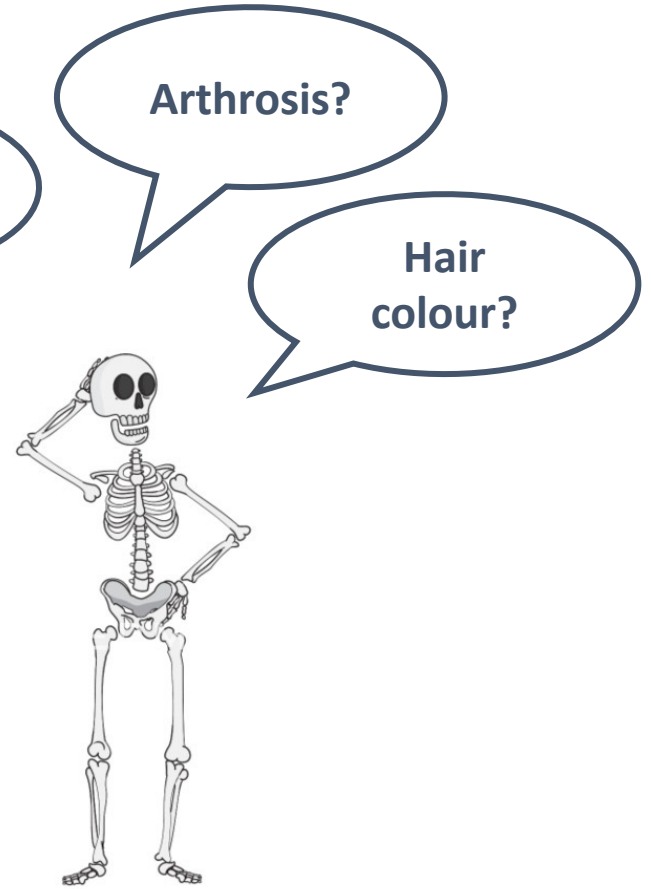
Skeleton Abstraction for Universal Temporal Properties

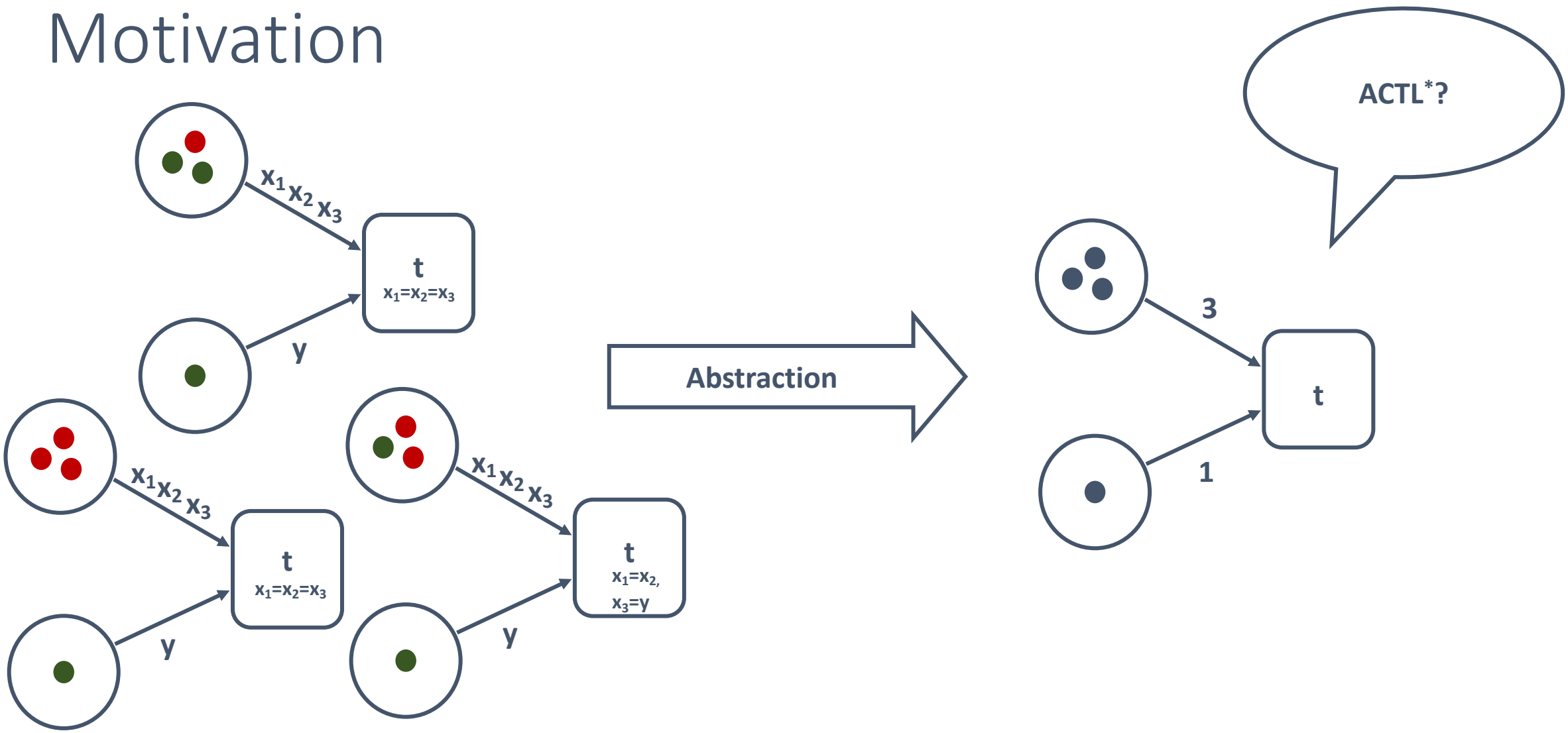
Sophie Wallner Karsten Wolf

Motivation

Abstraction



Motivation


x1 x2 x3
t
x1=x2=x3
y
x1 x2 x3
t
x1=x2=x3
y
x1 x2 x3
t
x1=x2,
x3=y
y
Abstraction
ACTL*?
3
t
1

1. Problem Analysis

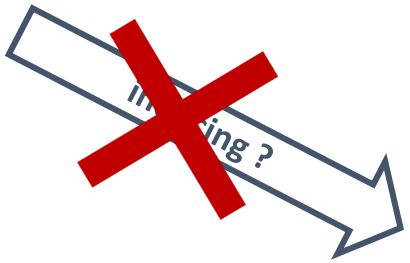
When and why do problems occur with the skeleton abstraction?

Problem Analysis

Relation on net level

net morphism

Preserves reachability



Relation on reachability graph level

abstraction relation

Related states fulfill atomic propositions equally

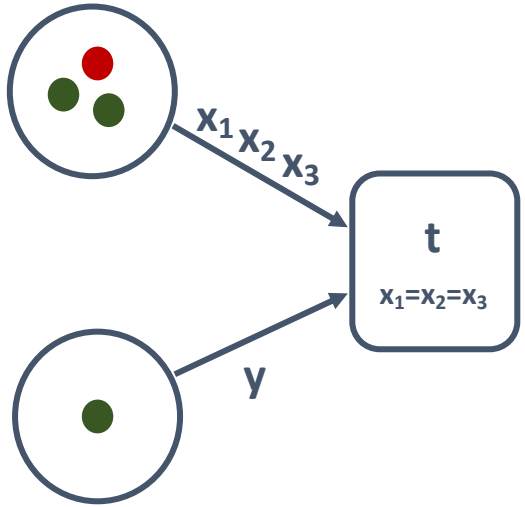
For ACTL* preservation

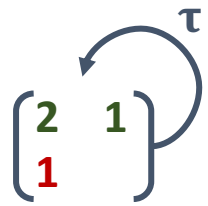
simulation relation

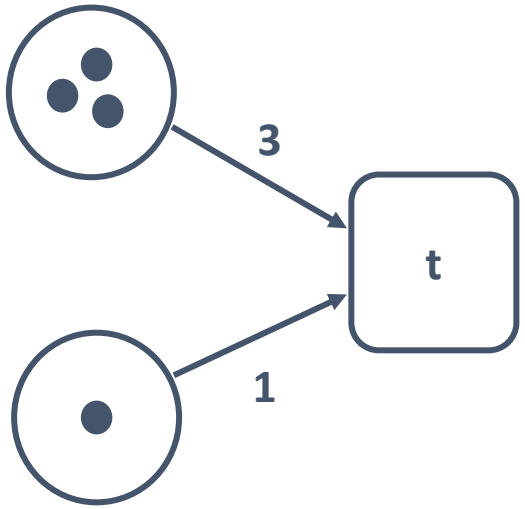
Transitions are preserved additionally

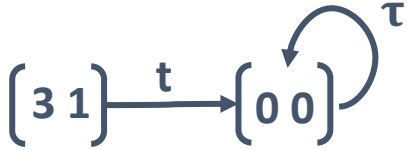
Problem Analysis

⚠ Net Morphisms do not always induce a simulation relation









2. Application Fields

For which net type or formula type we can use the skeleton abstraction?

Application of Skeleton Abstraction

Net Type	Relation	ACTL*
Deadlock-Free Deadlock-Preserving Deadlock-Injected		

Formula Type	Relation	ACTL*
Safety Properties		

Application of Skeleton Abstraction

Net Type	Relation	ACTL*
Deadlock-Free Deadlock-Preserving Deadlock-Injected	**Simulation Relation**	

Formula Type	Relation	ACTL*
Safety Properties		

Application of Skeleton Abstraction

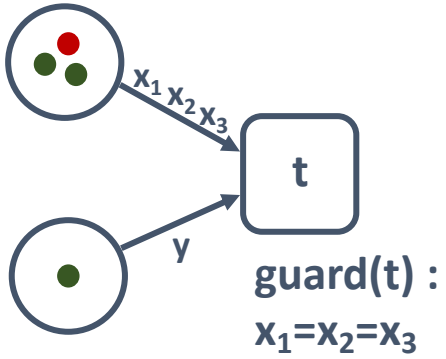
Net Type	Relation	ACTL*
Deadlock-Free Deadlock-Preserving Deadlock-Injected	**Simulation Relation**	✓

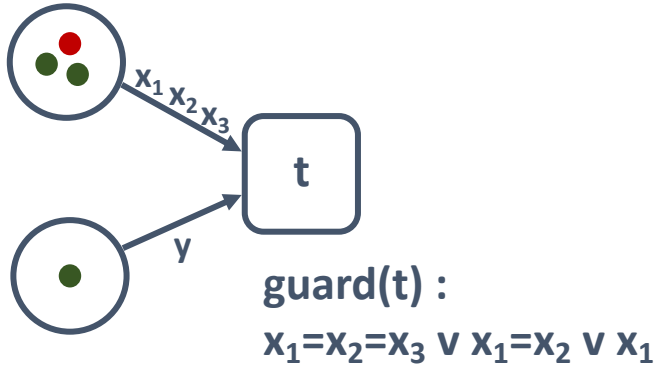
Formula Type	Relation	ACTL*
Safety Properties		

Application of Skeleton Abstraction

Net Type	Relation	ACTL*
Deadlock-Free **Deadlock-Preserving**	Simulation Relation	✓

Formula Type	Relation	ACTL*
Safety Properties		

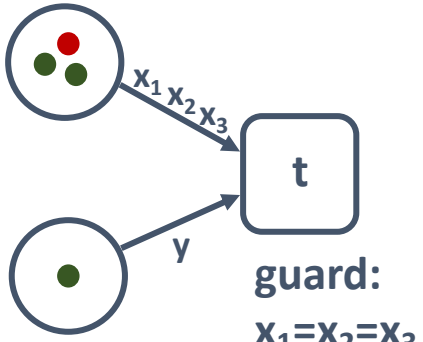


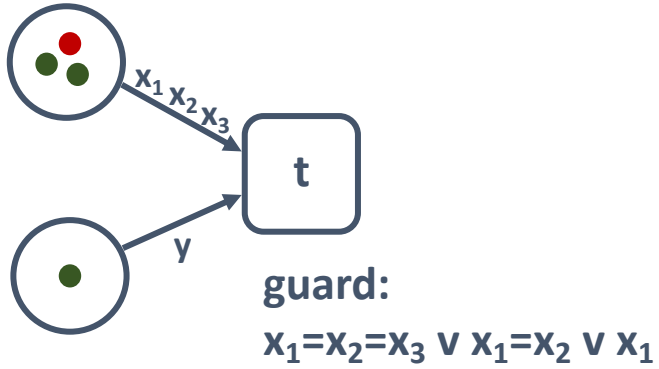


Application of Skeleton Abstraction

Net Type	Relation	ACTL*
Deadlock-Free	Simulation Relation	✓
Deadlock-Preserving	**Simulation Relation**	

Formula Type	Relation	ACTL*
Safety Properties		

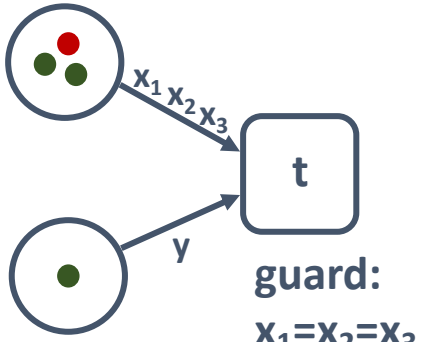


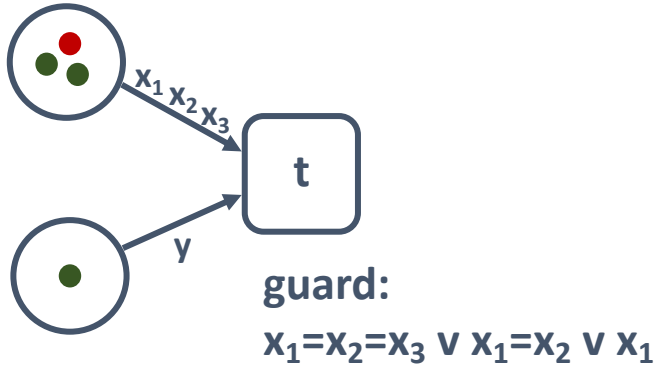


Application of Skeleton Abstraction

Net Type	Relation	ACTL*
Deadlock-Free	Simulation Relation	✓
Deadlock-Preserving	**Simulation Relation**	✓

Formula Type	Relation	ACTL*
Safety Properties		

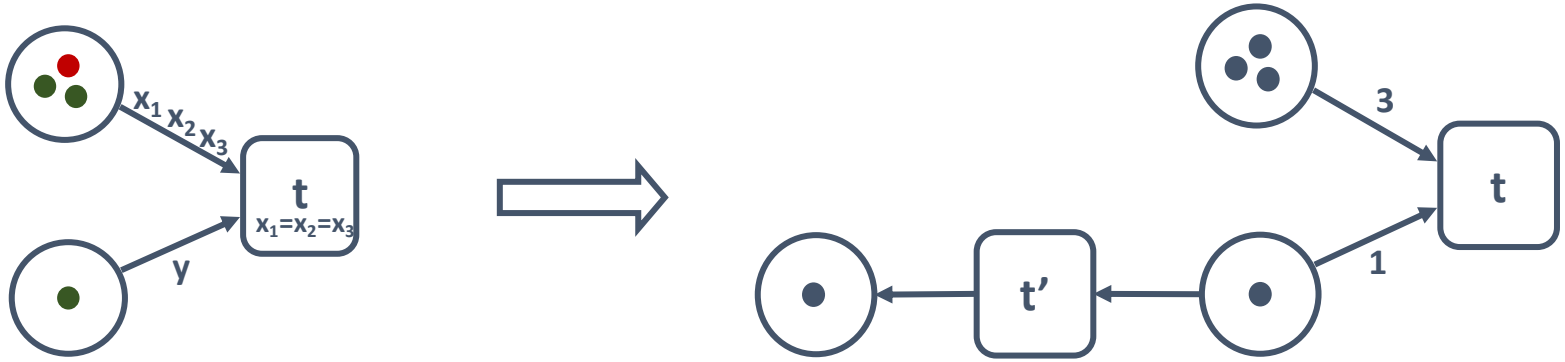




Application of Skeleton Abstraction

Net Type	Relation	ACTL*
Deadlock-Free	Simulation Relation	✓
Deadlock-Preserving	Simulation Relation	✓
Deadlock-Injected		

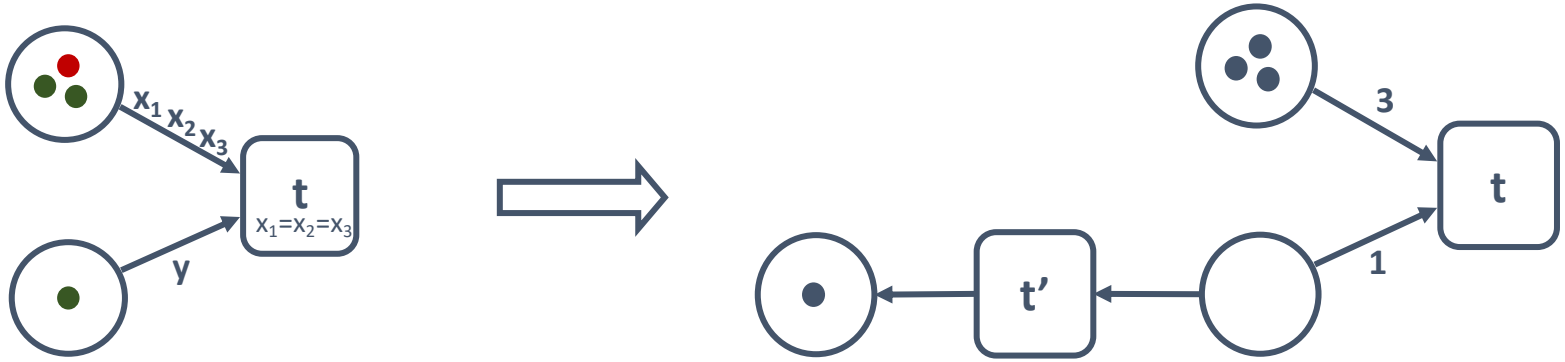
Formula Type	Relation	ACTL*
Safety Properties		



Application of Skeleton Abstraction

Net Type	Relation	ACTL*
Deadlock-Free	Simulation Relation	✓
Deadlock-Preserving	Simulation Relation	✓
Deadlock-Injected	**Stuttering Simulation Relation**	

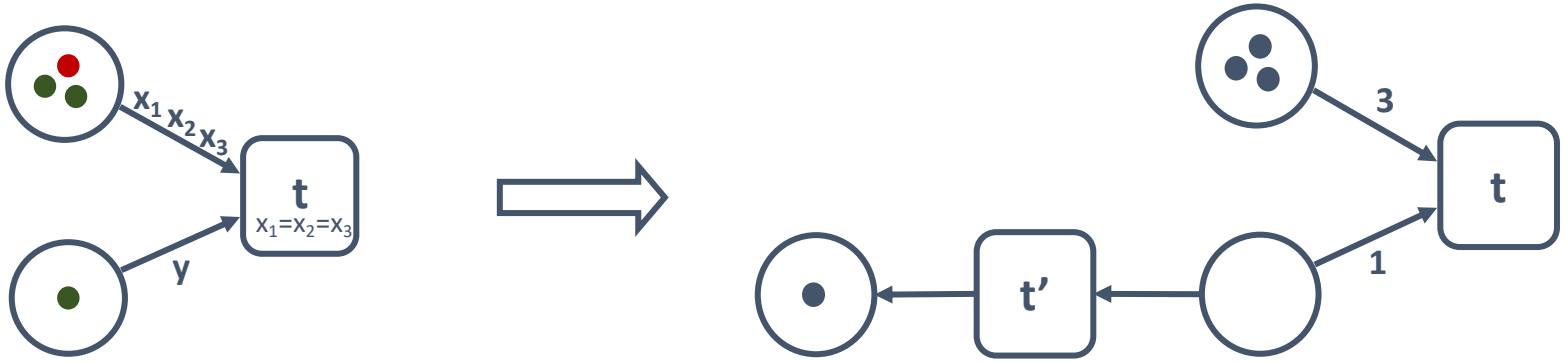
Formula Type	Relation	ACTL*
Safety Properties		



Application of Skeleton Abstraction

Net Type	Relation	ACTL*
Deadlock-Free	Simulation Relation	✓
Deadlock-Preserving	Simulation Relation	✓
Deadlock-Injected	**Stuttering Simulation Relation**	**ACTL*$_X$**

Formula Type	Relation	ACTL*
Safety Properties		



Application of Skeleton Abstraction

Net Type	Relation	ACTL*
Deadlock-Free	Simulation Relation	✓
Deadlock-Preserving	Simulation Relation	✓
Deadlock-Injected	Stuttering Simulation Relation	$ACTL^*_X$

Formula Type	Relation	ACTL*
Safety Properties		

Application of Skeleton Abstraction

Net Type	Relation	ACTL*
Deadlock-Free	Simulation Relation	✓
Deadlock-Preserving	Simulation Relation	✓
Deadlock-Injected	Stuttering Simulation Relation	$ACTL^*_X$

Formula Type	Relation	ACTL*
Safety Properties	**Abstraction Relation**	

Application of Skeleton Abstraction

Net Type	Relation	ACTL*
Deadlock-Free	Simulation Relation	✓
Deadlock-Preserving	Simulation Relation	✓
Deadlock-Injected	Stuttering Simulation Relation	$ACTL^*_X$

Formula Type	Relation	ACTL*
Safety Properties	**Abstraction Relation**	**✓ (safety only)**

Application of Skeleton Abstraction

Net Type	Relation	ACTL*
Deadlock-Free	Simulation Relation	✓
Deadlock-Preserving	Simulation Relation	✓
Deadlock-Injected	Stuttering Simulation Relation	$ACTL^*_X$

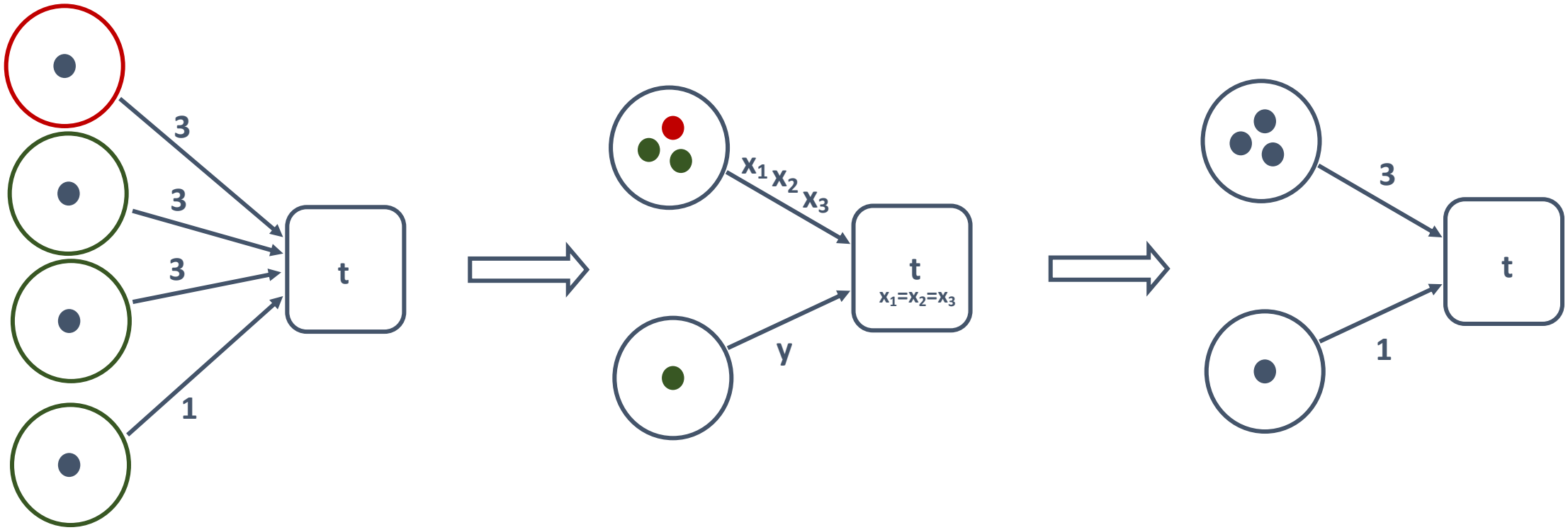
Formula Type	Relation	ACTL*
Safety Properties	Abstraction Relation	✓ (safety only)

3. Expansion to P/T-nets

Can we extend the scope of application?

Expanding Skeleton Abstraction to P/T nets

→ Folding procedure for **uniform** P/T nets



4. Performance Analysis

Is the skeleton abstraction a valuable addition for our model checking portfolio?

Performance of Skeleton Abstraction

- Benchmark: Model Checking Contest 2019
- when applicable, responsible for every third result
- solves yet unsolved task (~ 200 new results)

most suitable:

→ coloured nets with huge unfolding

→ regular, symmetrical P/T nets and formulas

Conclusion

1. When and why do problems occur with the skeleton abstraction?
   → Simulation relation not always induced
   → Beware of deadlock issue!

2. For which net or formula type we can use the skeleton abstraction?
   → Situative Application for deadlock-free/deadlock-preserving nets, otherwise inject deadlocks to skeleton
   →Special case: safety properties

3. Can we extend the scope of application?
   → Folding procedure makes skeleton applicable for uniform P/T nets

4. Is the skeleton abstraction a valuable addition for our model checking portfolio?
   → Yes!