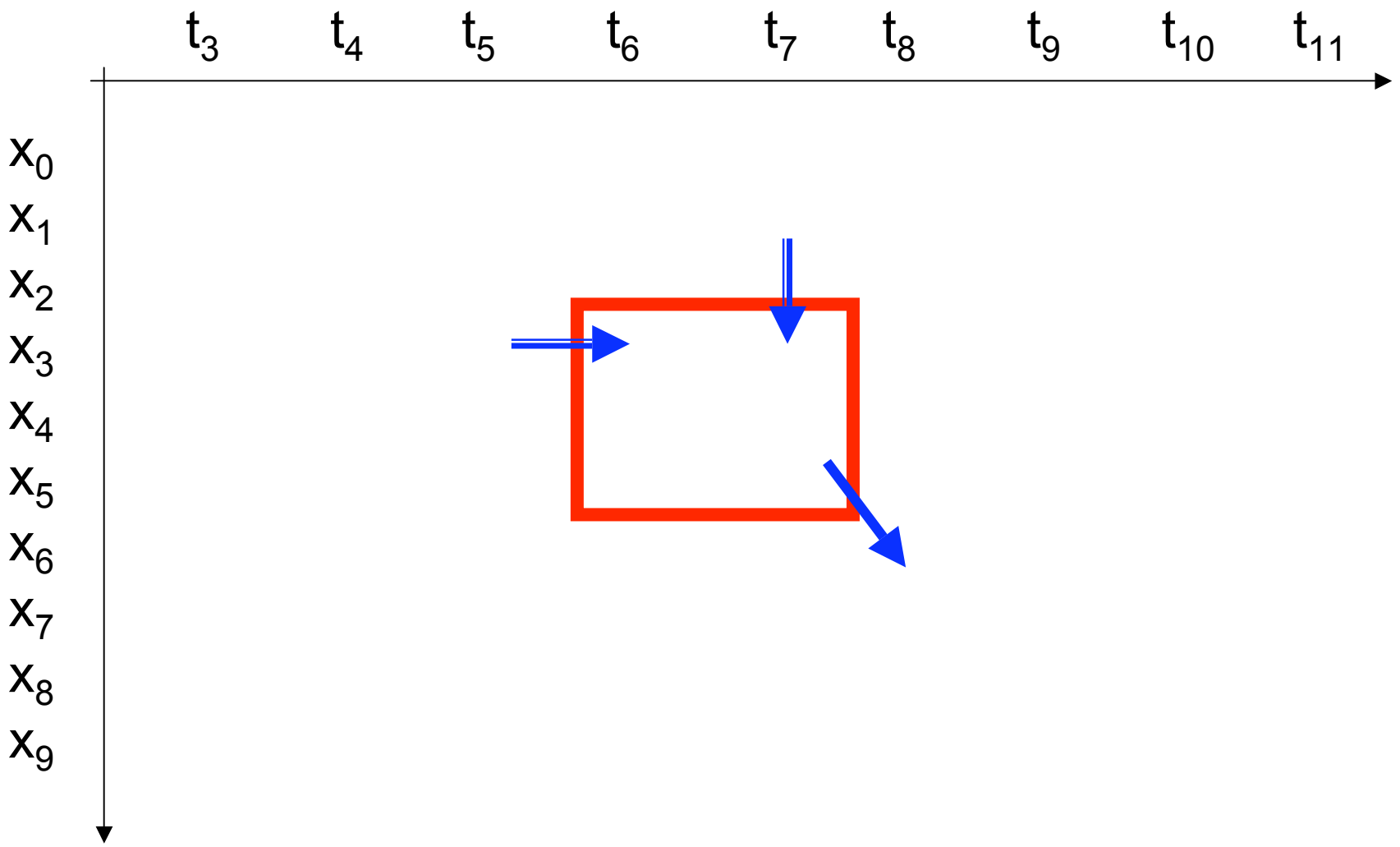

t3
t4
t5
t6
t7
t8
t9
t10
t11
x0
x1
x2
x3
x4
x5
x6
x7
x8
x9
5
6
P3
priorities

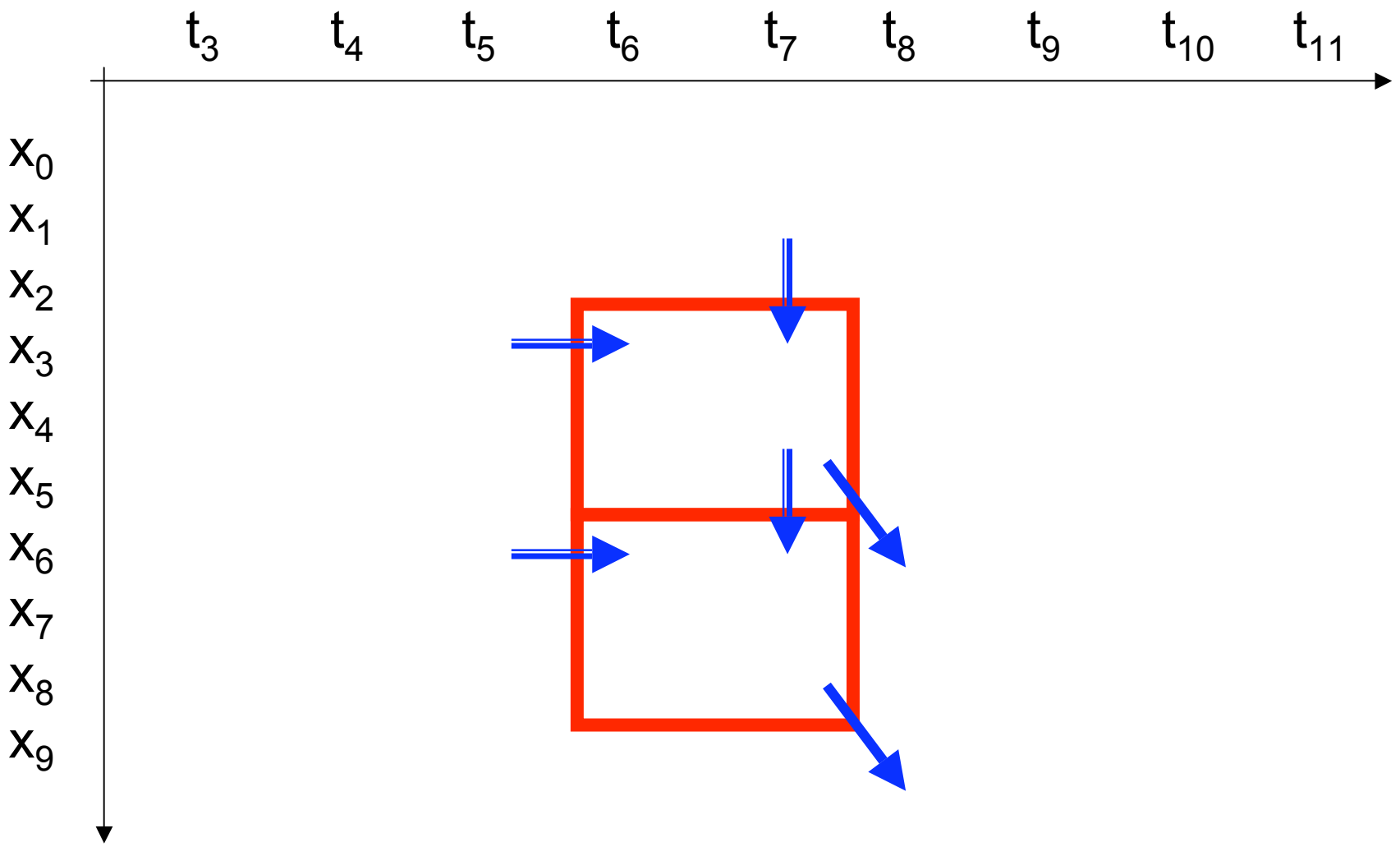
My Ideal Component Model

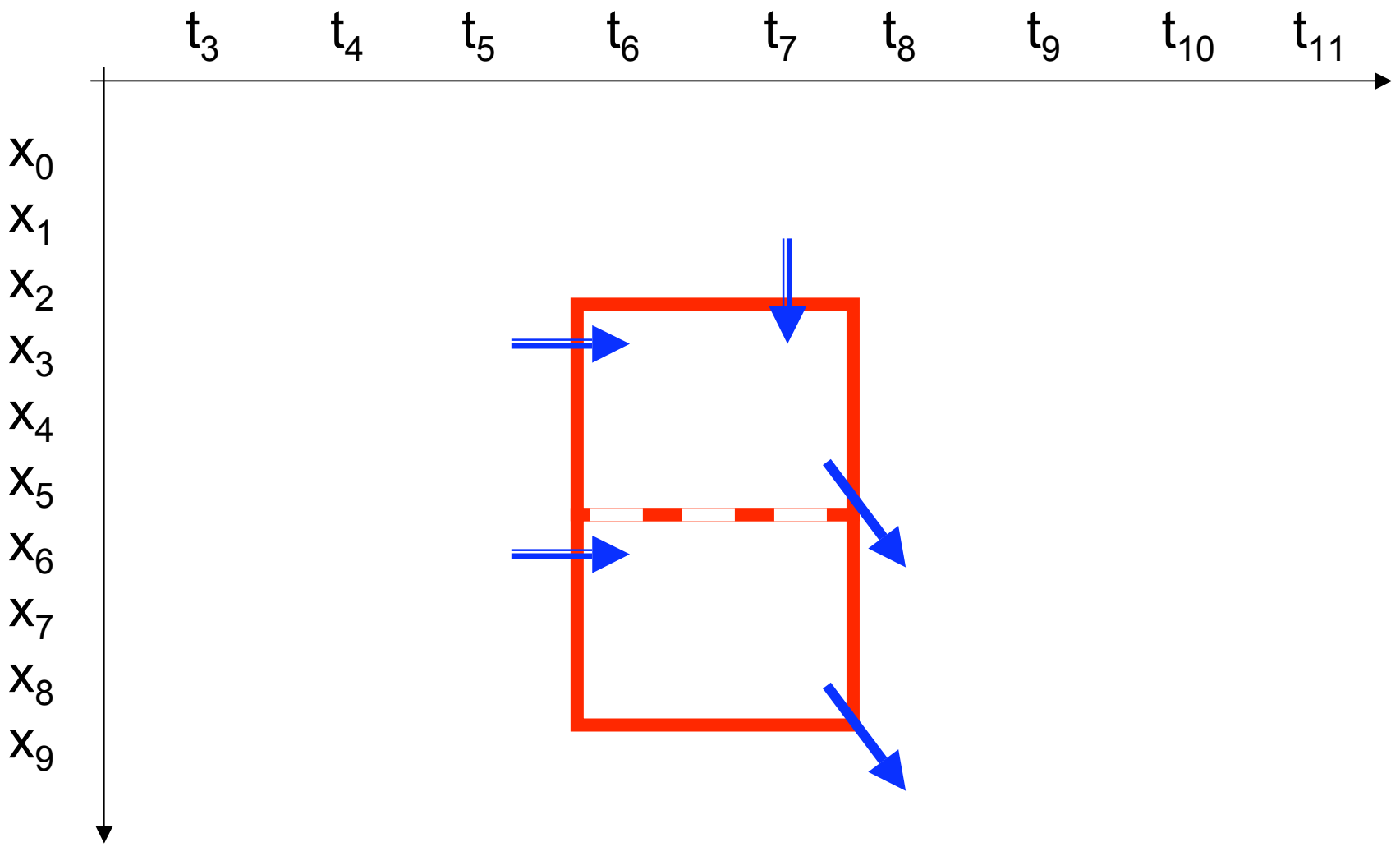
- value dependencies are the only computation and communication primitive
- component dependencies are bounded in both time and space
- component dependencies are dynamic in both time and space

My Ideal Component Model

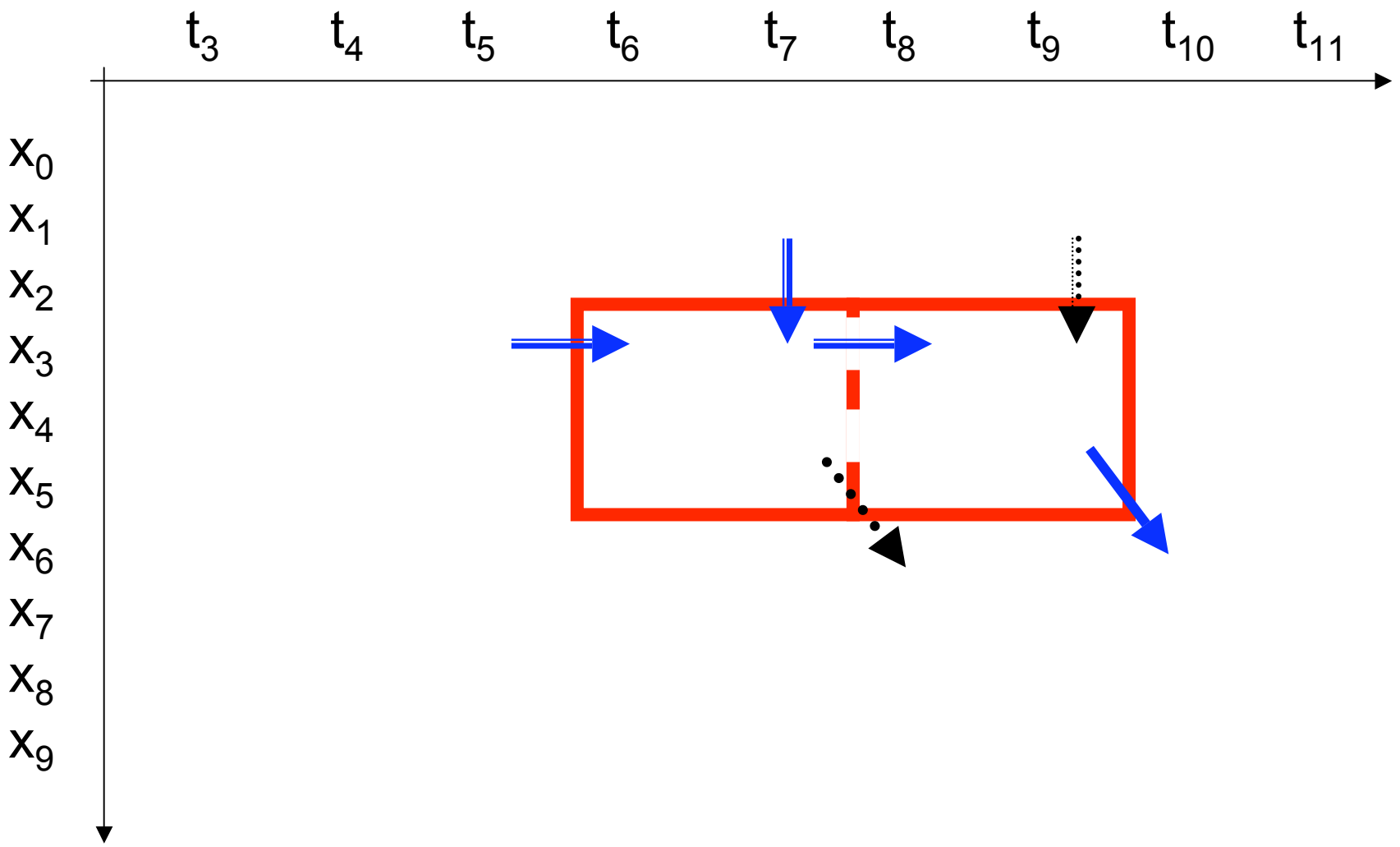
- value dependencies are the only computation and communication primitive
- component dependencies are bounded in both time and space
- component dependencies are dynamic in both time and space
- components can be aggregrated in both time and space


t3
t4
t5
t6
t7
t8
t9
t10
t11
x0
x1
x2
x3
x4
x5
x6
x7
x8
x9


t3
t4
t5
t6
t7
t8
t9
t10
t11
x0
x1
x2
x3
x4
x5
x6
x7
x8
x9



spatial aggegration



temporal aggegration:
logical execution time

My Ideal Component Model

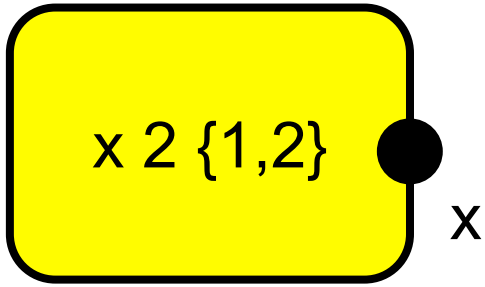
- value dependencies are the only computation and communication primitive
- component dependencies are bounded in both time and space
- component dependencies are dynamic in both time and space
- components can be aggregrated in both time and space
- components have multiple behaviors

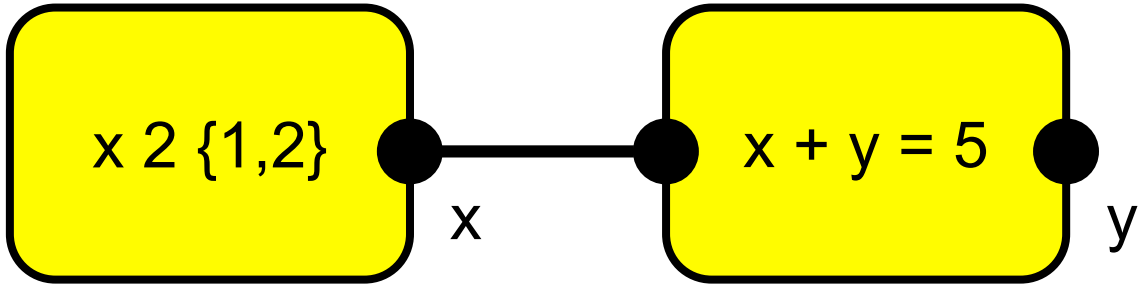
Reactive Modules [Alur & H 1996]

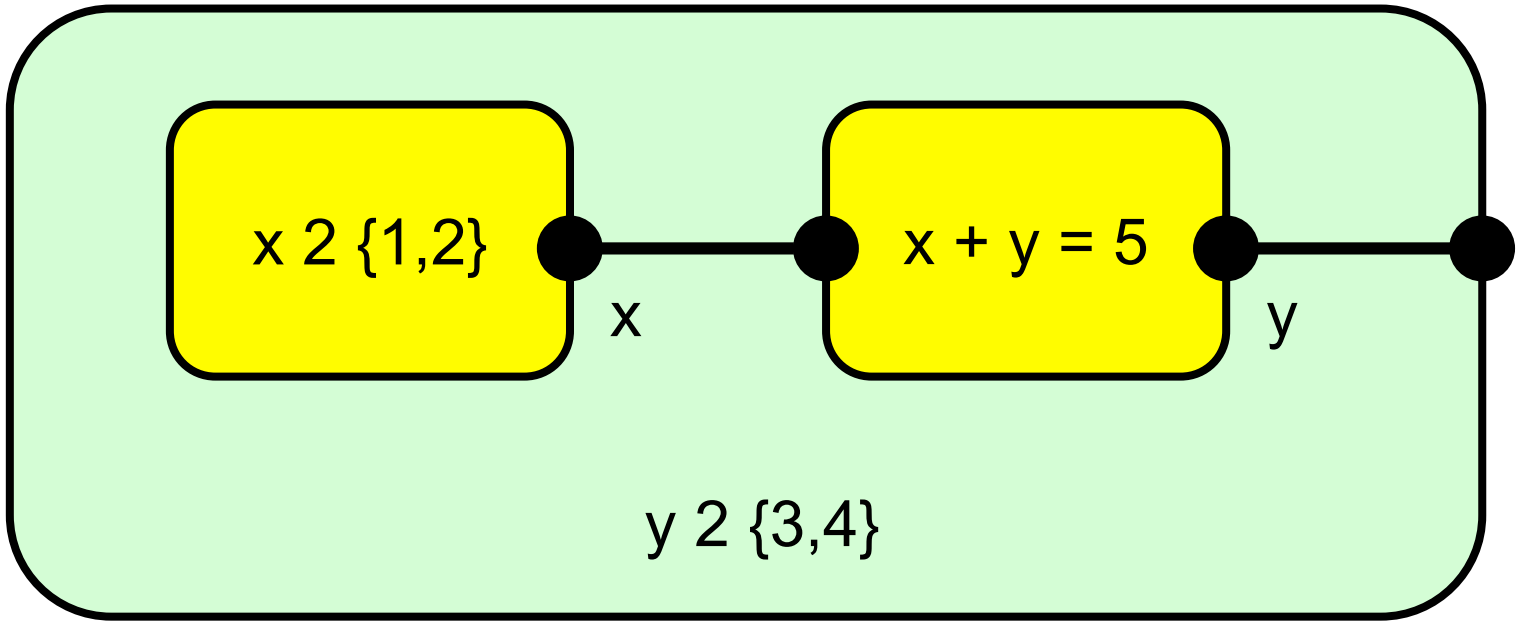
- value dependencies are the only computation and communication primitive
- component dependencies are bounded in both time and space
- component dependencies are dynamic in both time ~~and space~~
- components can be aggregrated in both time and space
- components have multiple behaviors

Some Thoughts on Interface Models

Tom Henzinger

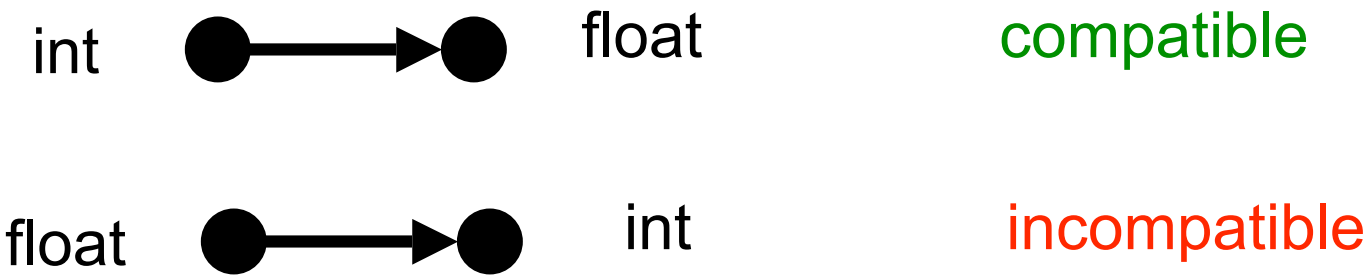

x 2 {1,2}
x


x 2 {1,2}
x
x + y = 5
y

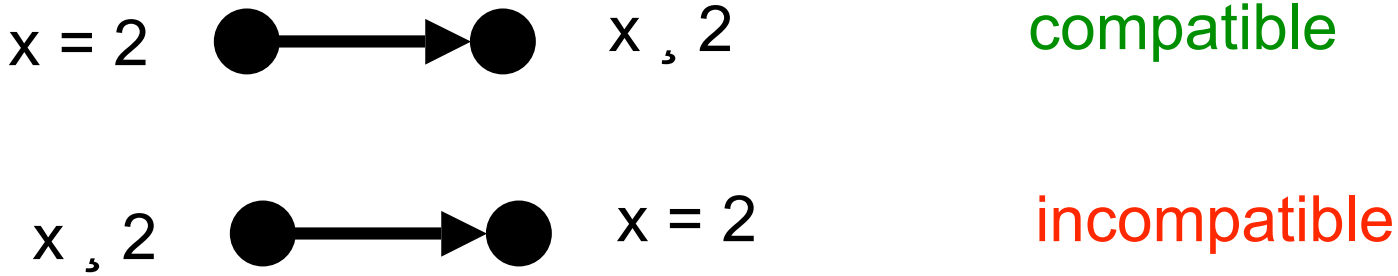


composition is conjunction:
compatibility is nonemptiness

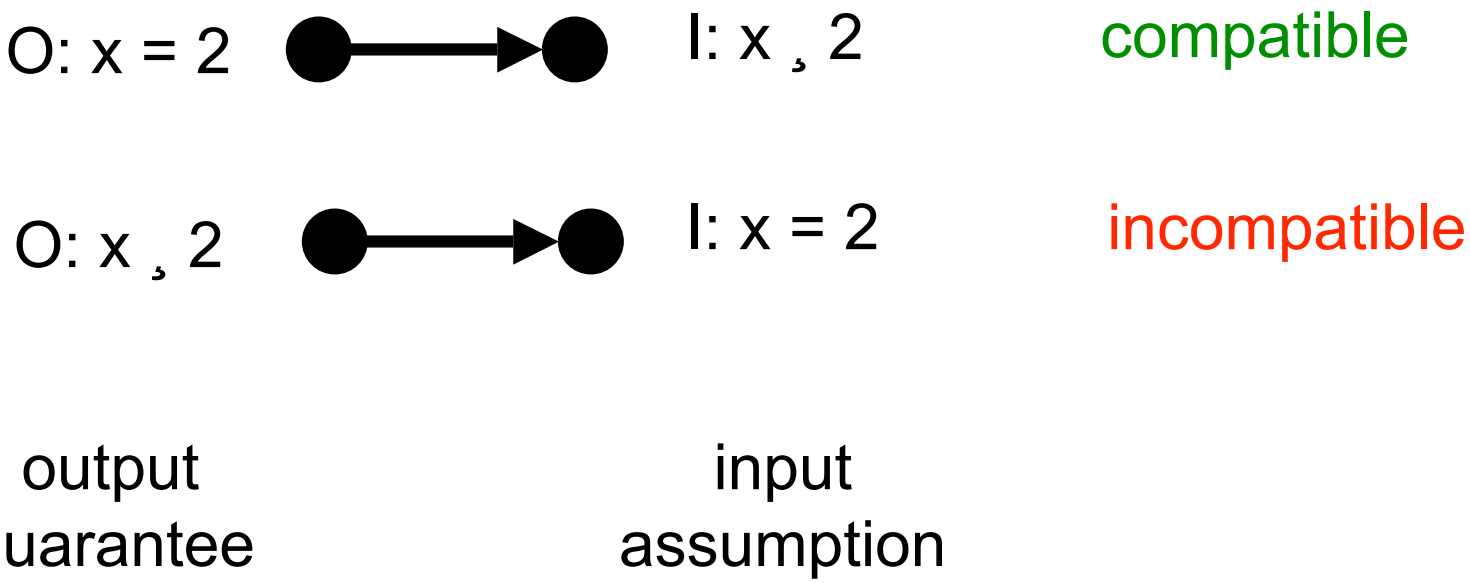
The I/O Symmetric View is Insufficient

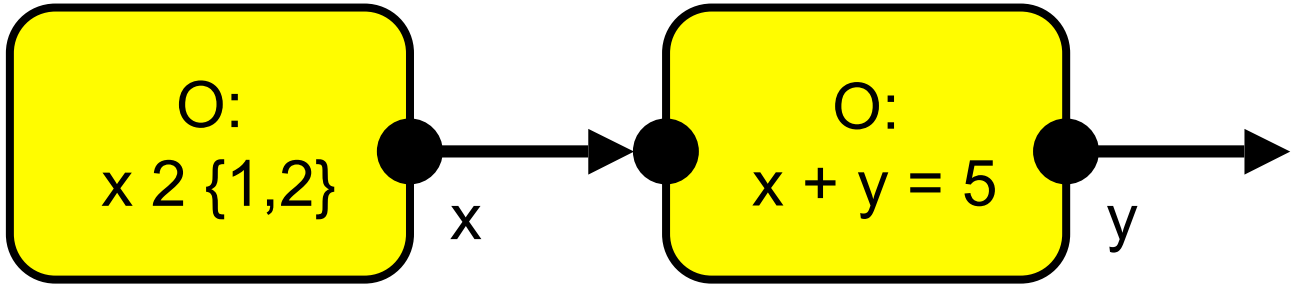


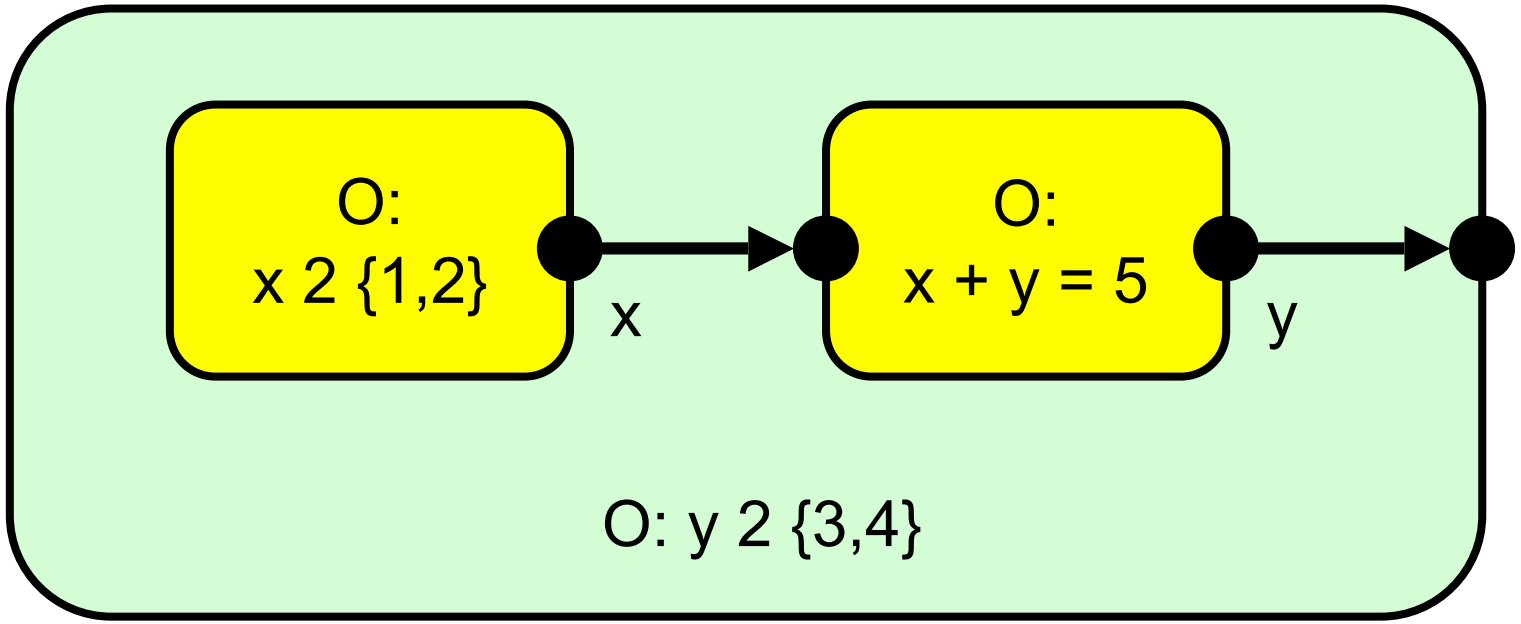
The I/O Symmetric View is Insufficient



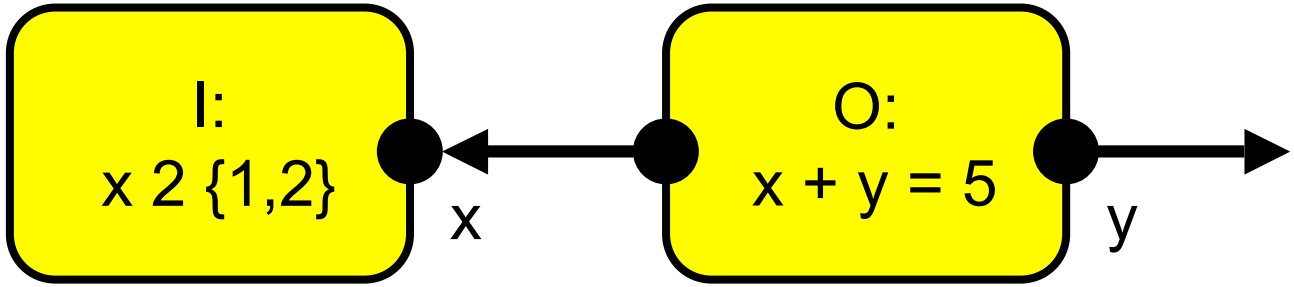
The I/O Symmetric View is Insufficient

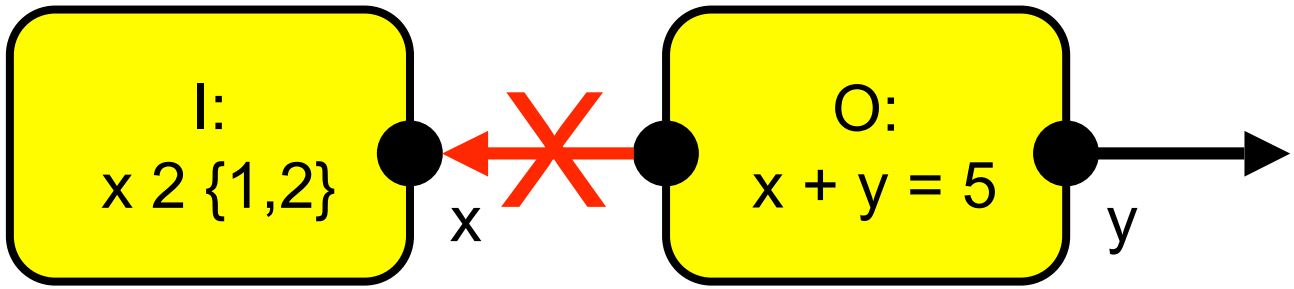



O:
x 2 {1,2}
x
O:
x + y = 5
y

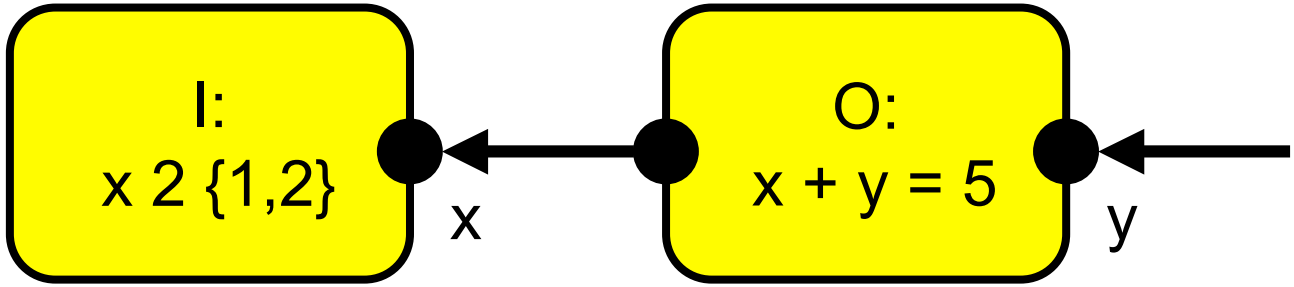


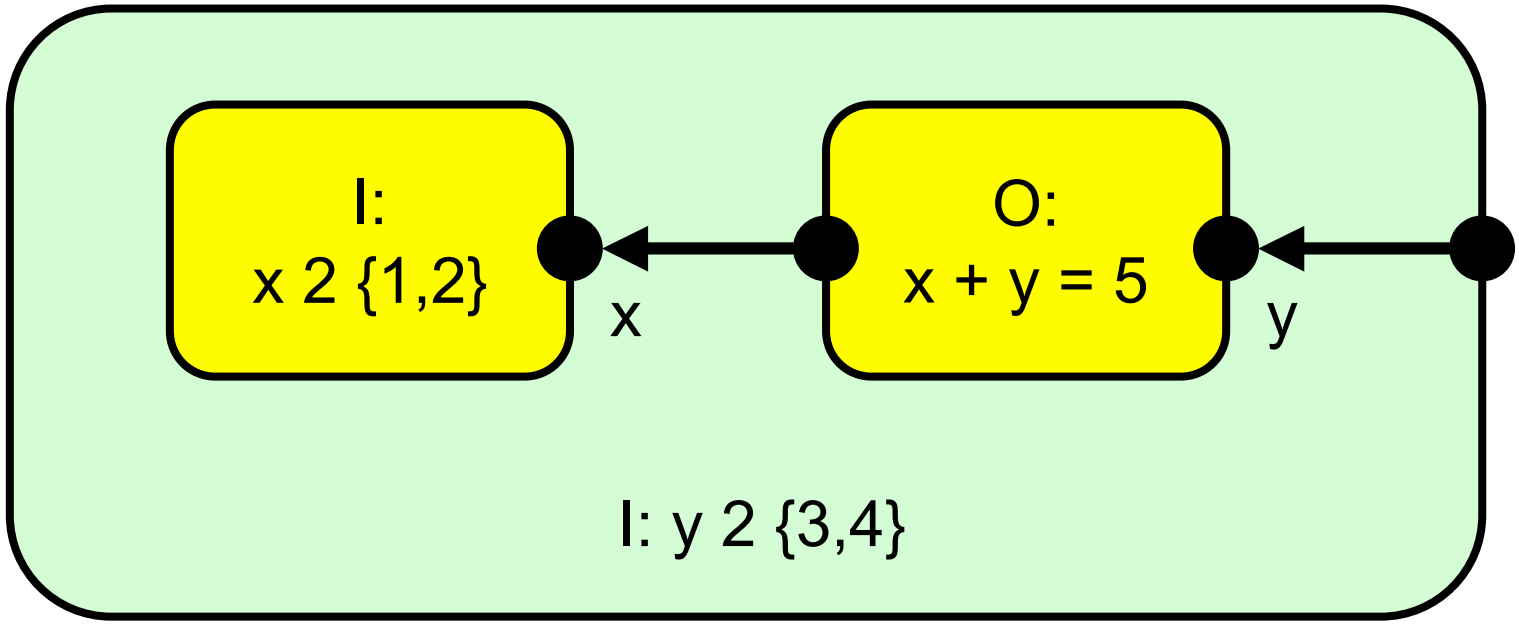
compatible


I:
x 2 {1,2}
x
O:
x + y = 5
y

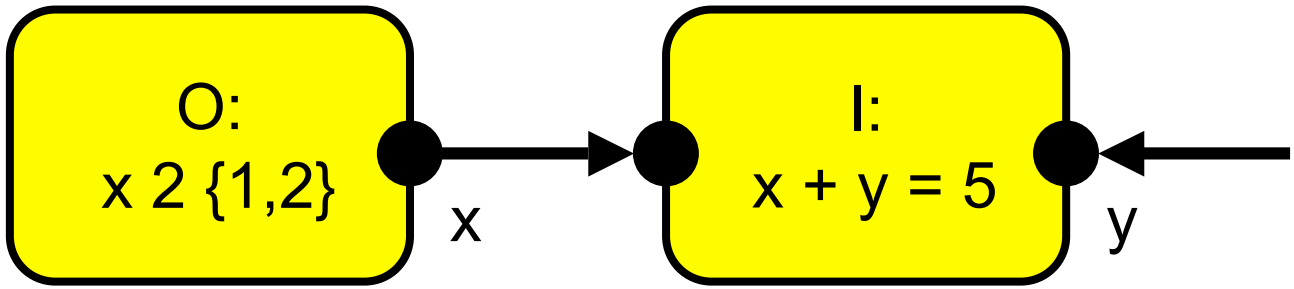


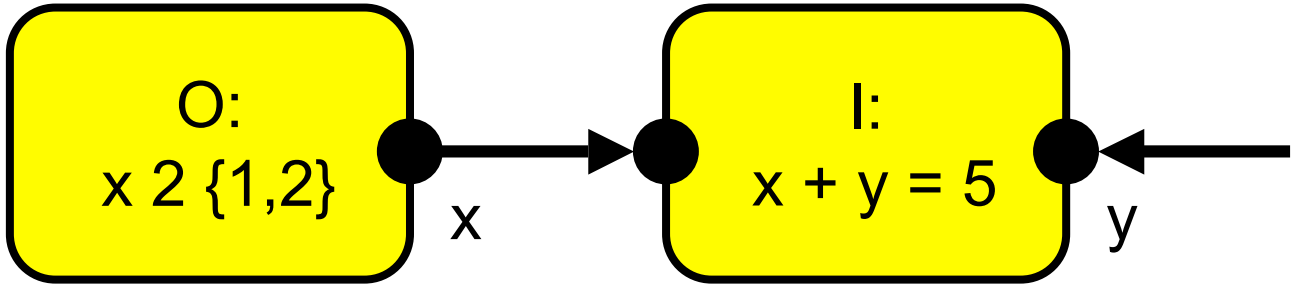
incompatible


I:
x 2 {1,2}
x
O:
x + y = 5
y

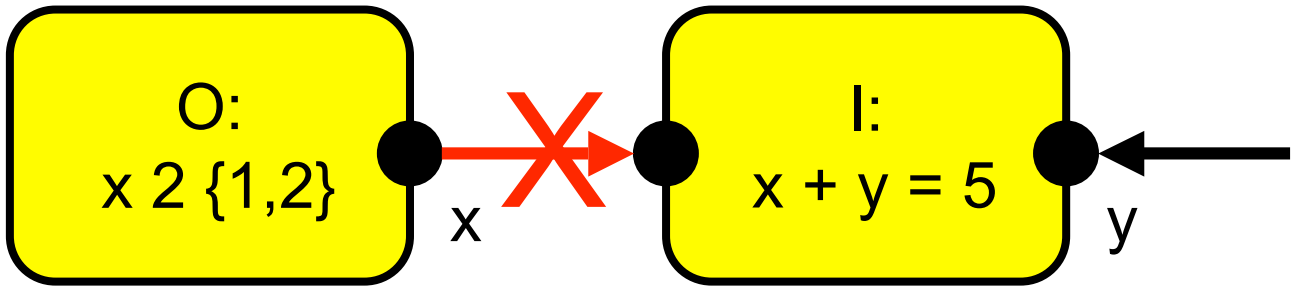


compatible:
compute weakest input assumption


O:
x 2 {1,2}
x
I:
x + y = 5
y



I: y = 3 ?



incompatible

Interfaces [de Alfaro & H 2001]

- inputs and outputs are contravariant
- composition is not conjunction: weakest input assumption is most general strategy in compatibility game (input player versus output player)
- refinement is not implication: output guarantees can be strengthened, but input assumption must be weakened
- input assumptions and output guarantees can be about timing and resource usage