Trace Abstraction vs. IMPACT

A Study of Software Model Checkers based on Interpolation

Alexander Nutz

Universität Freiburg, Institut für Informatik, Lehrstuhl für Softwaretechnik

21. September 2010

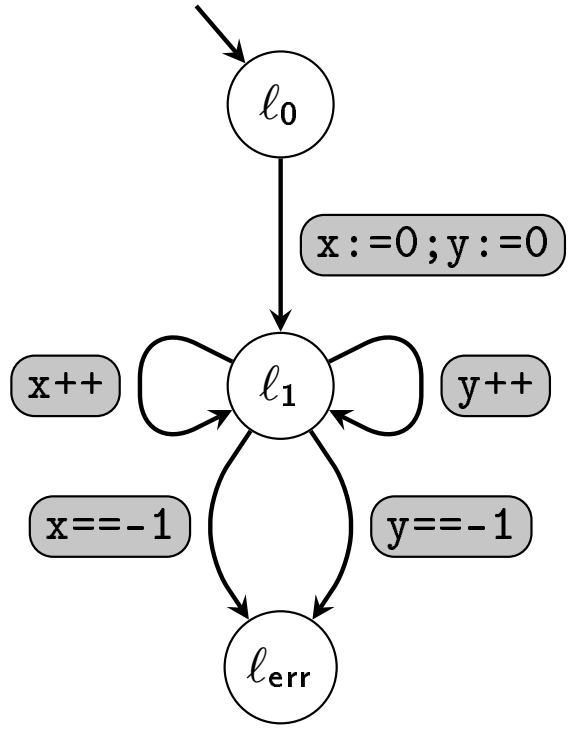
Fragestellung

IMPACT und Trace Abstraction: zwei auf Interpolation basierende Software Model Checker

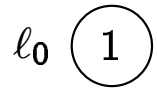
- Frage nach der Vergleichbarkeit von IMPACT und Trace Abstraction
- Suche nach geeigneten Vergleichskriterien
- Ist das eine durch das andere ausdrückbar/eine Variante oder Teilmenge des anderen?

Vorstellung der beiden Verfahren - Beispielprogramm

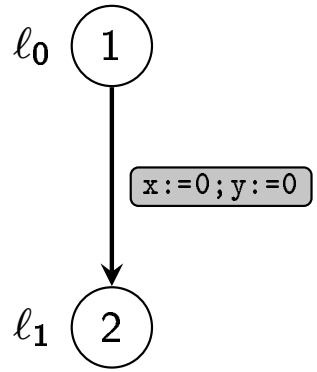
```
x:=0
y:=0
while(*) {
 if(+)
  x++
 else
  y++
}
assert(x!=-1)
assert(y!=-1)
```



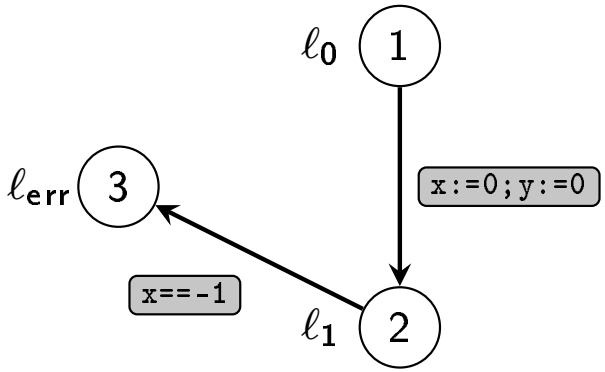
Kurzvorstellung IMPACT



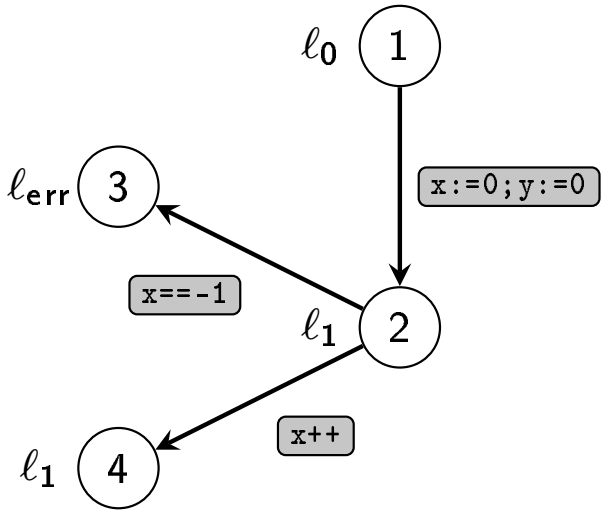
Kurzvorstellung IMPACT



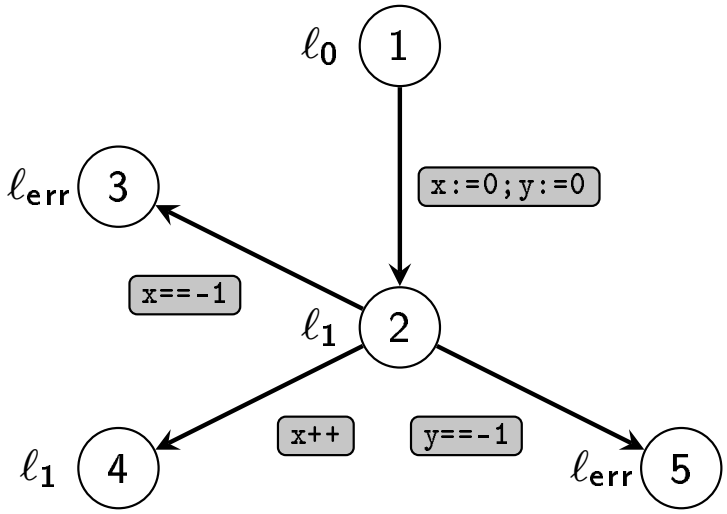
Kurzvorstellung IMPACT



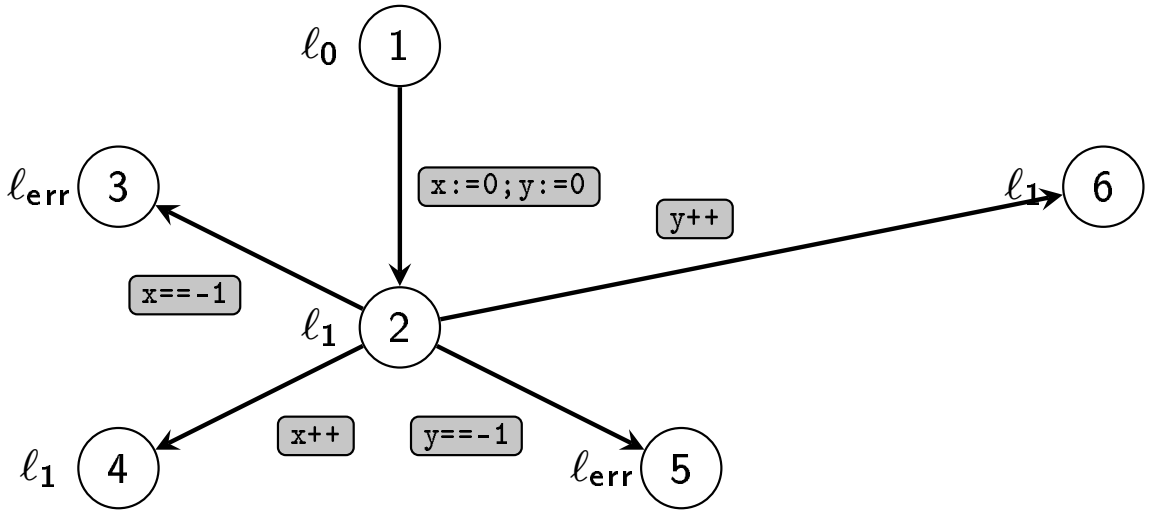
Kurzvorstellung IMPACT



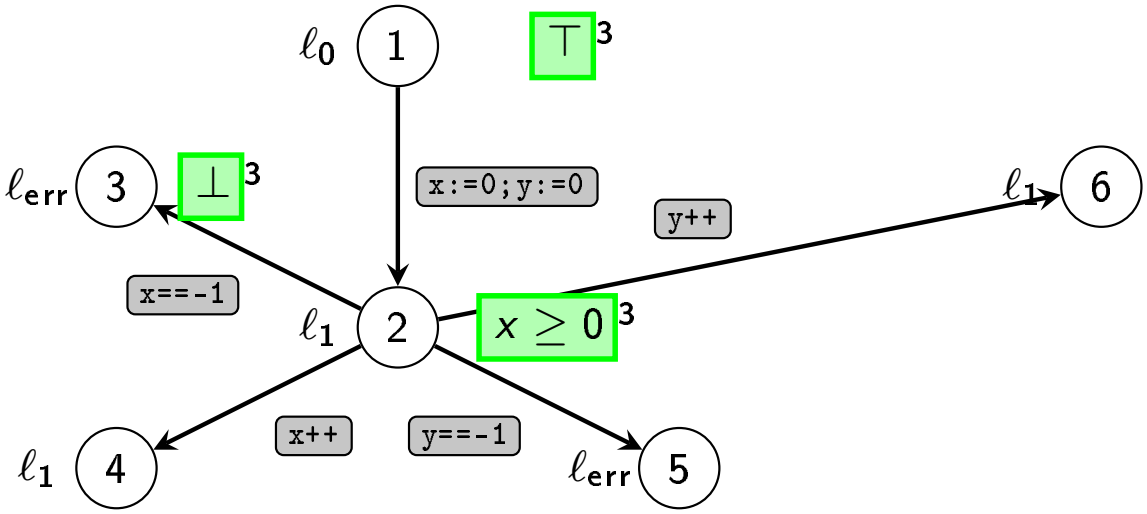
Kurzvorstellung IMPACT



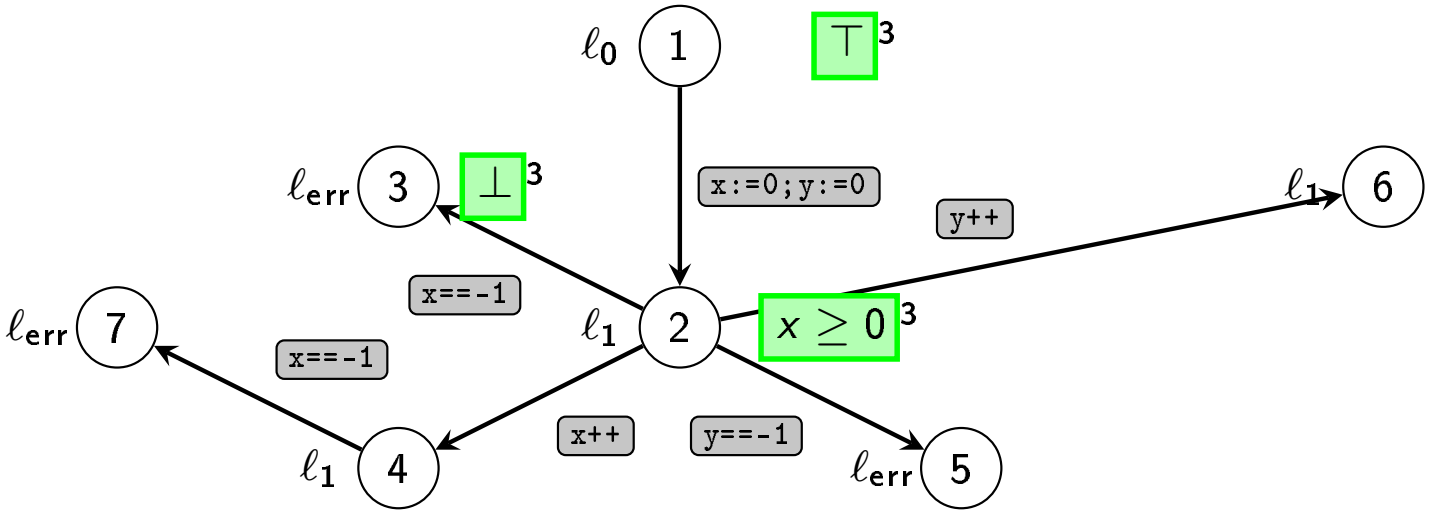
Kurzvorstellung IMPACT



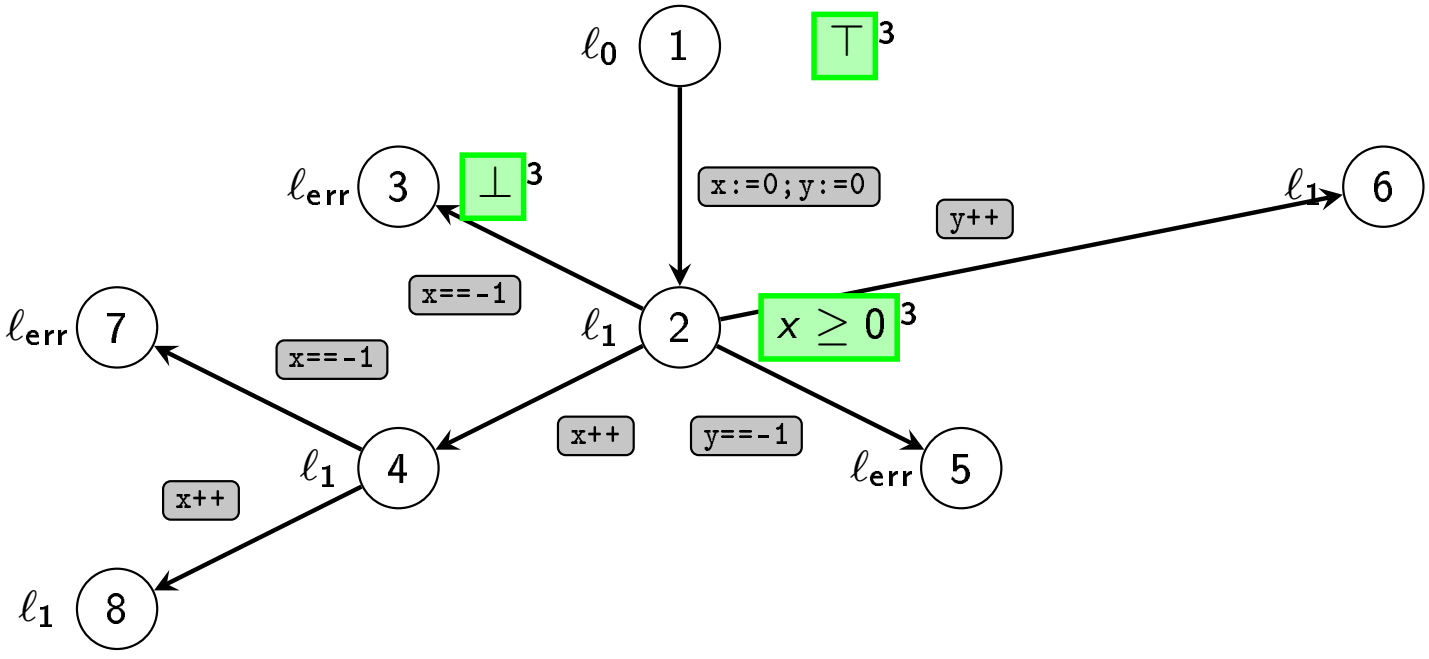
Kurzvorstellung IMPACT



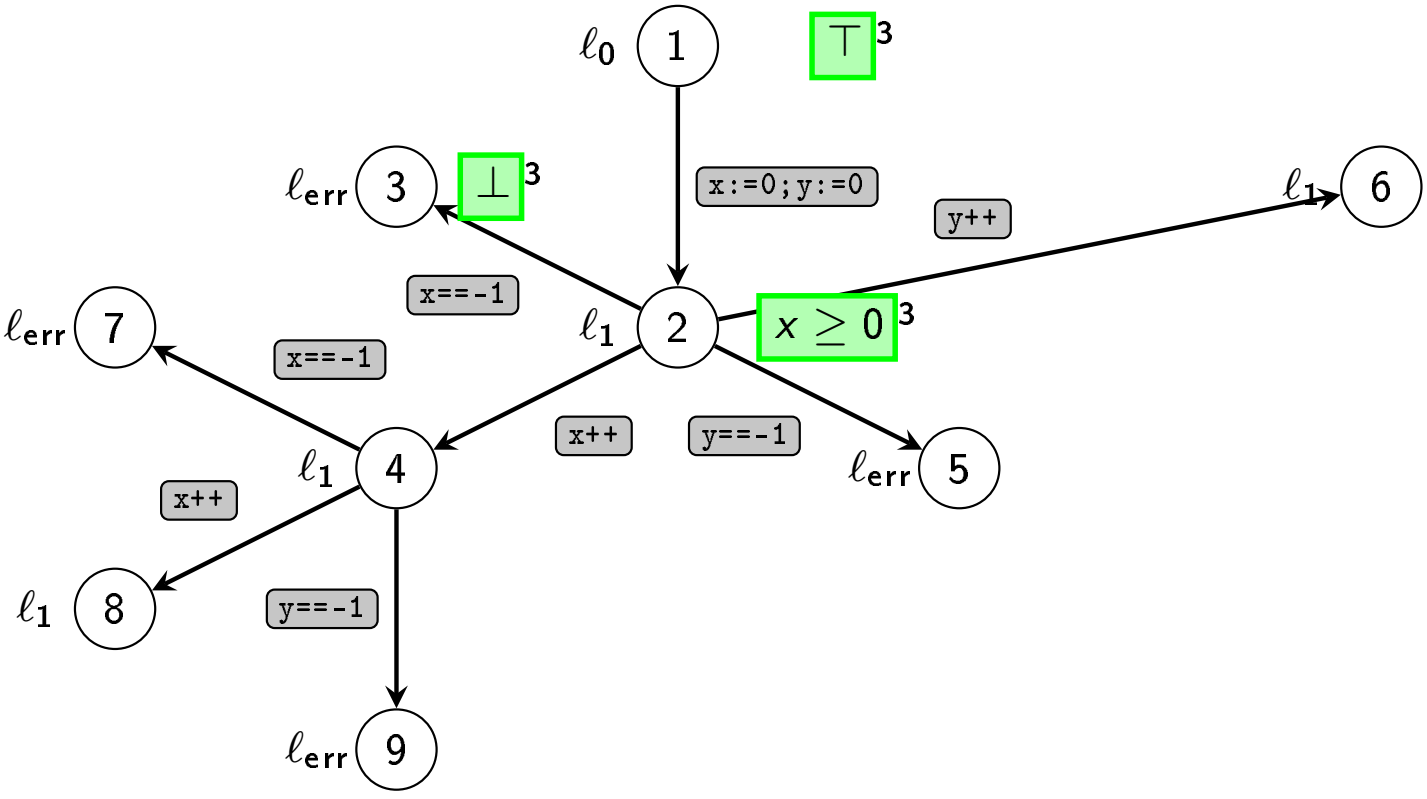
Kurzvorstellung IMPACT



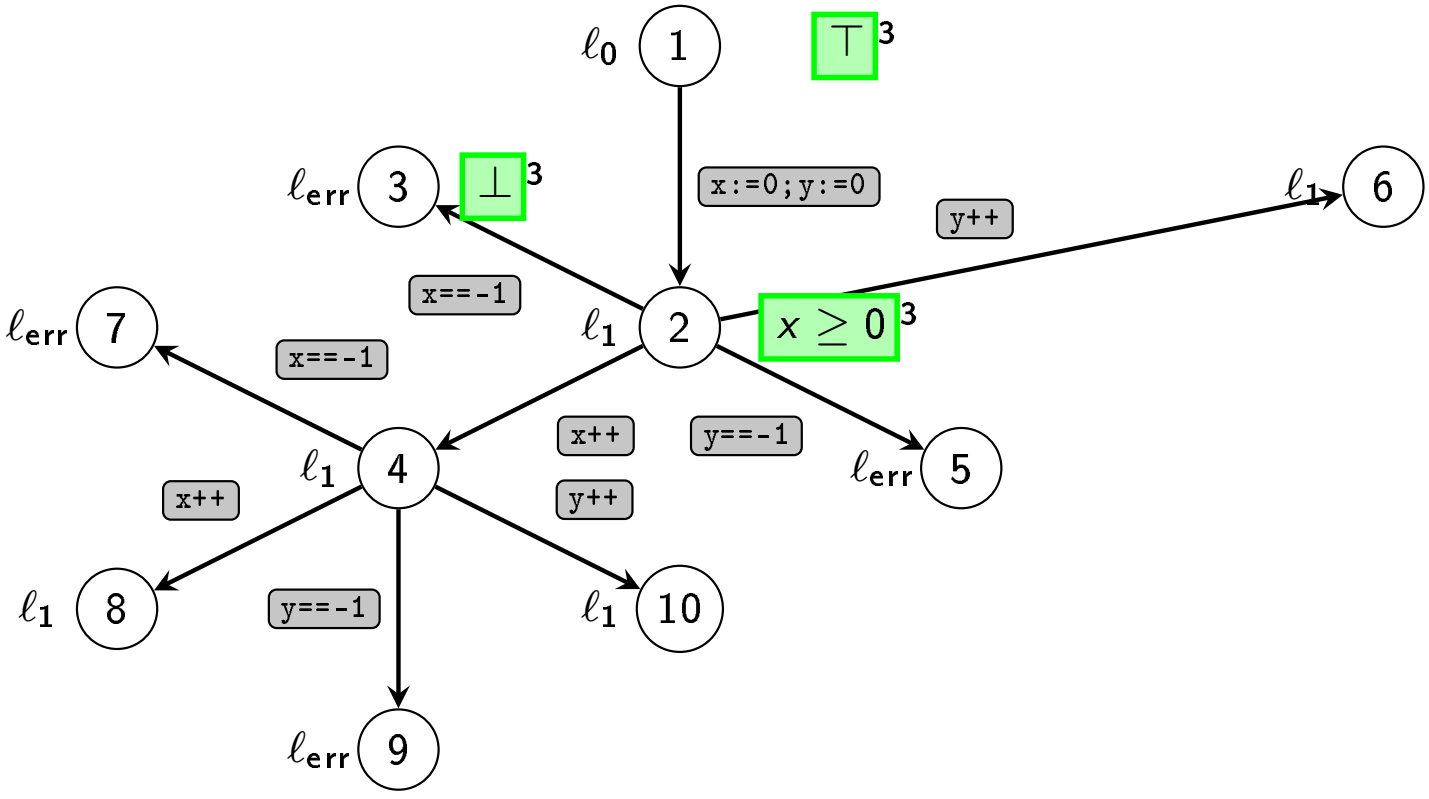
Kurzvorstellung IMPACT



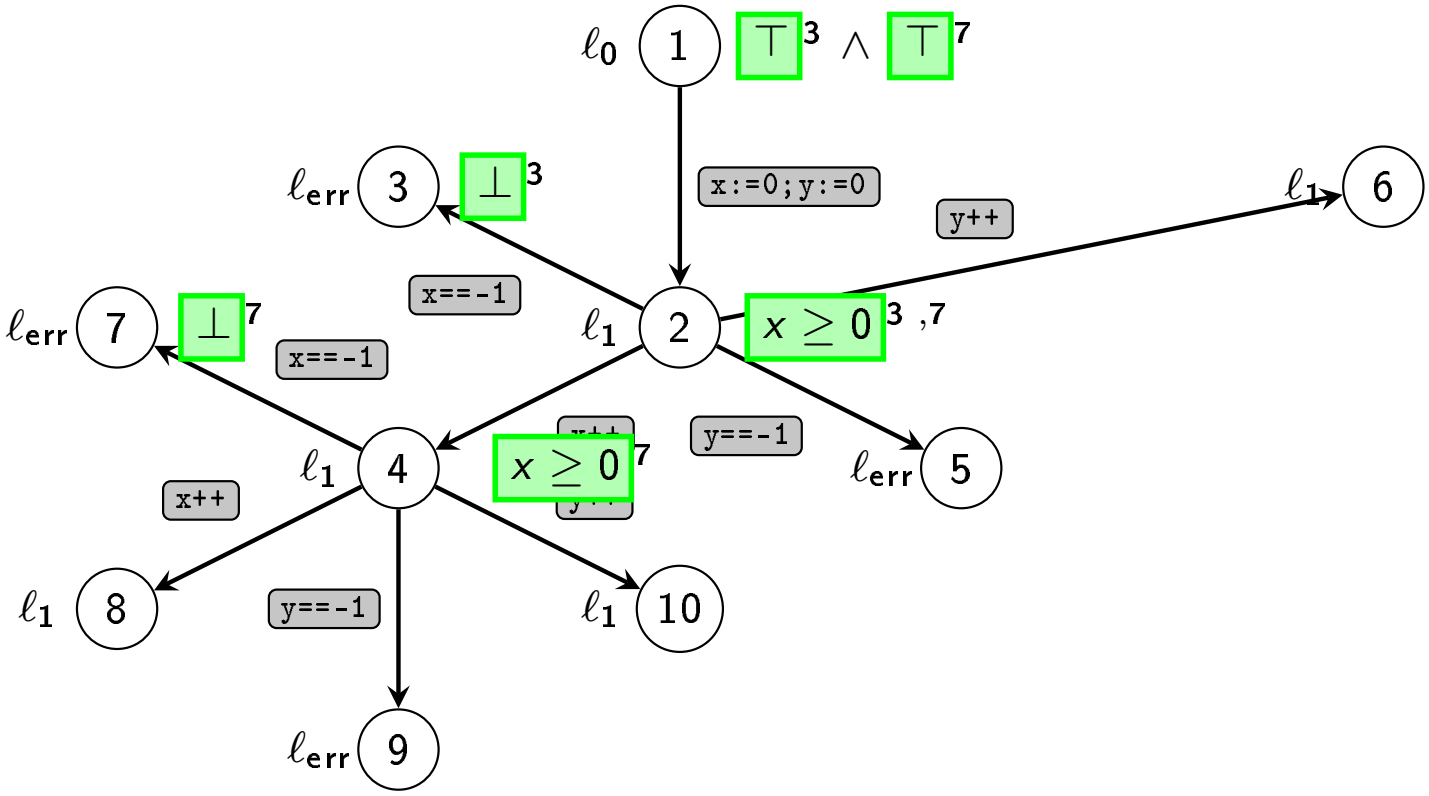
Kurzvorstellung IMPACT



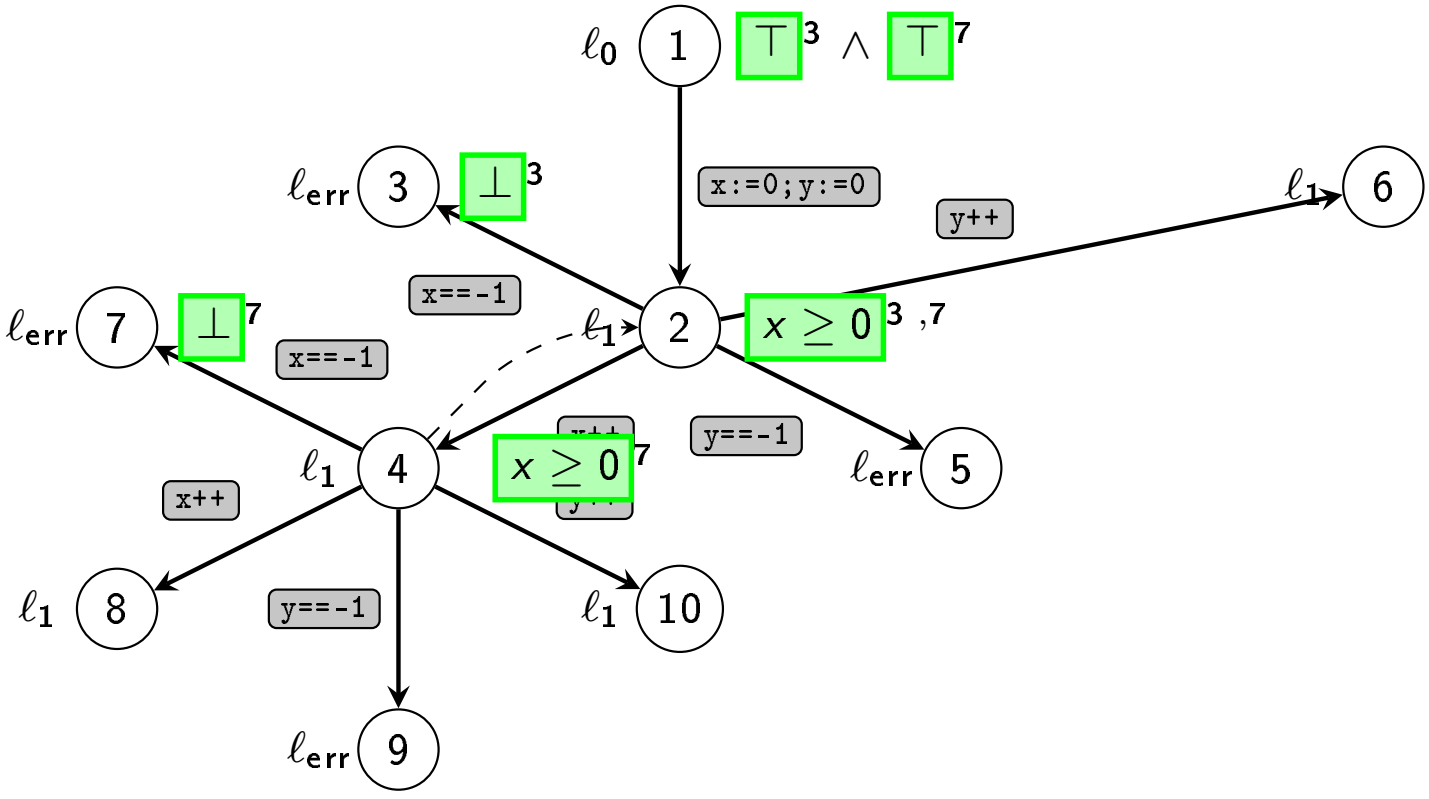
Kurzvorstellung IMPACT



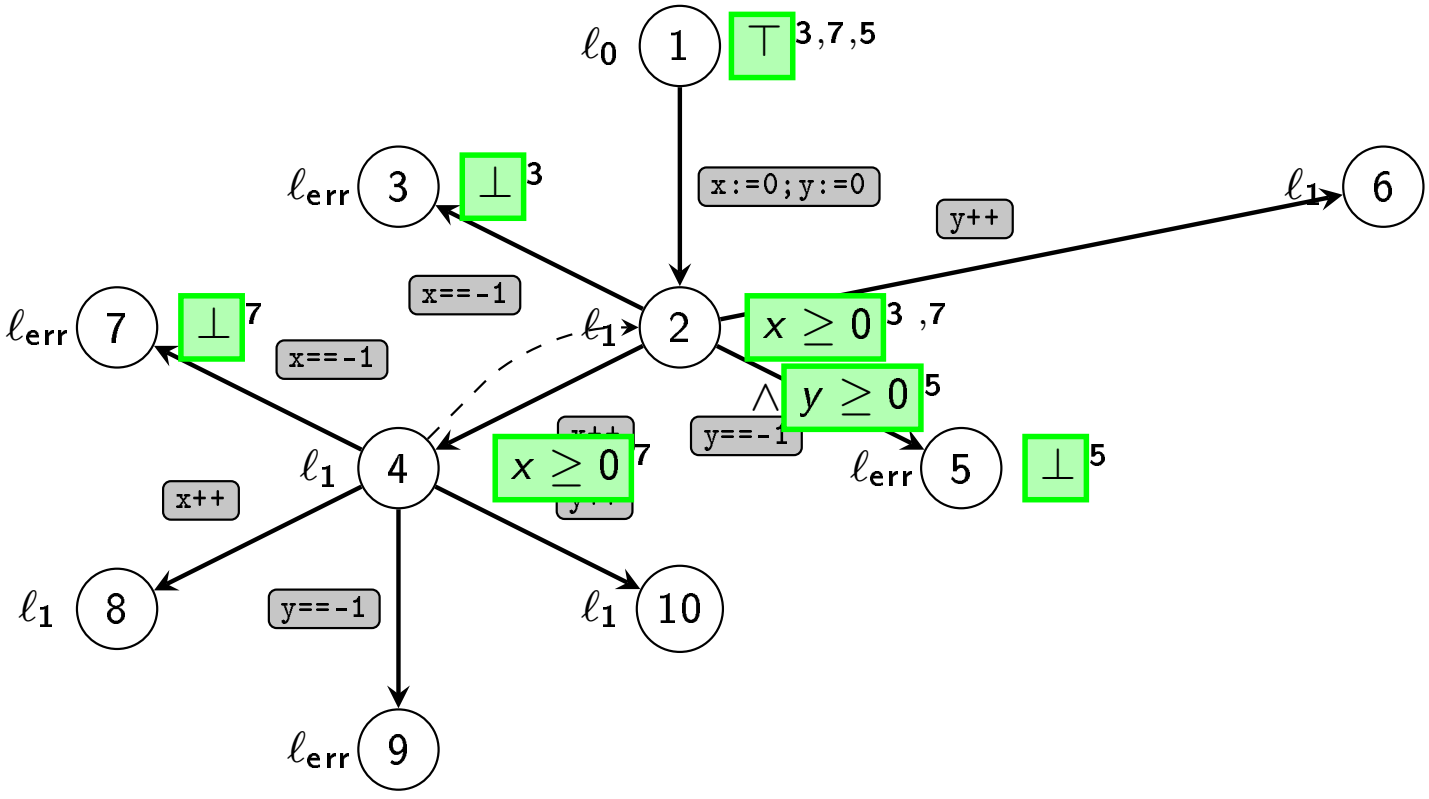
Kurzvorstellung IMPACT



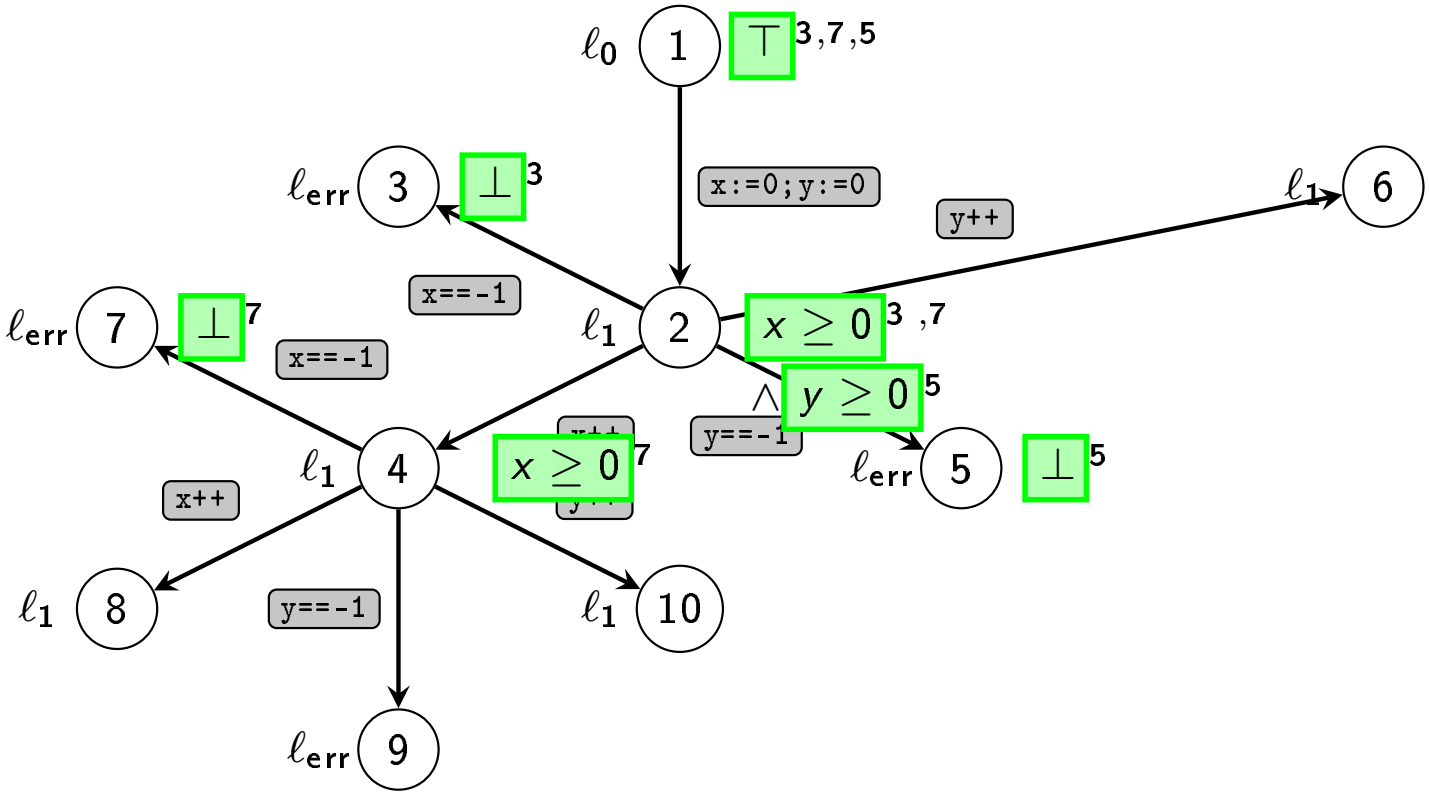
Kurzvorstellung IMPACT



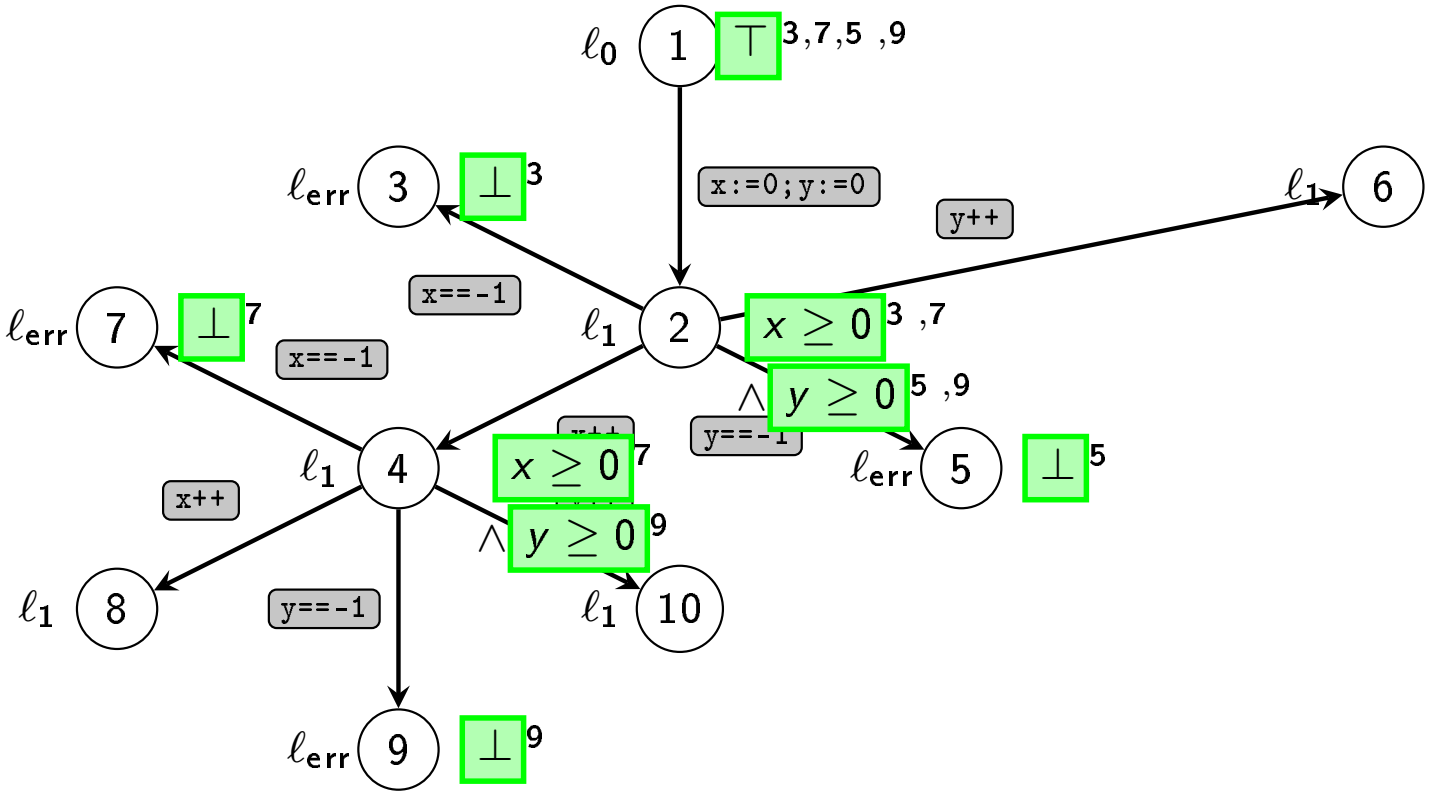
Kurzvorstellung IMPACT



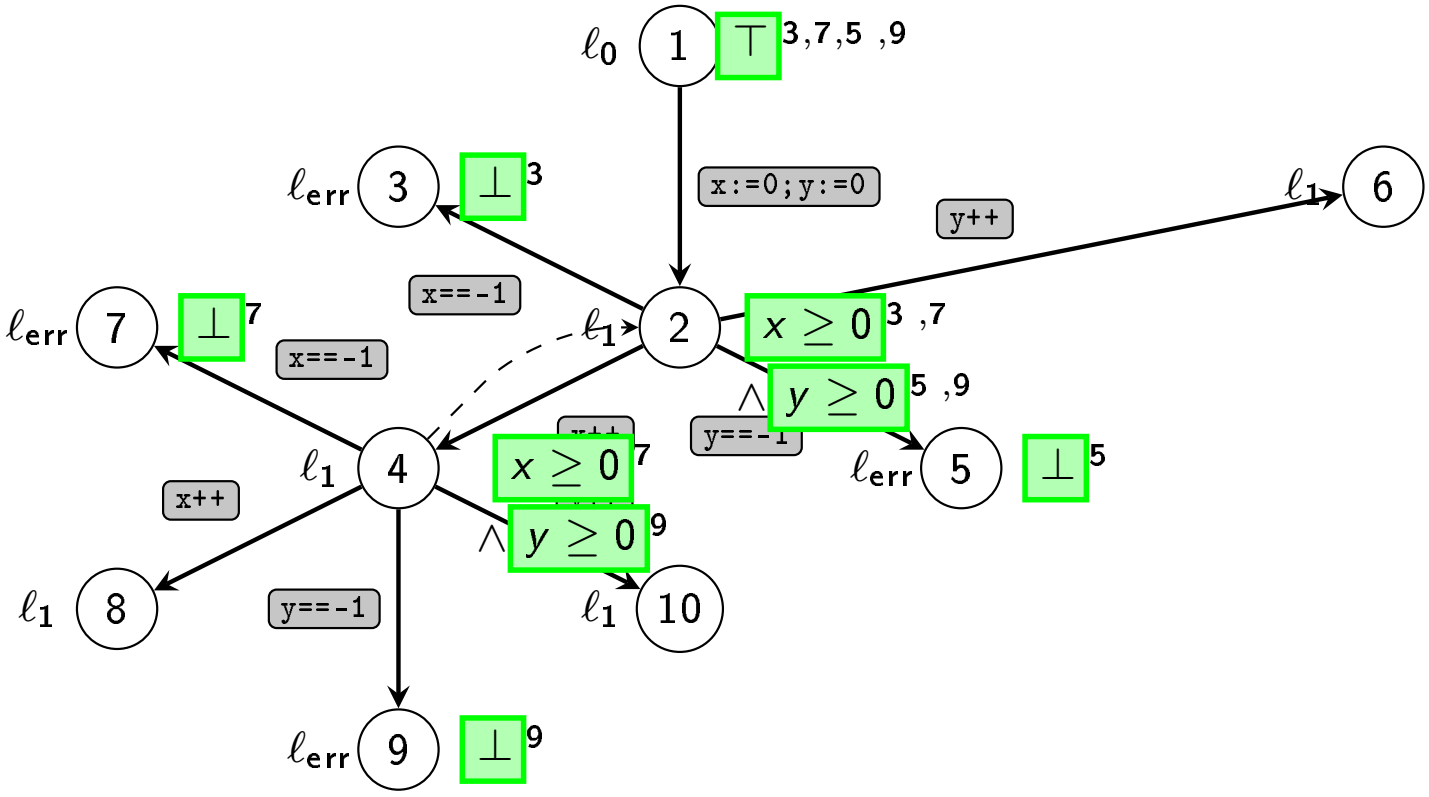
Kurzvorstellung IMPACT



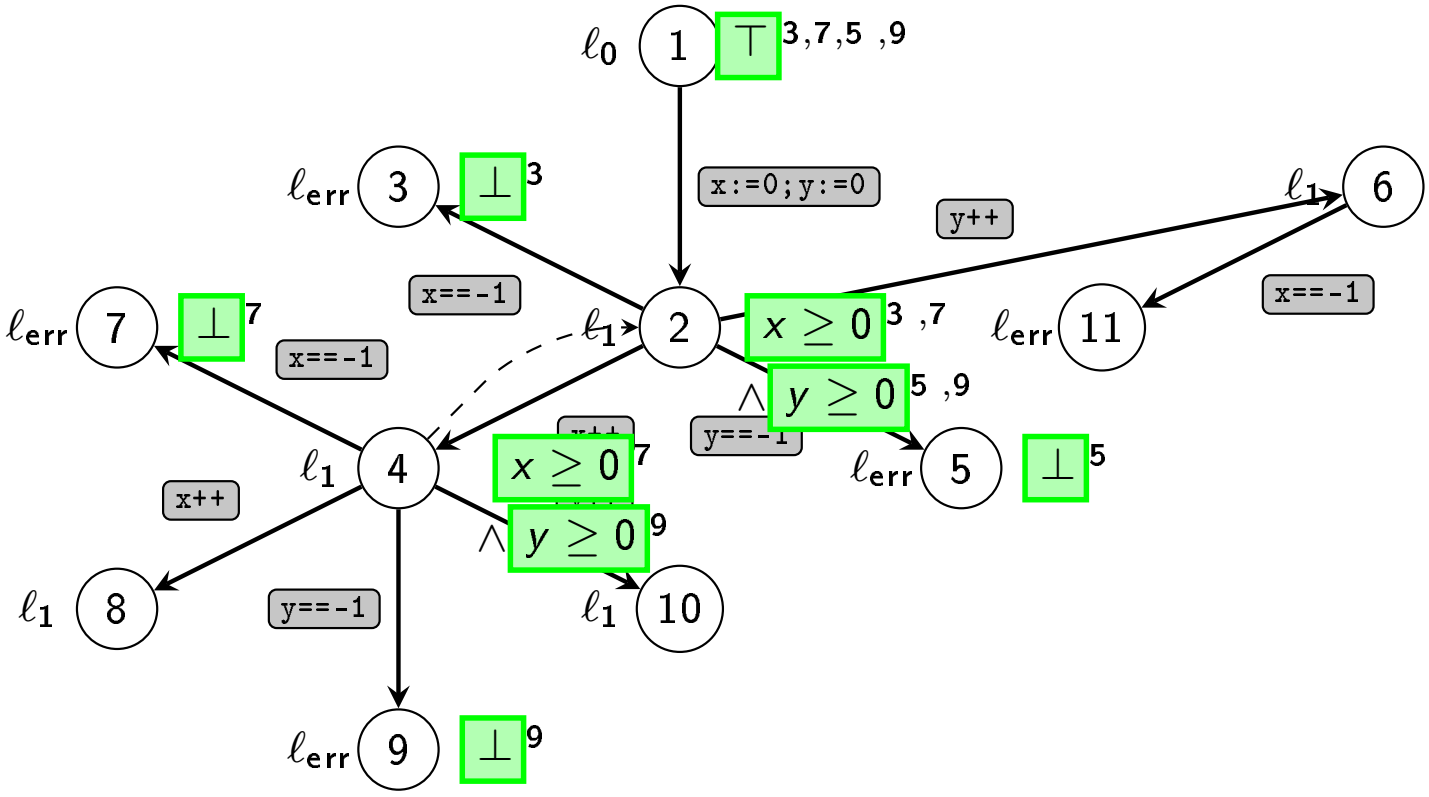
Kurzvorstellung IMPACT



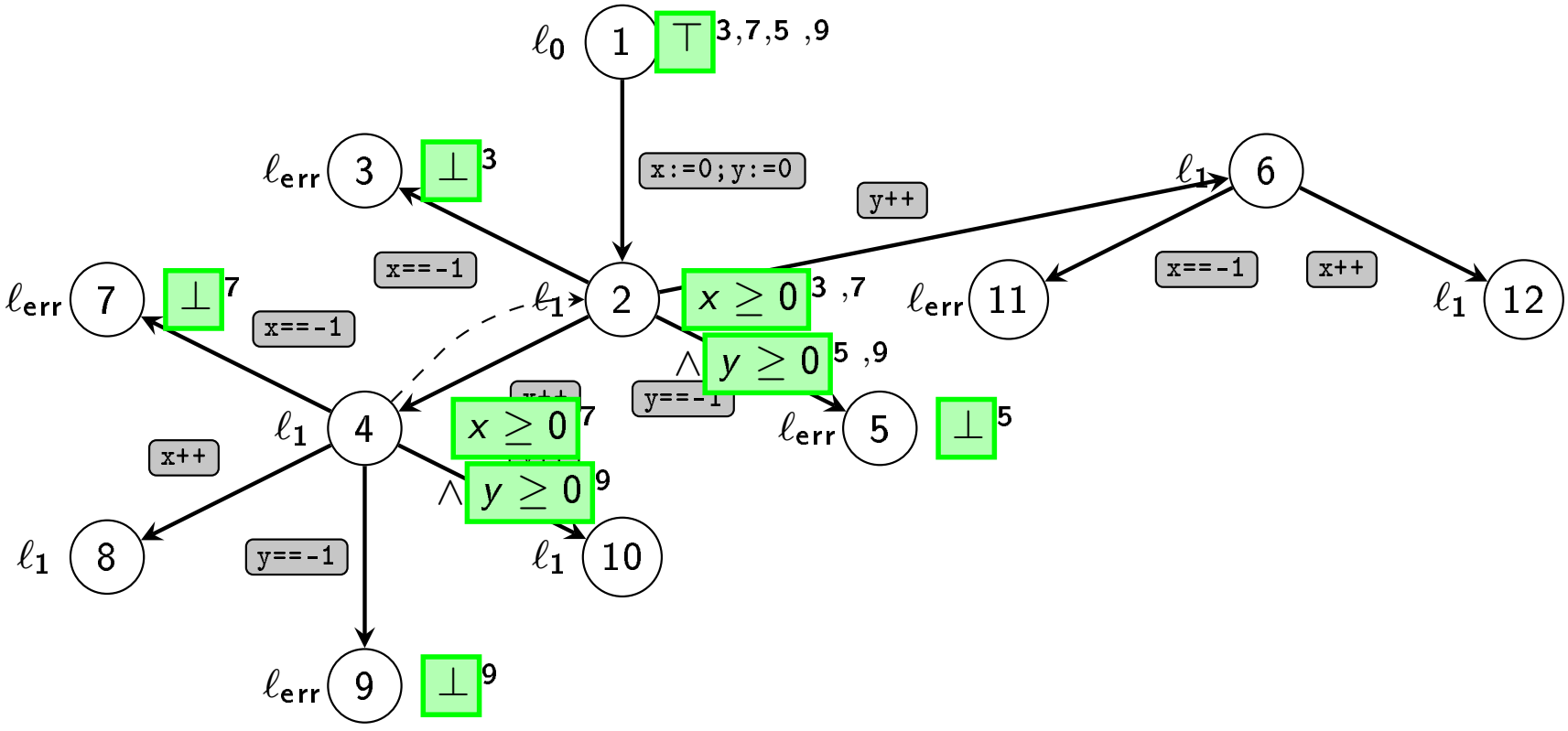
Kurzvorstellung IMPACT



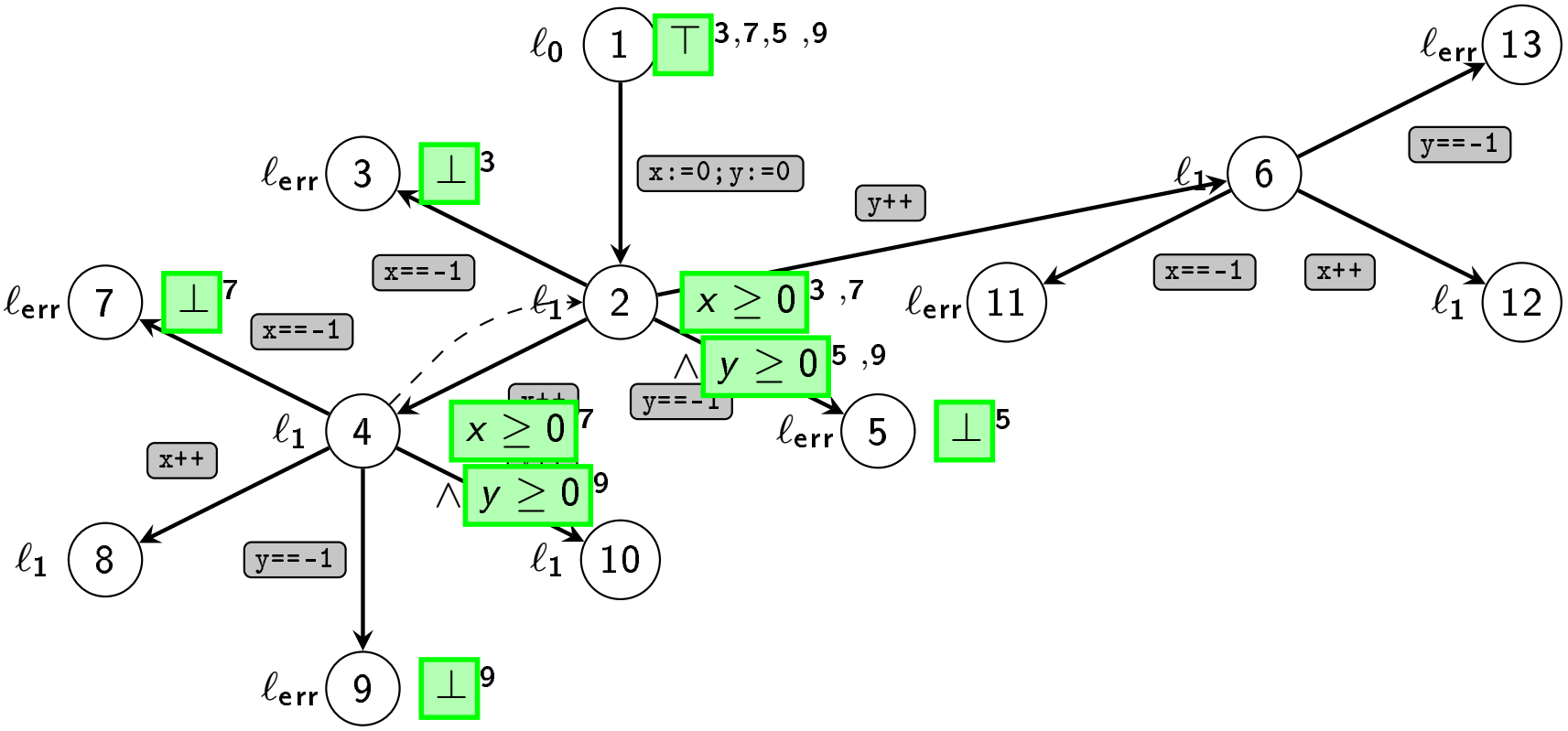
Kurzvorstellung IMPACT



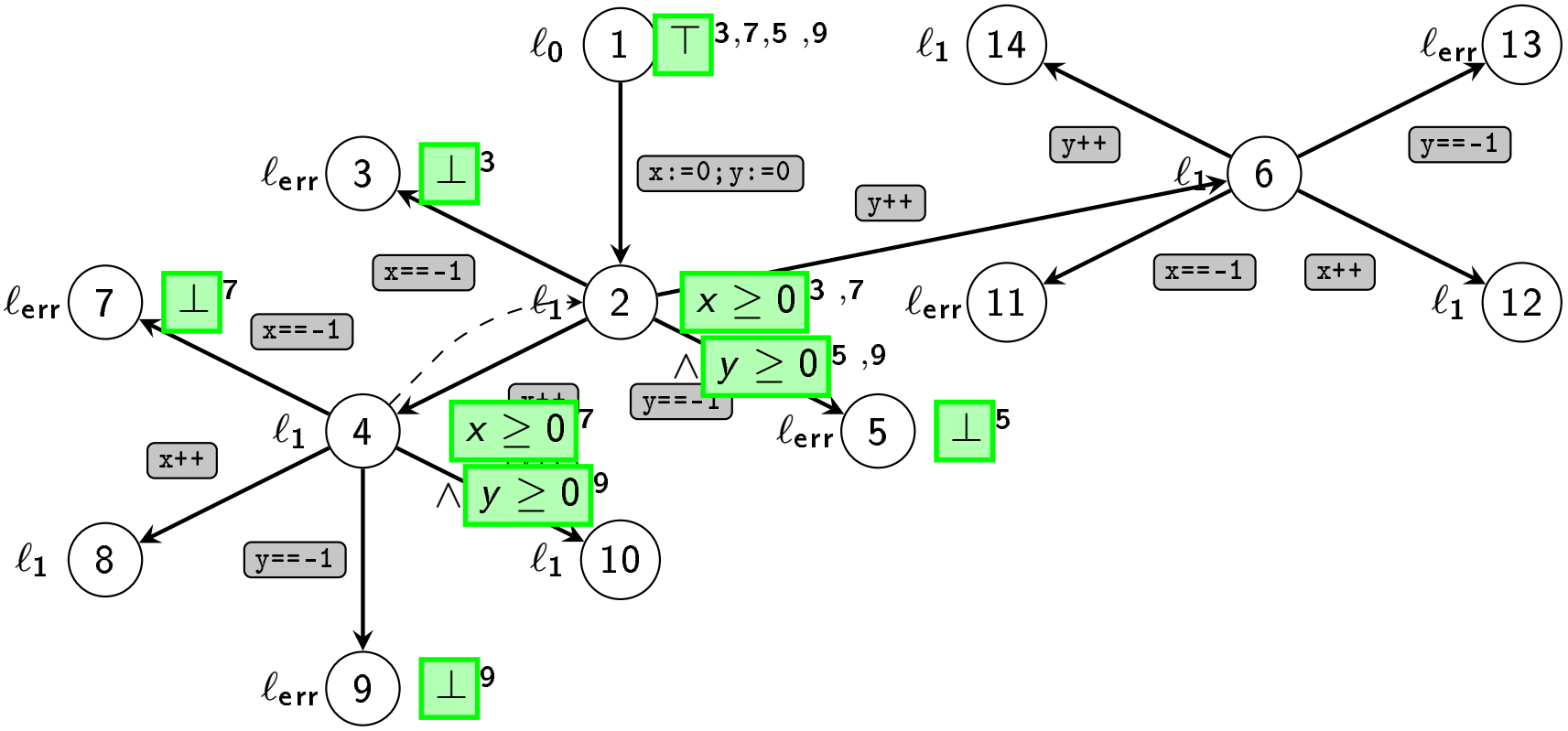
Kurzvorstellung IMPACT



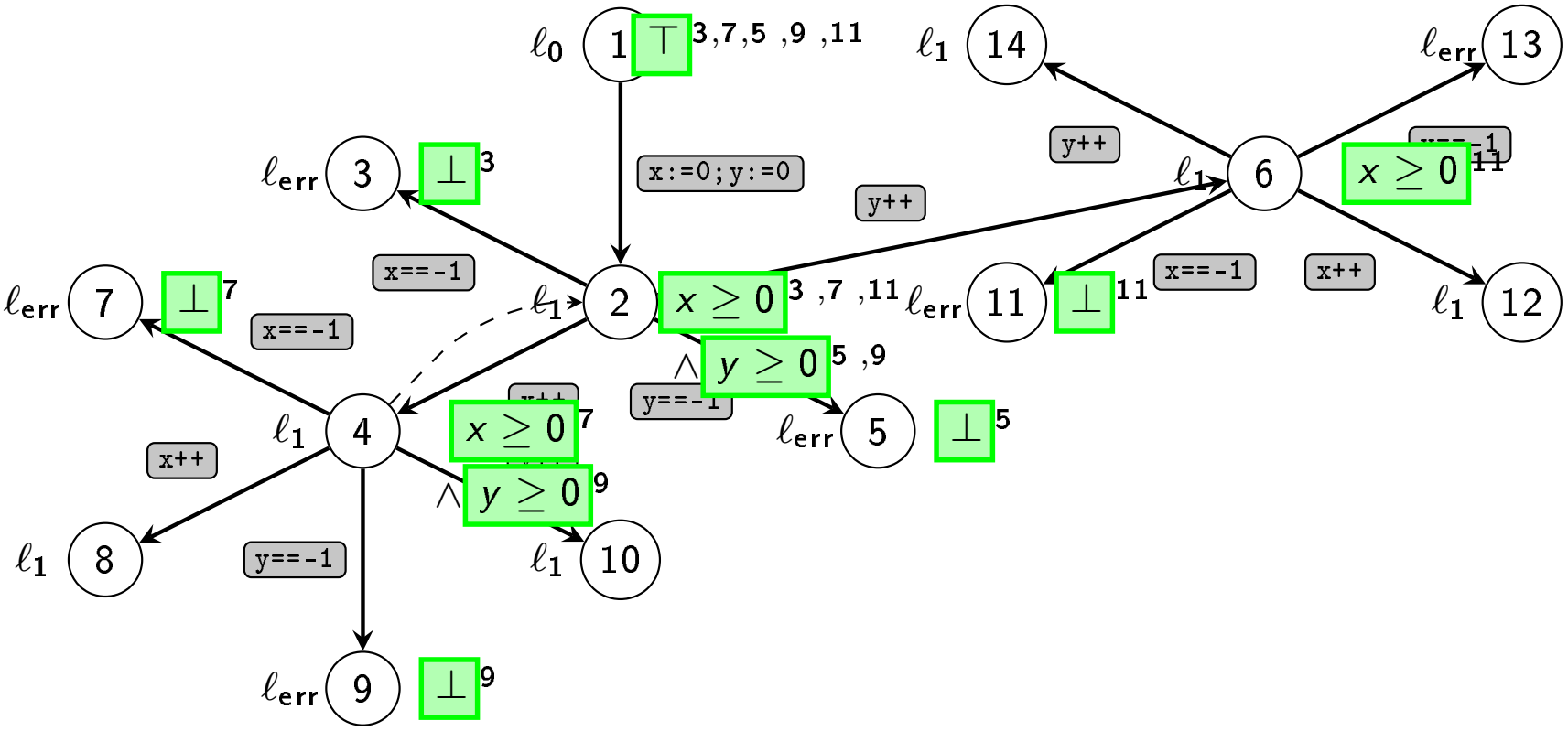
Kurzvorstellung IMPACT



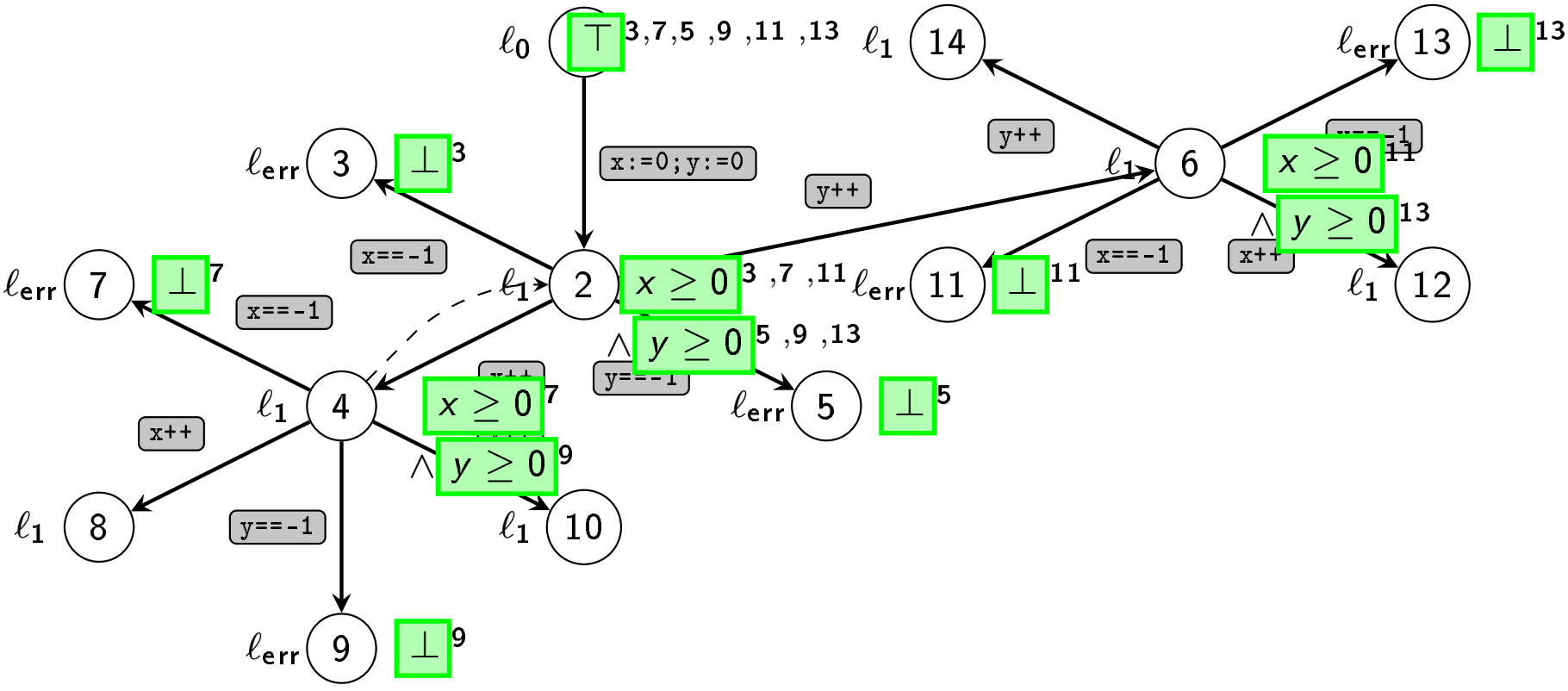
Kurzvorstellung IMPACT



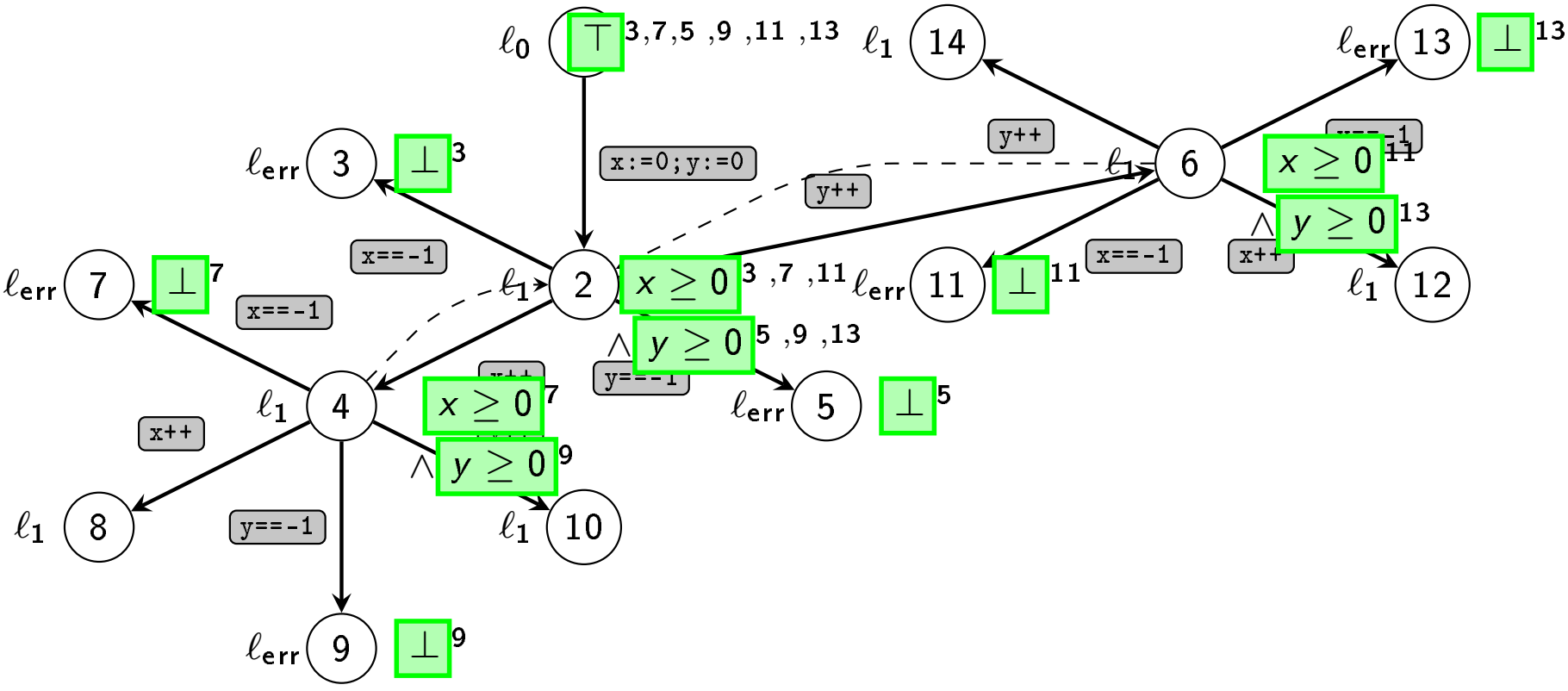
Kurzvorstellung IMPACT



Kurzvorstellung IMPACT

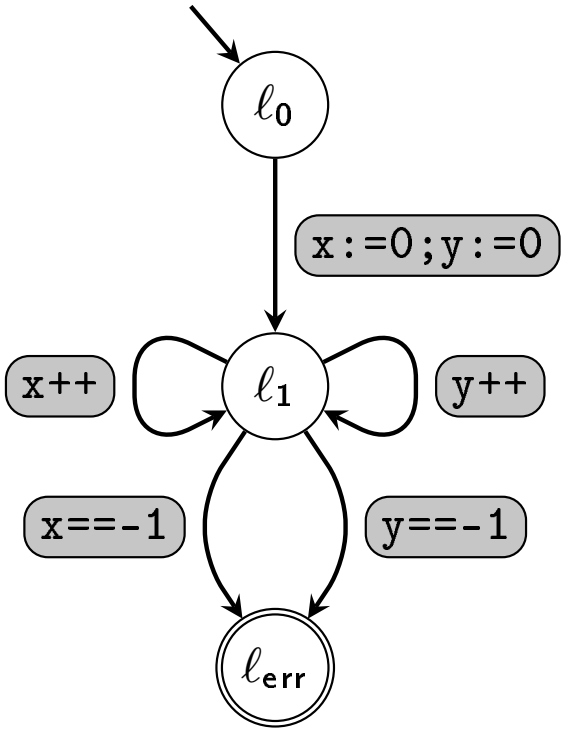


Kurzvorstellung IMPACT



Vorstellung Trace Abstraction

A_P :



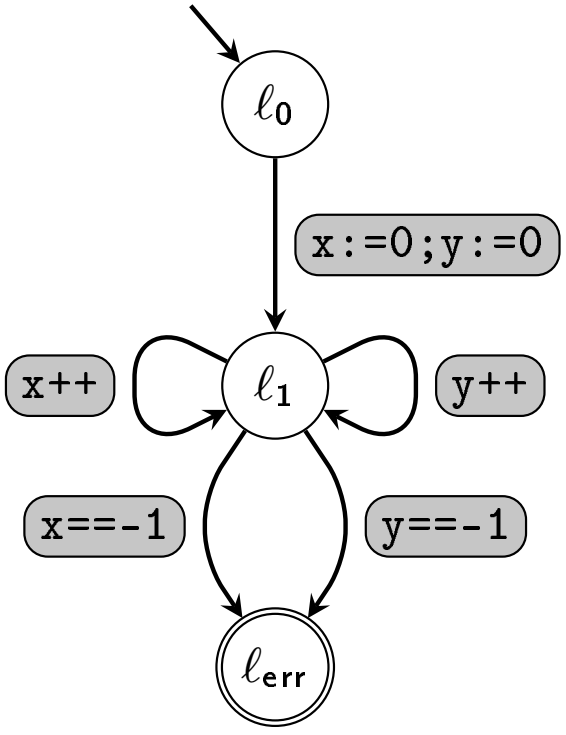
$A := A \cap \overline{A_I}$

Wiederhole bis $\mathcal{L}(A) = \emptyset$

1. Finde Error Trace π in A
2. Theorembeweiser:
   1. Falls π infeasible: liefere Interpolantensequenz
   2. Sonst: Programm unsafe
3. Konstruiere Interpolantenautomat A_I
4. $A := A \cap \overline{A_I}$

Vorstellung Trace Abstraction

A_P :



π = (x:=0;y:=0);x++;x==-1

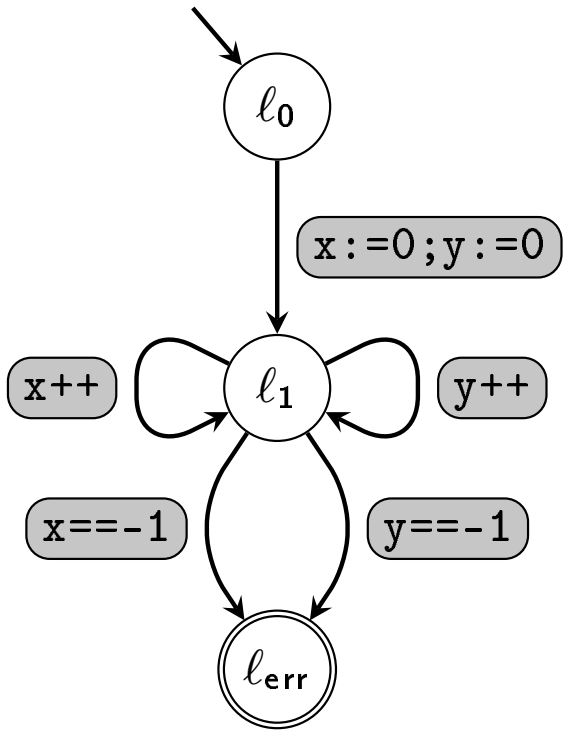
$A := A \cap \overline{A_I}$

Wiederhole bis $\mathcal{L}(A) = \emptyset$

1. **Finde Error Trace π in A**
2. Theorembeweiser:
   1. Falls π infeasible: liefere Interpolantensequenz
   2. Sonst: Programm unsafe
3. Konstruiere Interpolantenautomat A_I
4. $A := A \cap \overline{A_I}$

Vorstellung Trace Abstraction

A_P :



$\pi = (\texttt{x:=0;y:=0})\,;\texttt{x++};\texttt{x==-1}$

$I = \top, x \geq 0, x \geq 0, \bot$

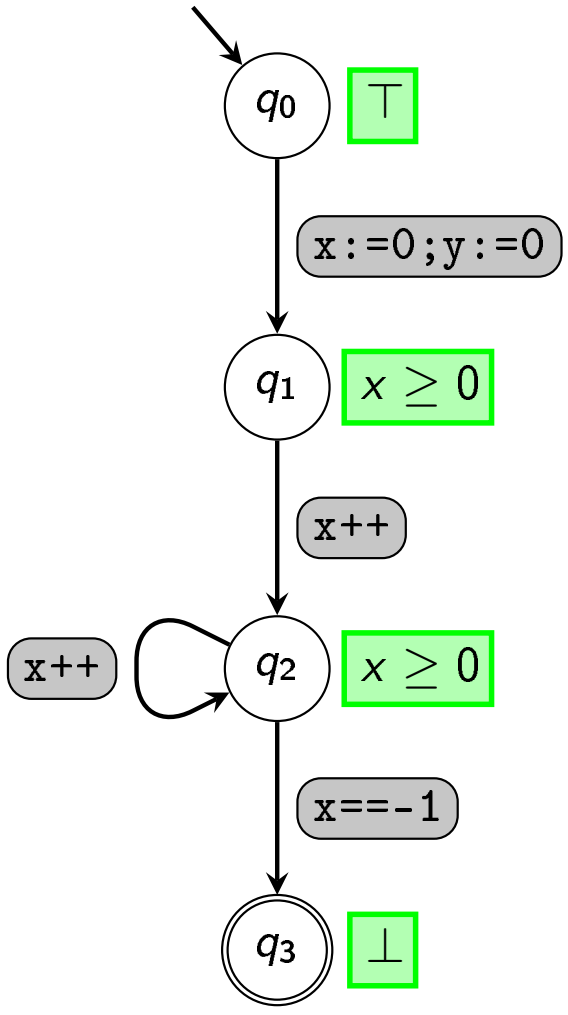
$A := A \cap \overline{A_I}$

Wiederhole bis $\mathcal{L}(A) = \emptyset$

1. Finde Error Trace π in A
2. **Theorembeweiser:**
   1. **Falls π infeasible: liefere Interpolantensequenz**
   2. Sonst: Programm unsafe
3. Konstruiere Interpolantenautomat A_I
4. $A := A \cap \overline{A_I}$

Vorstellung Trace Abstraction

A_I :



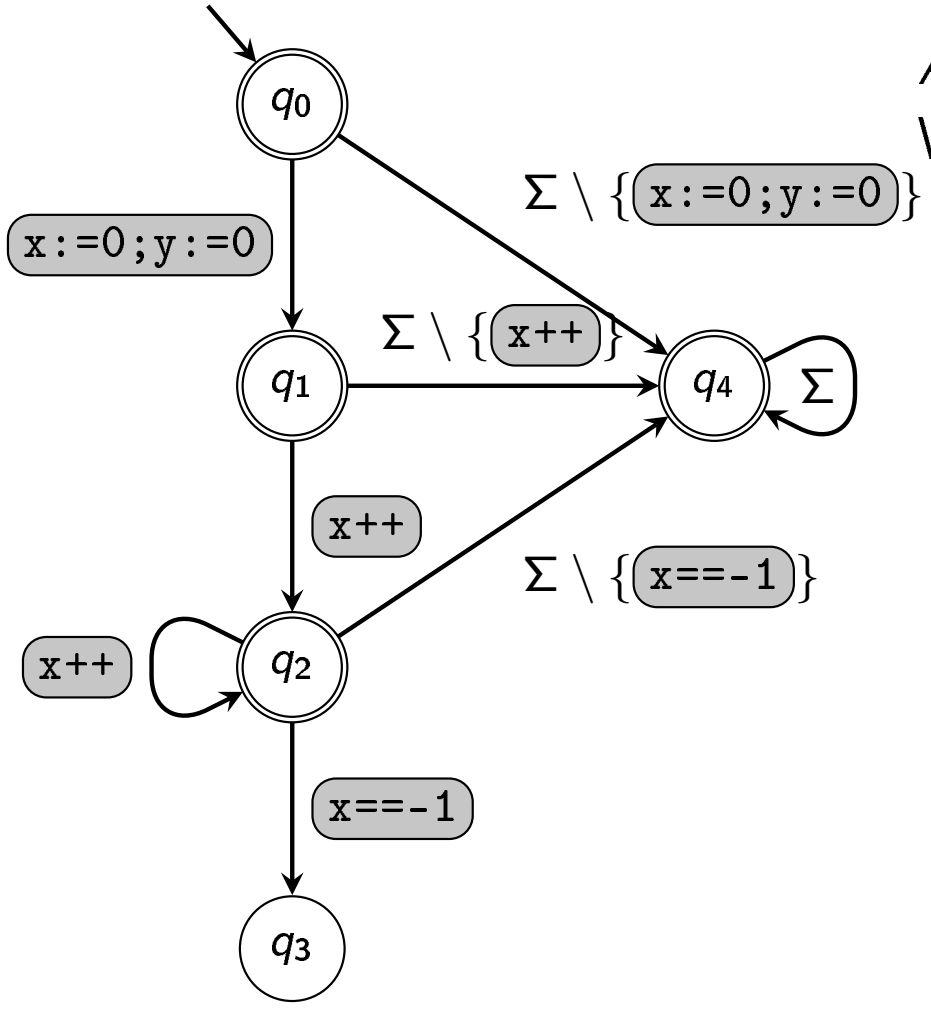
$A := A \cap \overline{A_I}$

Wiederhole bis $\mathcal{L}(A) = \emptyset$

1. Finde Error Trace π in A
2. Theorembeweiser:
   1. Falls π infeasible: liefere Interpolantensequenz
   2. Sonst: Programm unsafe
3. **Konstruiere Interpolantenautomat** A_I
4. $A := A \cap \overline{A_I}$

Vorstellung Trace Abstraction

$\overline{A_I}$:

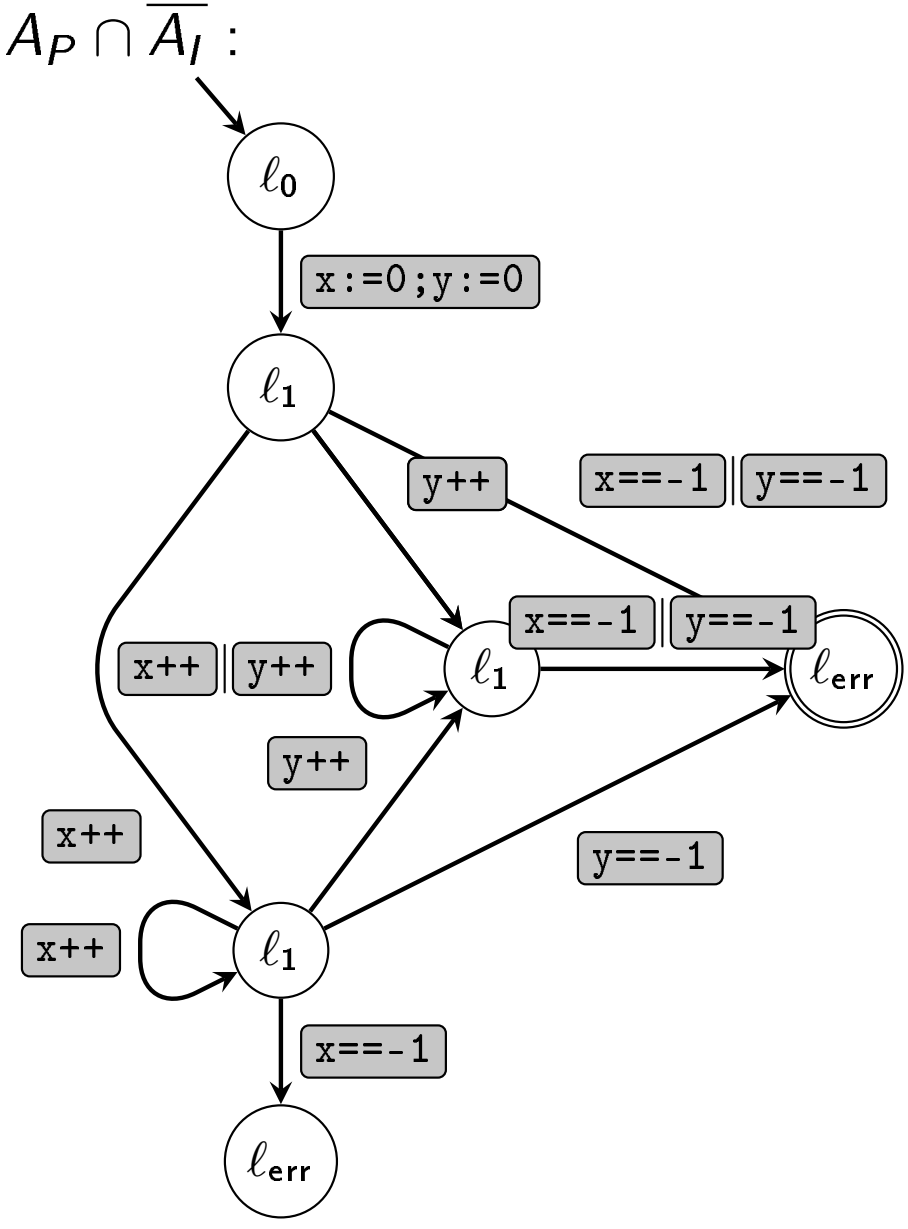


$A := A \cap \overline{A_I}$

Wiederhole bis $\mathcal{L}(A) = \emptyset$

1. Finde Error Trace π in A
2. Theorembeweiser:
   1. Falls π infeasible: liefere Interpolantensequenz
   2. Sonst: Programm unsafe
3. Konstruiere Interpolantenautomat A_I
4. $\boldsymbol{A := A \cap \overline{A_I}}$

Vorstellung Trace Abstraction



$A := A \cap \overline{A_I}$

Wiederhole bis $\mathcal{L}(A) = \emptyset$

1. Finde Error Trace π in A
2. Theorembeweiser:
   1. Falls π infeasible: liefere Interpolantensequenz
   2. Sonst: Programm unsafe
3. Konstruiere Interpolantenautomat A_I
4. $\boldsymbol{A := A \cap \overline{A_I}}$

Vorstellung Trace Abstraction

Canonical Interpolant Automaton: 14 Interpolantenautomaten nötig

1 `x:=0;y:=0` `x==-1`

2 `x:=0;y:=0` `y==-1`

3+4 `x:=0;y:=0` (`x++`)$^+$ (`x==-1` | `y==-1`)

5+6 `x:=0;y:=0` (`y++`)$^+$ (`x==-1` | `y==-1`)

7+8 `x:=0;y:=0` ((`x++`)$^+$(`y++`)$^+$)$^+$ (`x==-1` | `y==-1`)

9+10 `x:=0;y:=0` ((`y++`)$^+$(`x++`)$^+$)$^+$ (`x==-1` | `y==-1`)

11+12 `x:=0;y:=0` (((`x++`)$^+$ (`y++`)$^+$)$^+$ (`x++`)$^+$)$^+$ (`x==-1` | `y==-1`)

`x:=0;y:=0` ((`x++`)$^+$ ((`y++`)$^+$ (`x++`)$^+$)$^+$)$^+$ (`x==-1` | `y==-1`)

13+14 `x:=0;y:=0` (((`y++`)$^+$ (`x++`)$^+$)$^+$ (`y++`)$^+$)$^+$ (`x==-1` | `y==-1`)

`x:=0;y:=0` ((`y++`)$^+$ ((`x++`)$^+$ (`y++`)$^+$)$^+$)$^+$ (`x==-1` | `y==-1`)

Erste Gegenüberstellung

	IMPACT	Trace Abstraction
Datenstruktur	Unwinding	DEA/NEA
Endlichkeit	covering	backedges
Terminierung	alle knoten gecovert oder mit $\perp$ markiert	$\mathcal{L}(A) = \emptyset$

Überblick

1. Gemeinsame Darstellungen: IMPACT mit Automaten, Trace Abstraction mit Unwindings
2. Vergleiche von Variationen von IMPACT mit Trace Abstraction

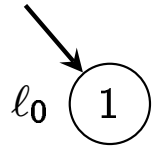
Unwindings als Automaten

Für jeden IMPACT-Iterationsschritt: Gib die aktuelle Unterapproximation der Infeasible Traces an

- Fasse Unwindings als endliche Automaten auf
- Error locations, die mit $\perp$ markiert sind werden als akzeptierend markiert
- Covering edges werden zu ϵ-Transitionen

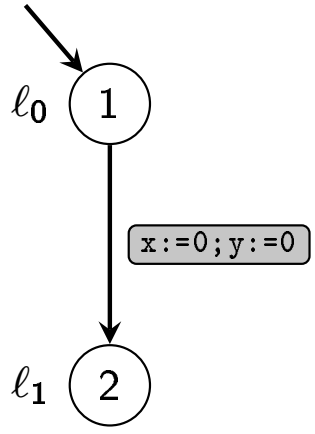
Die jeweils akzeptierte Sprache ist die aktuelle Unterapproximation.

Unwindings als Automaten



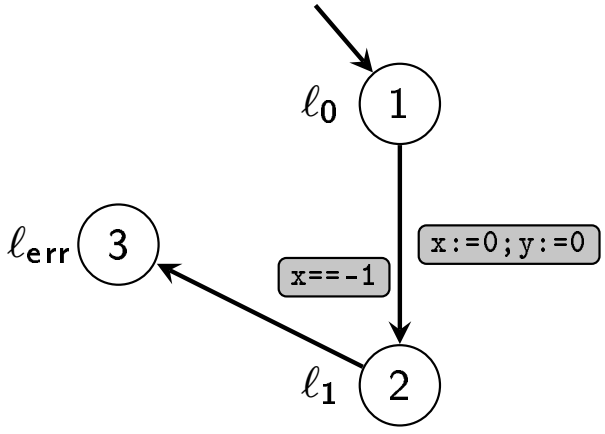
$$\mathcal{L} =$$

Unwindings als Automaten



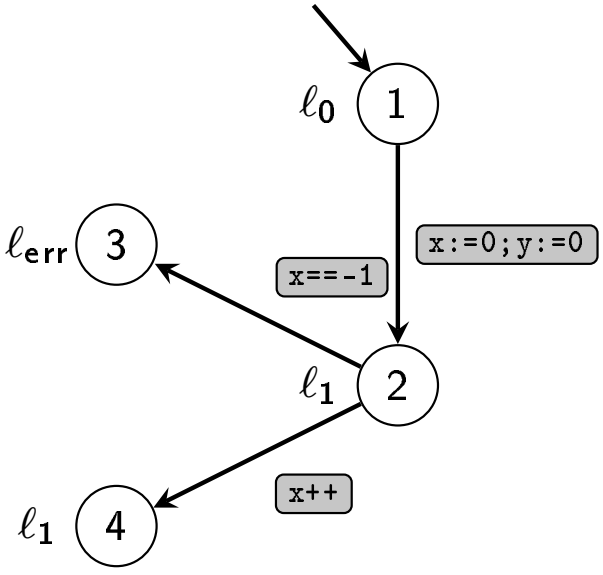
$$\mathcal{L} =$$

Unwindings als Automaten



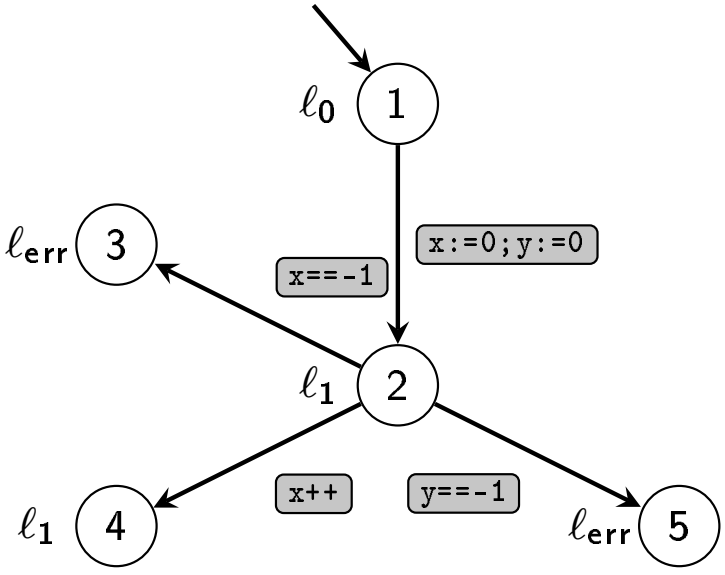
$$\mathcal{L} =$$

Unwindings als Automaten



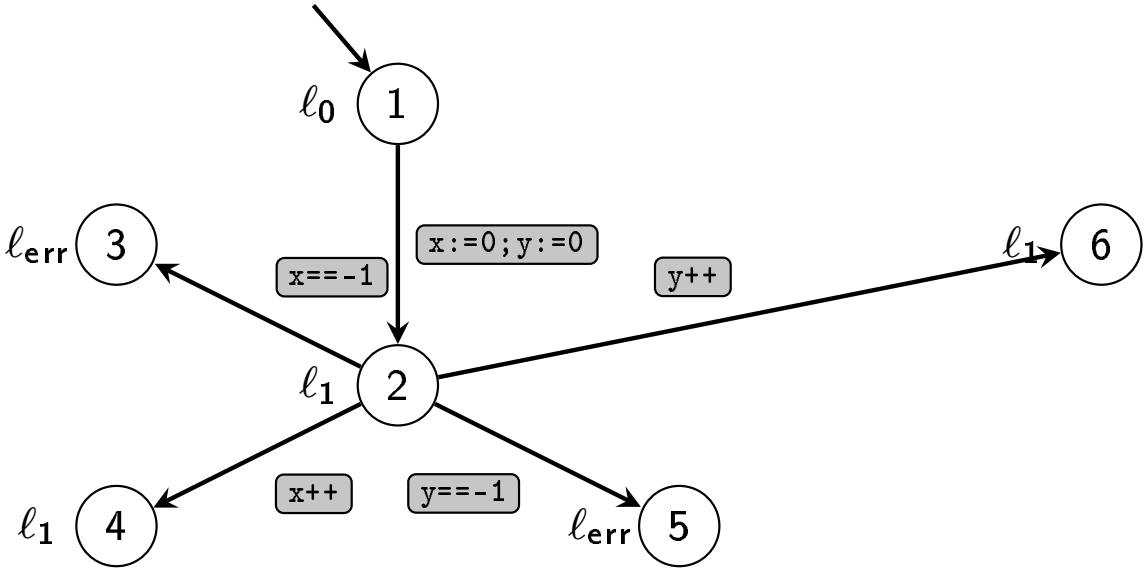
$$\mathcal{L} =$$

Unwindings als Automaten



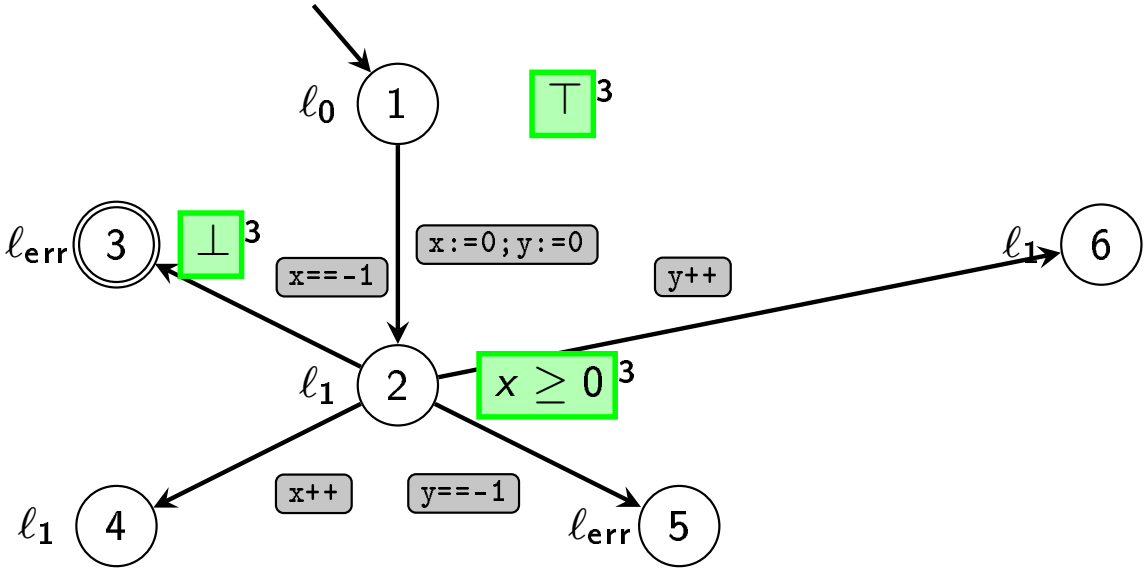
$\mathcal{L} =$

Unwindings als Automaten



$\mathcal{L} =$

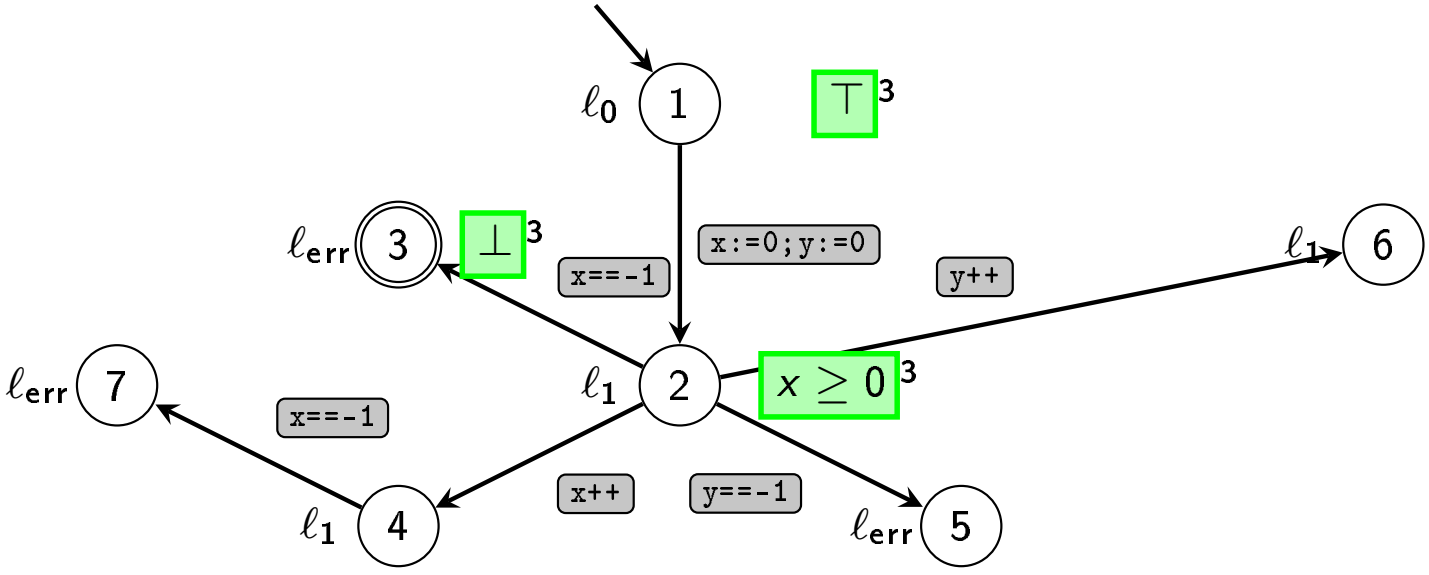
Unwindings als Automaten



$\mathcal{L} =$

3: `x:=0;y:=0` `x==-1`

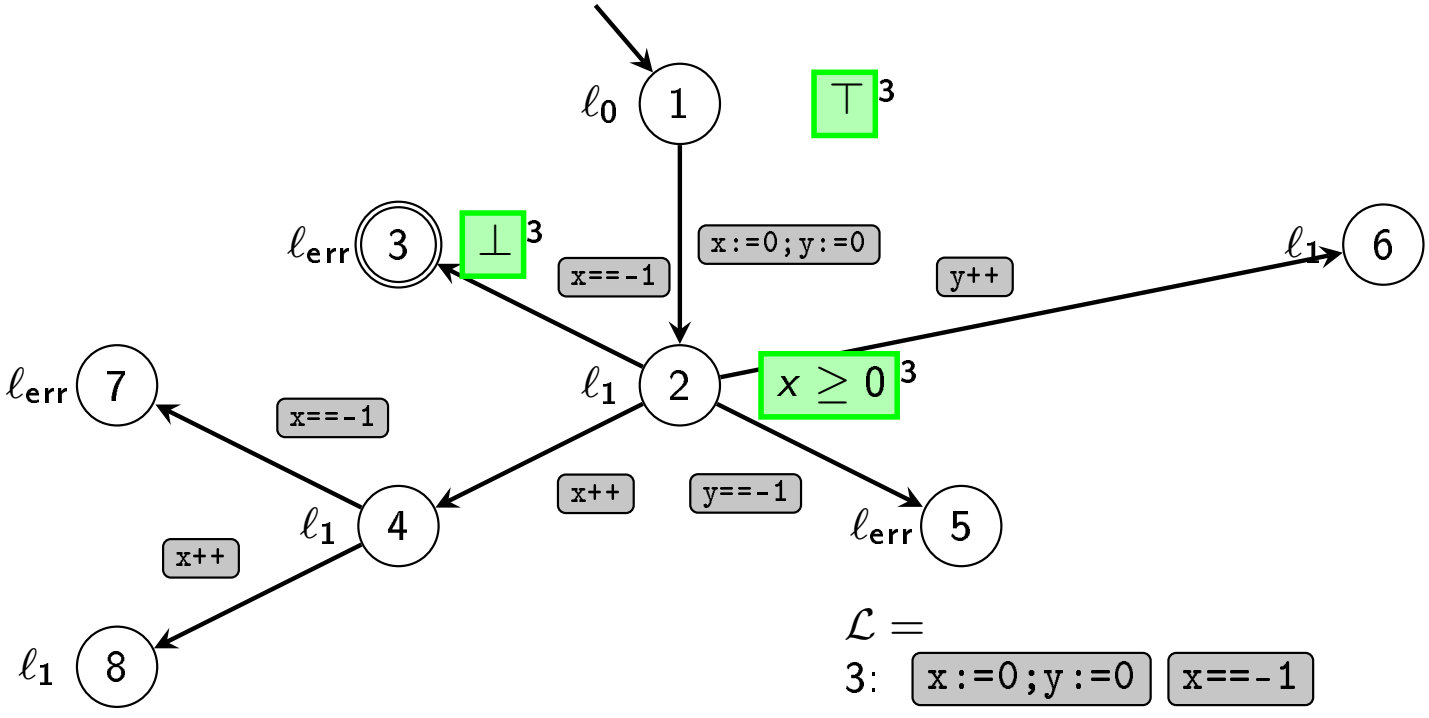
Unwindings als Automaten



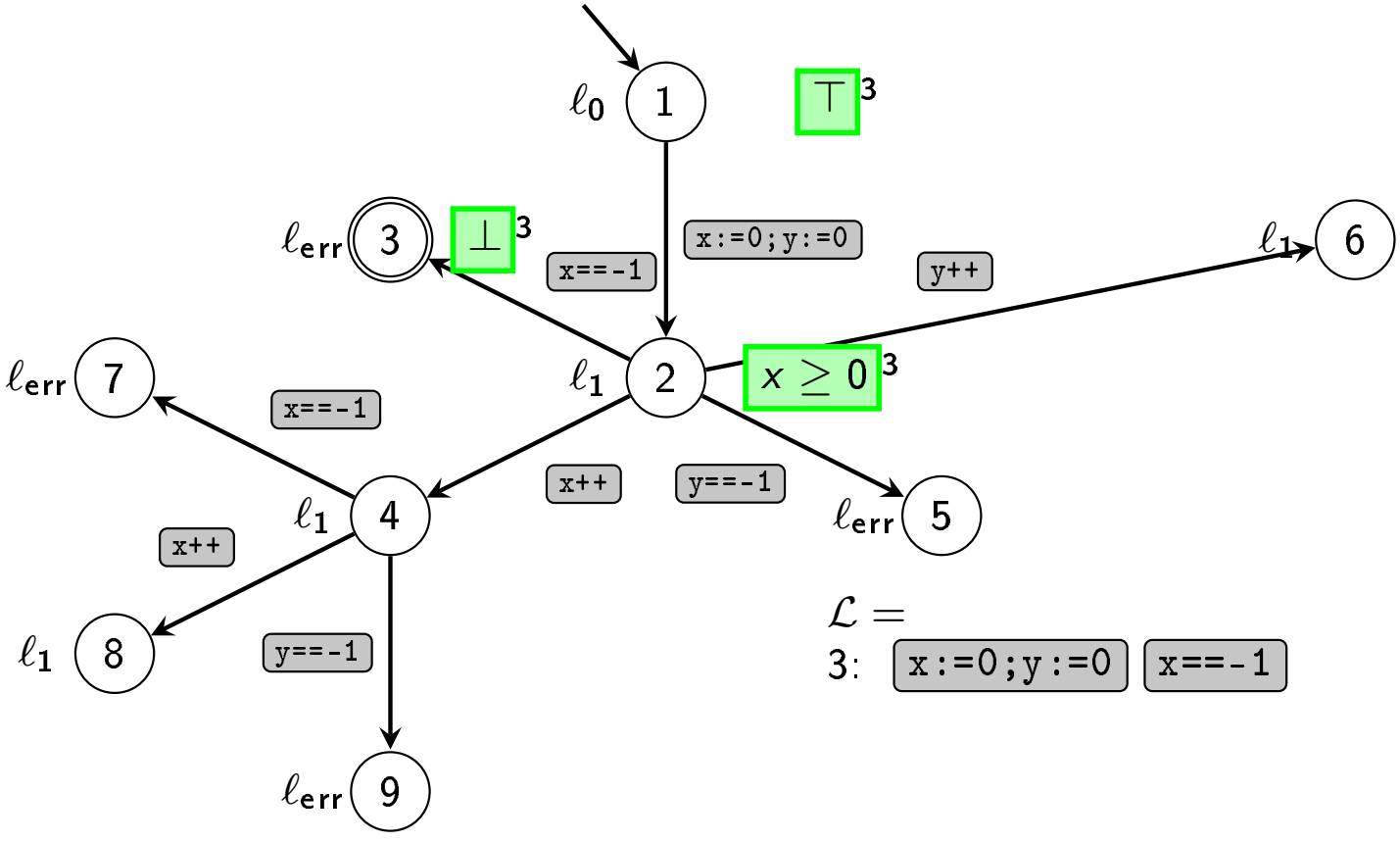
$\mathcal{L} =$

3: x:=0;y:=0 x==-1

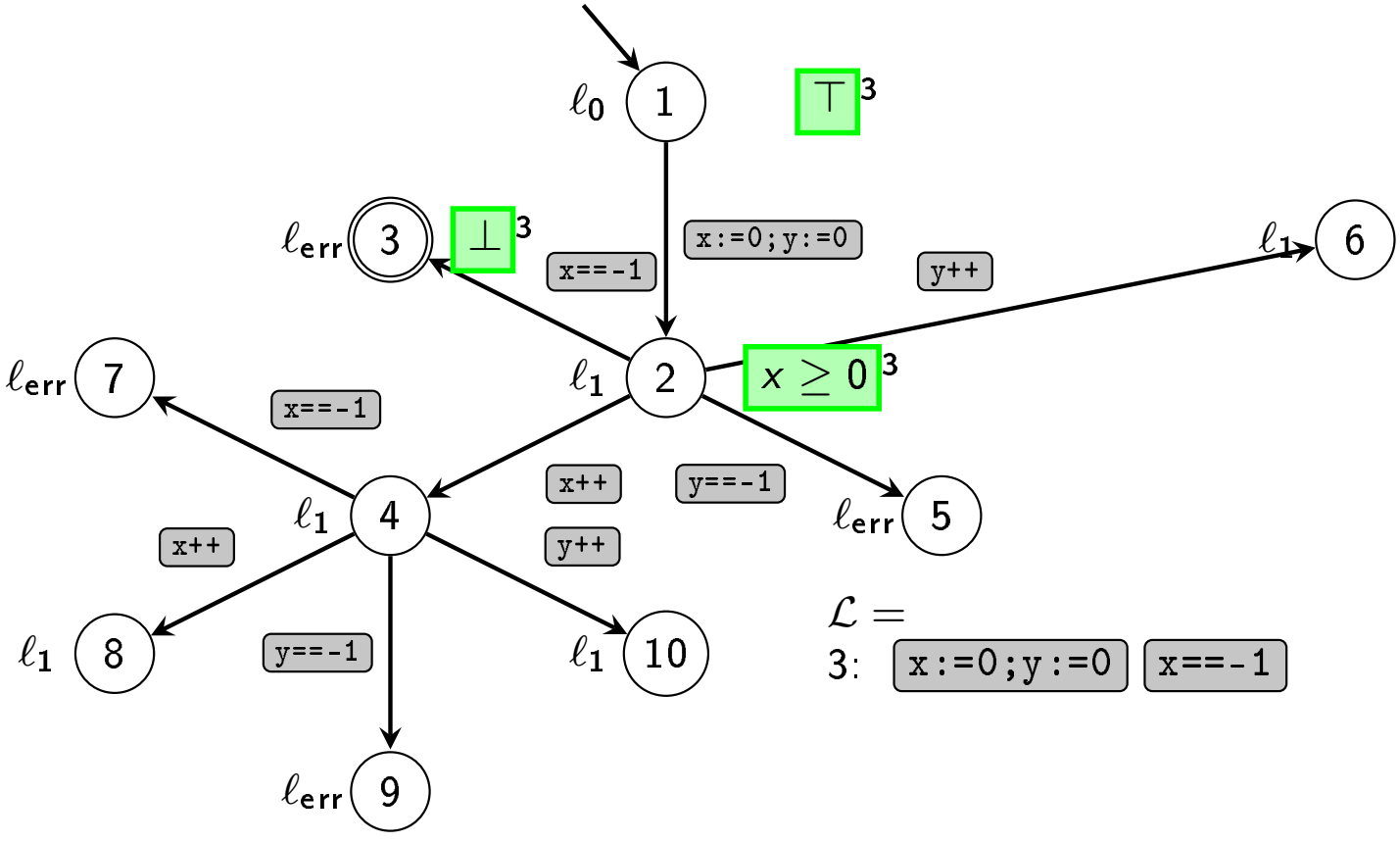
Unwindings als Automaten



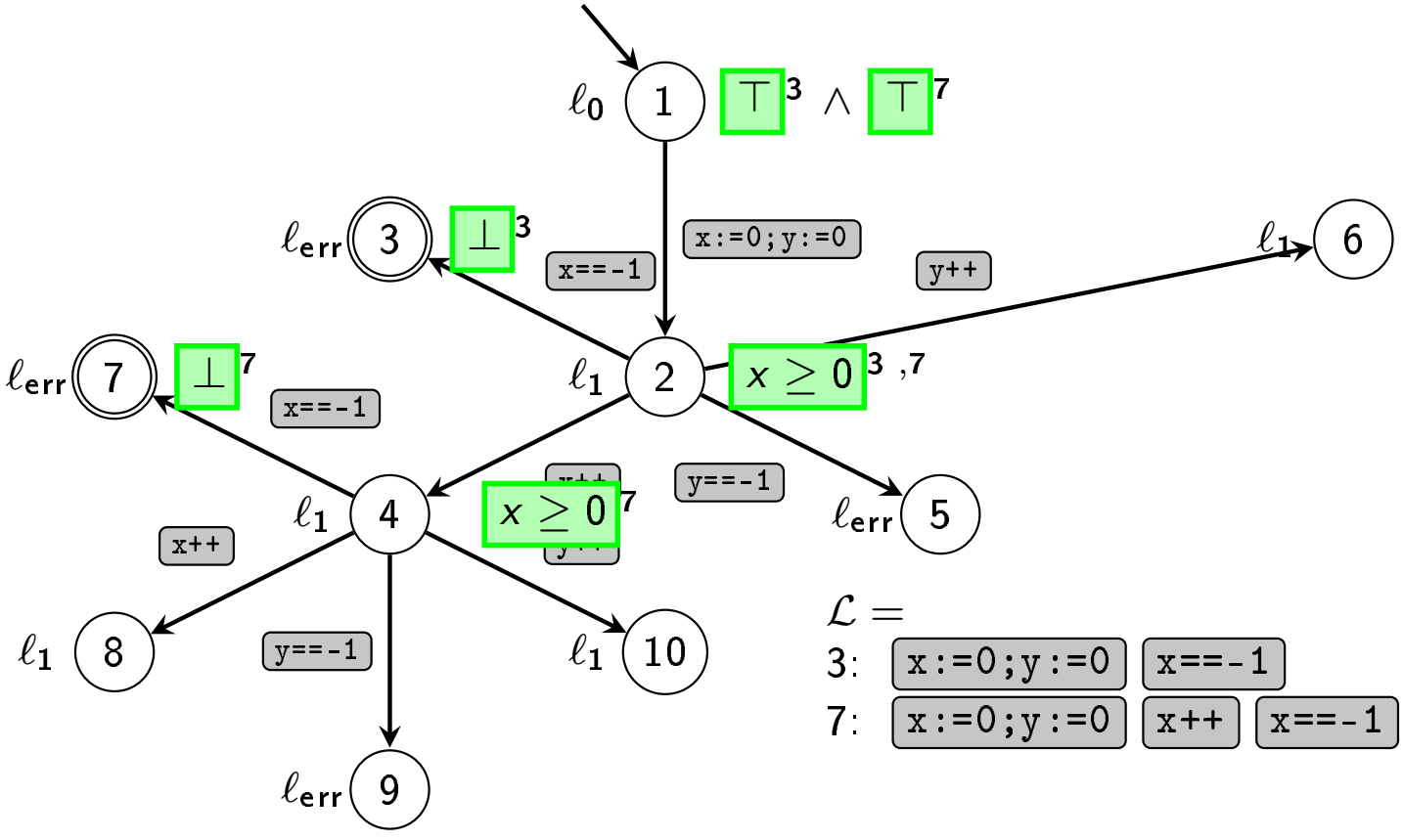
Unwindings als Automaten



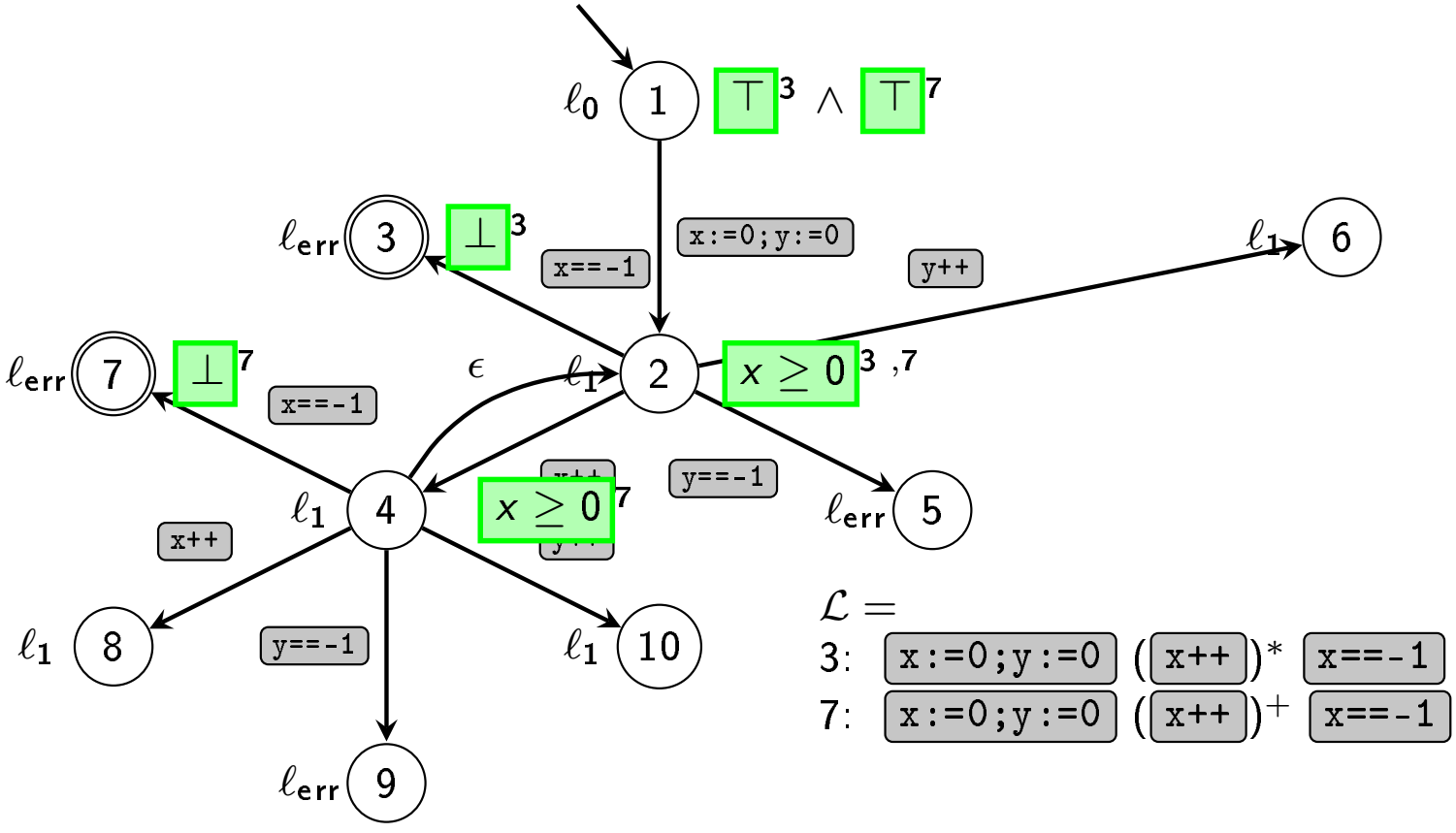
Unwindings als Automaten



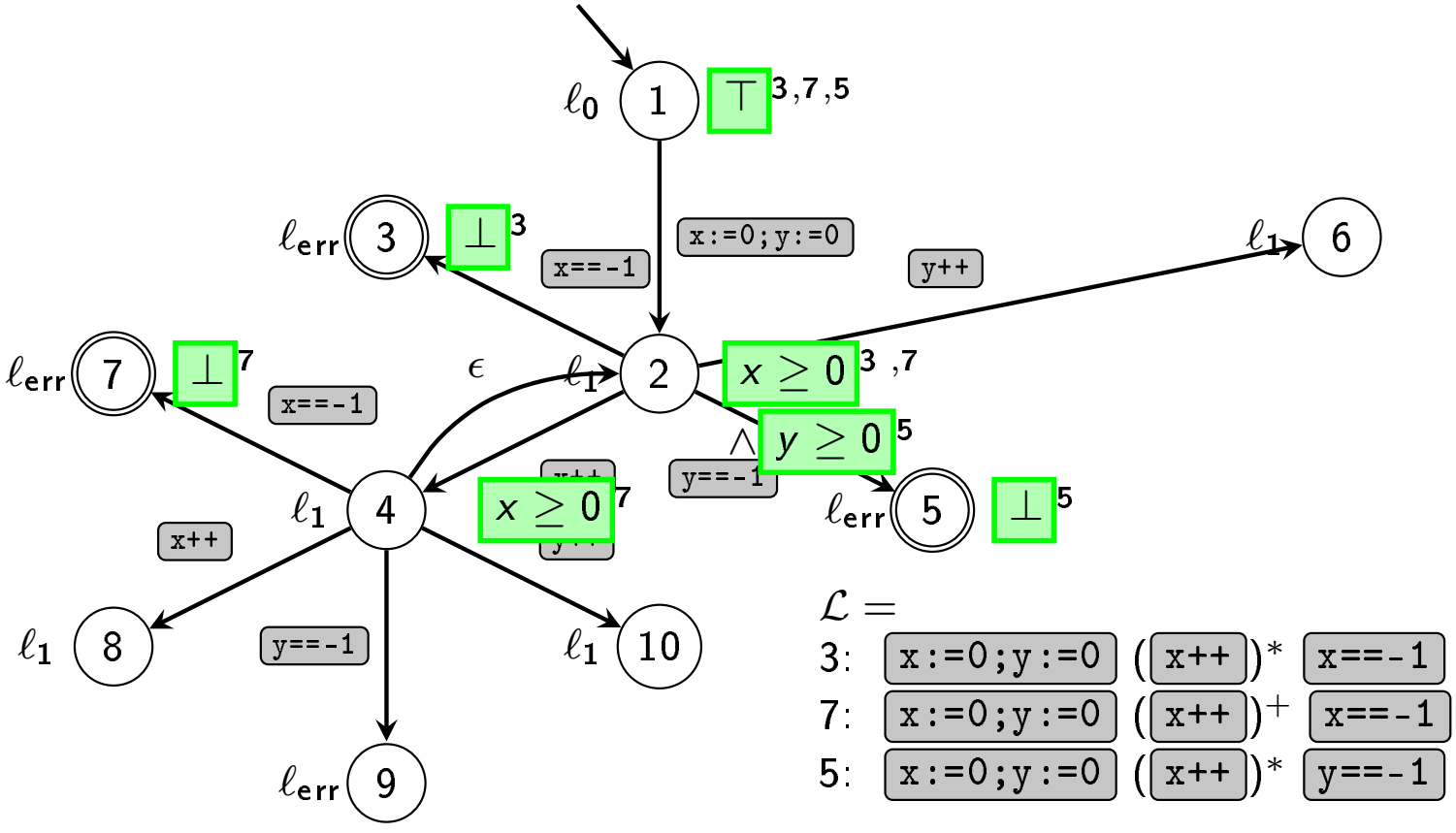
Unwindings als Automaten



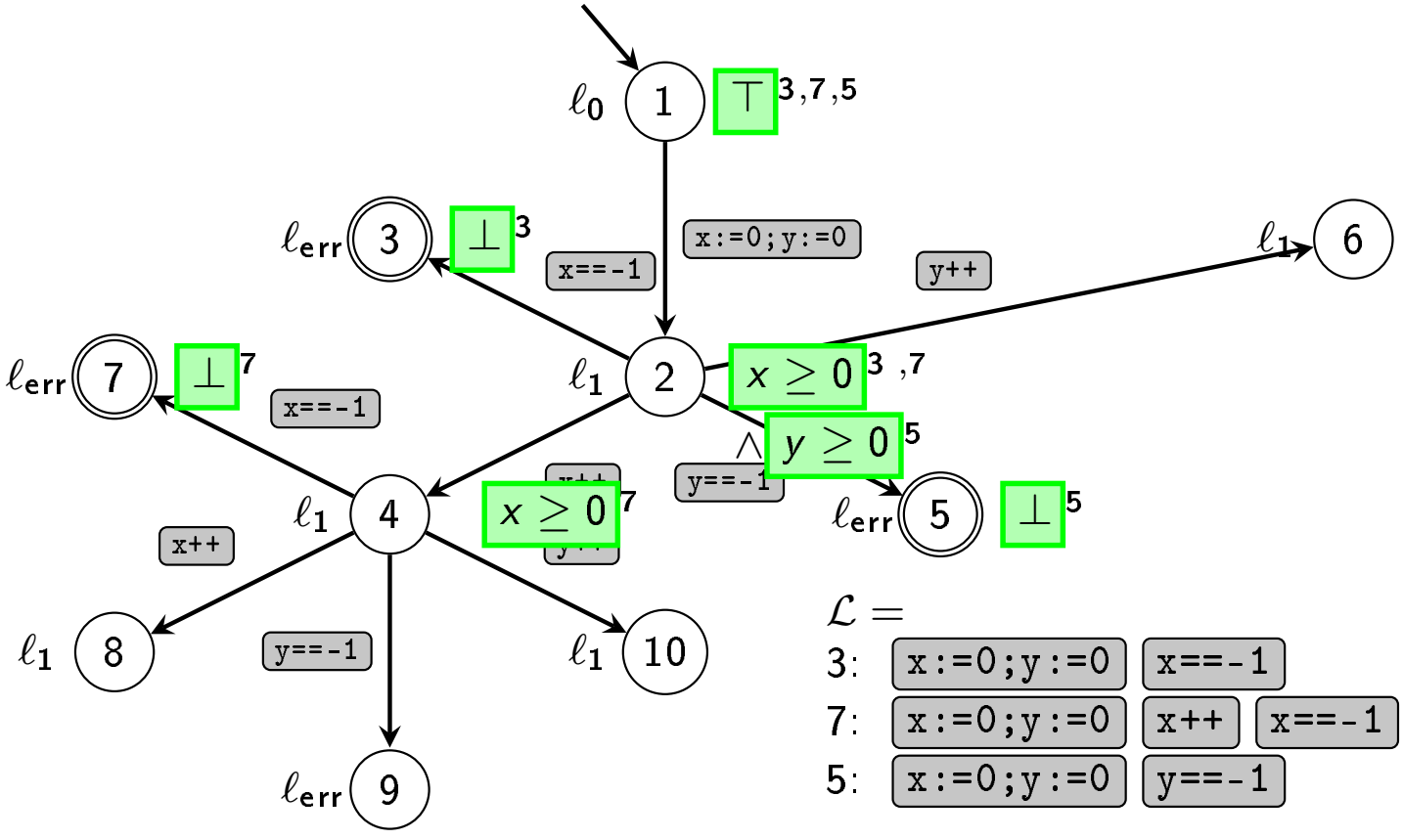
Unwindings als Automaten



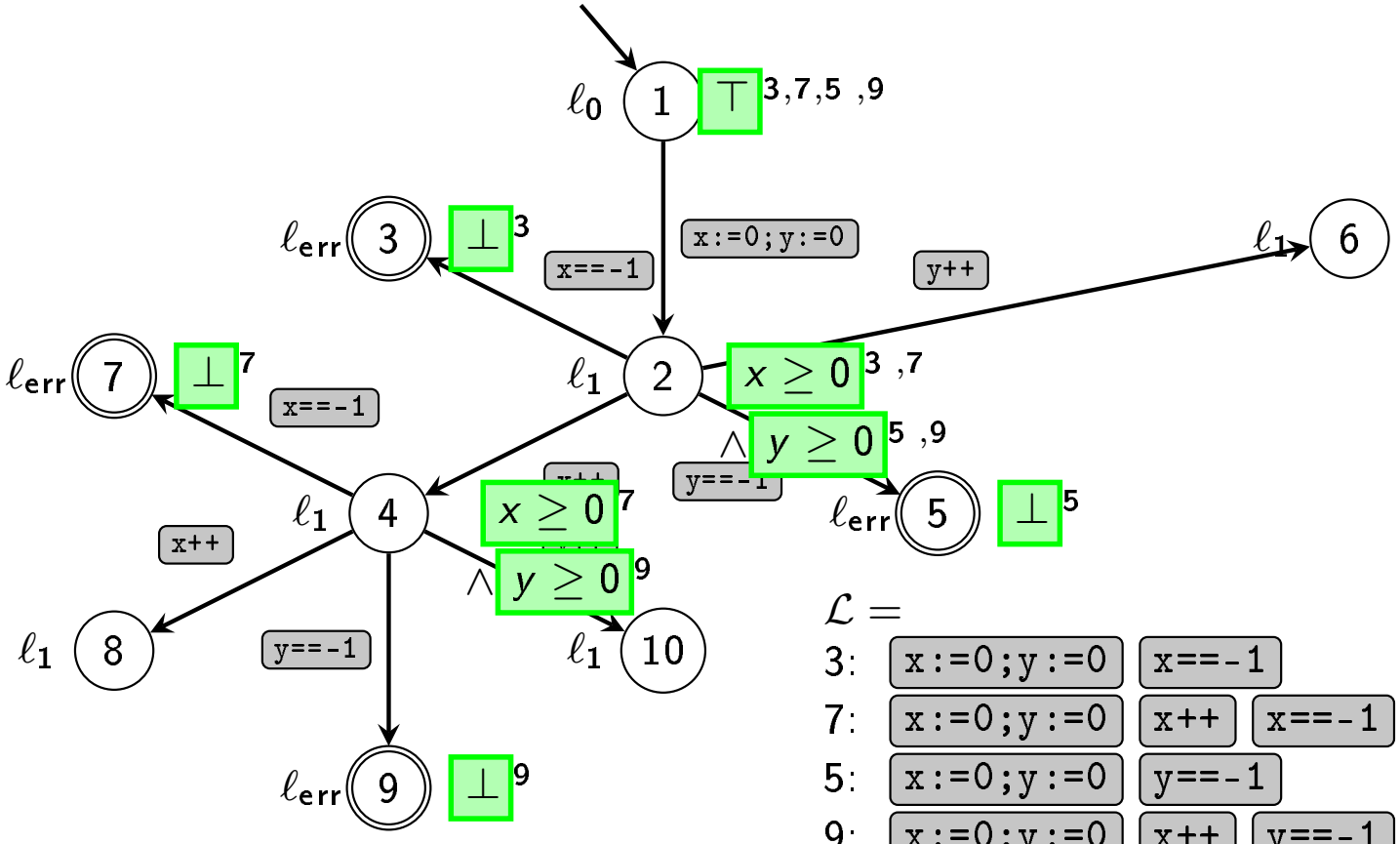
Unwindings als Automaten



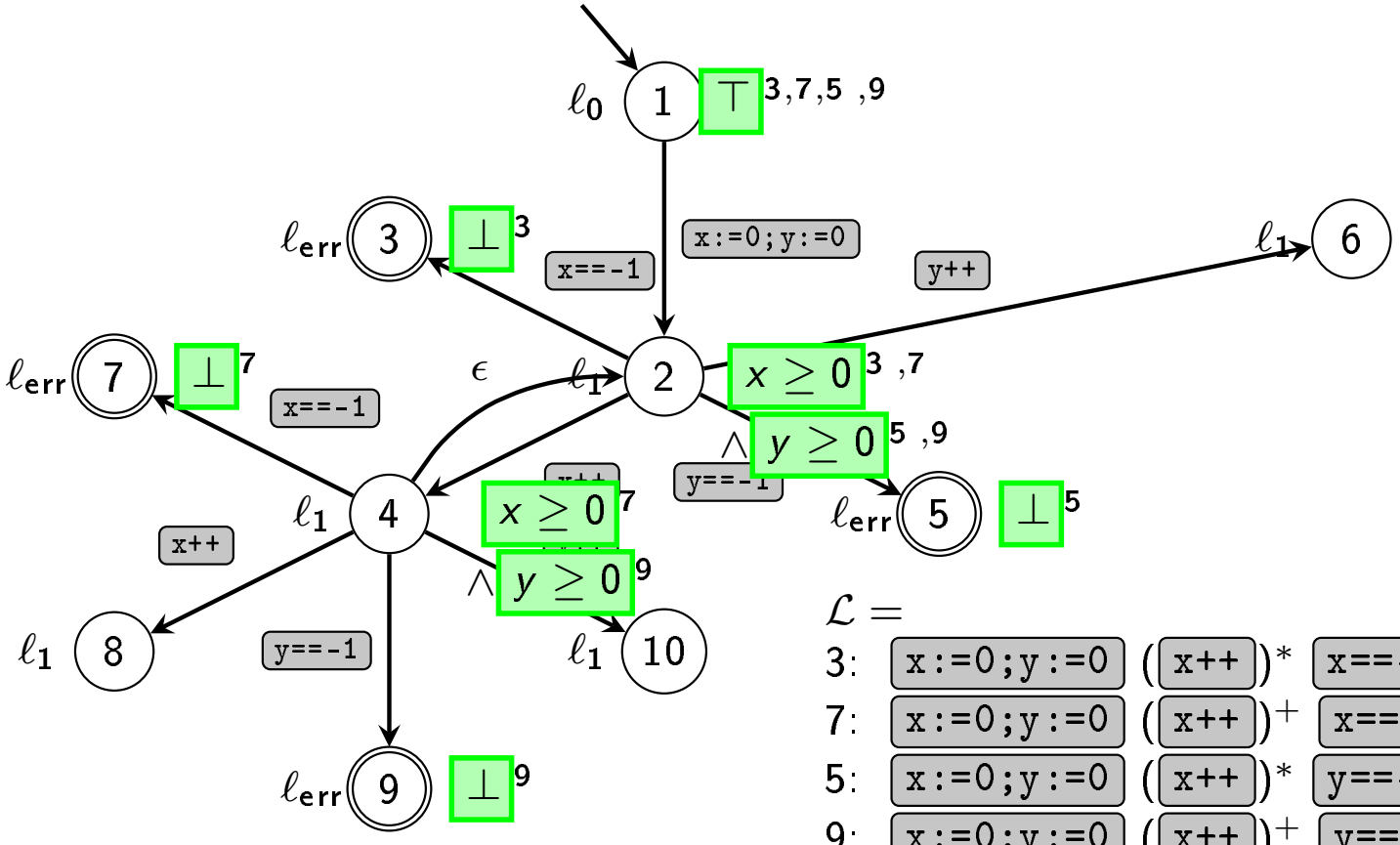
Unwindings als Automaten



Unwindings als Automaten



Unwindings als Automaten



$\mathcal{L} =$

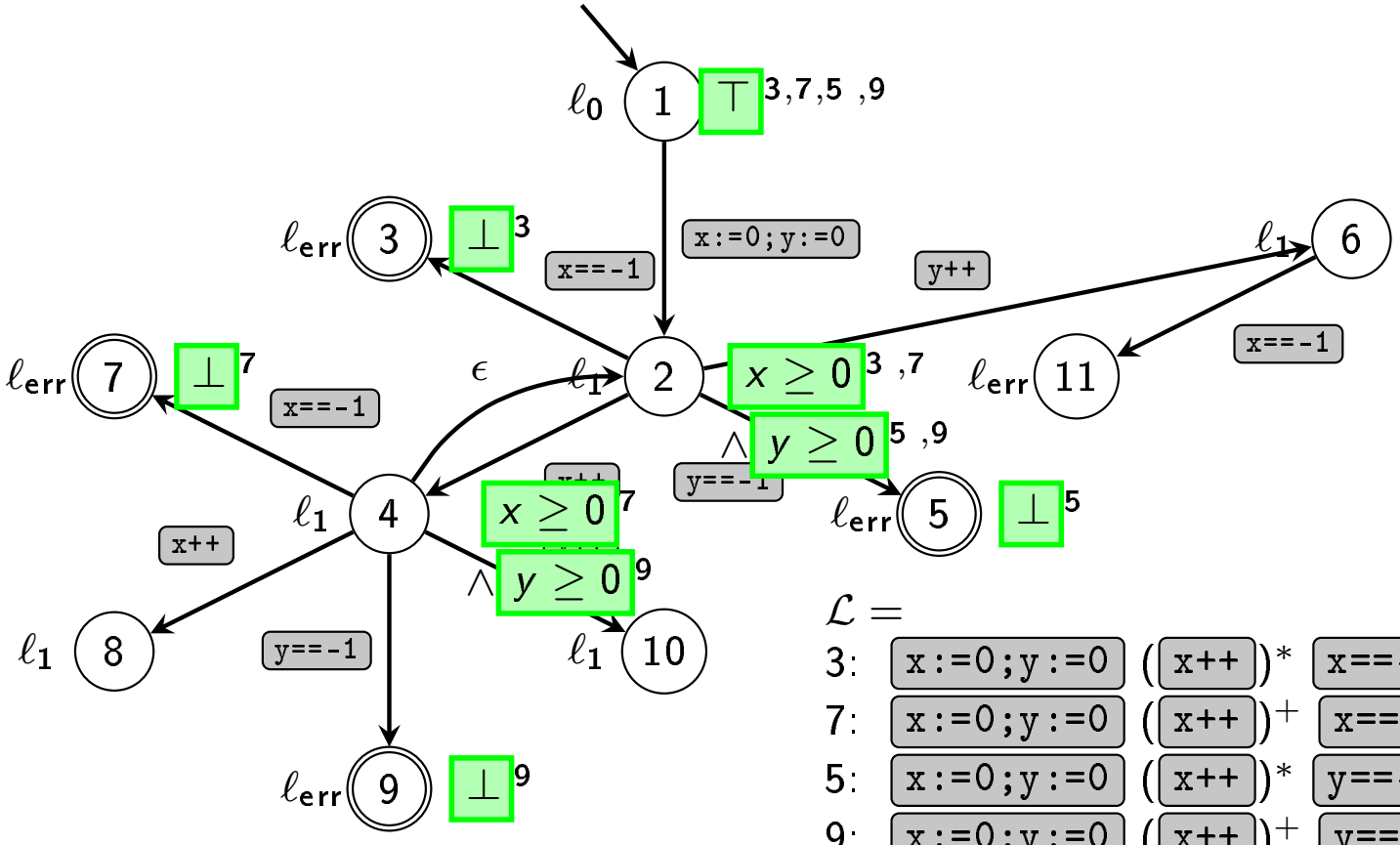
3: x:=0;y:=0 (x++)* x==-1

7: x:=0;y:=0 (x++)+ x==-1

5: x:=0;y:=0 (x++)* y==-1

9: x:=0;y:=0 (x++)+ y==-1

Unwindings als Automaten



$\mathcal{L} =$

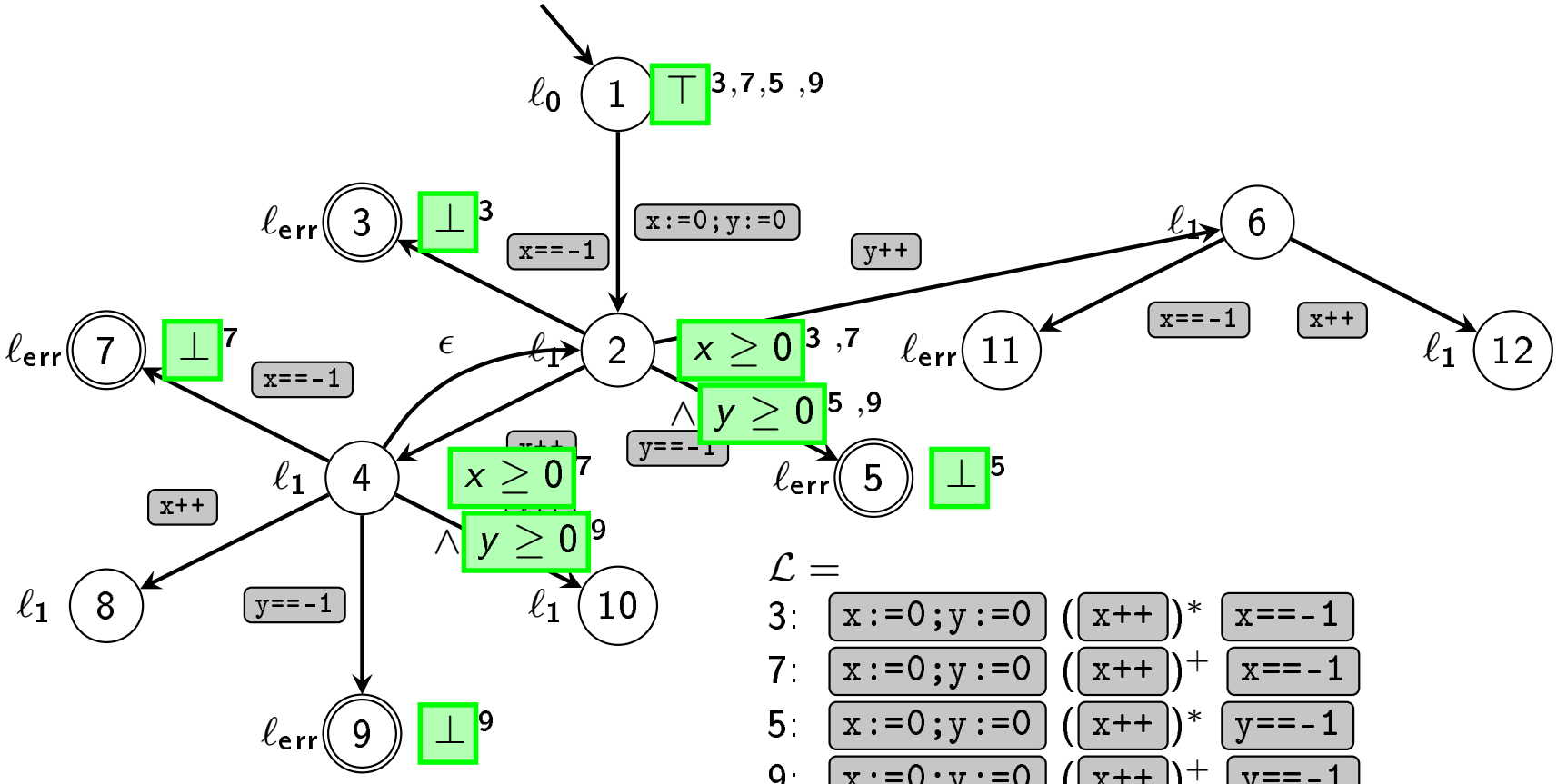
3: x:=0;y:=0 (x++)* x==-1

7: x:=0;y:=0 (x++)+ x==-1

5: x:=0;y:=0 (x++)* y==-1

9: x:=0;y:=0 (x++)+ y==-1

Unwindings als Automaten



$\mathcal{L} =$

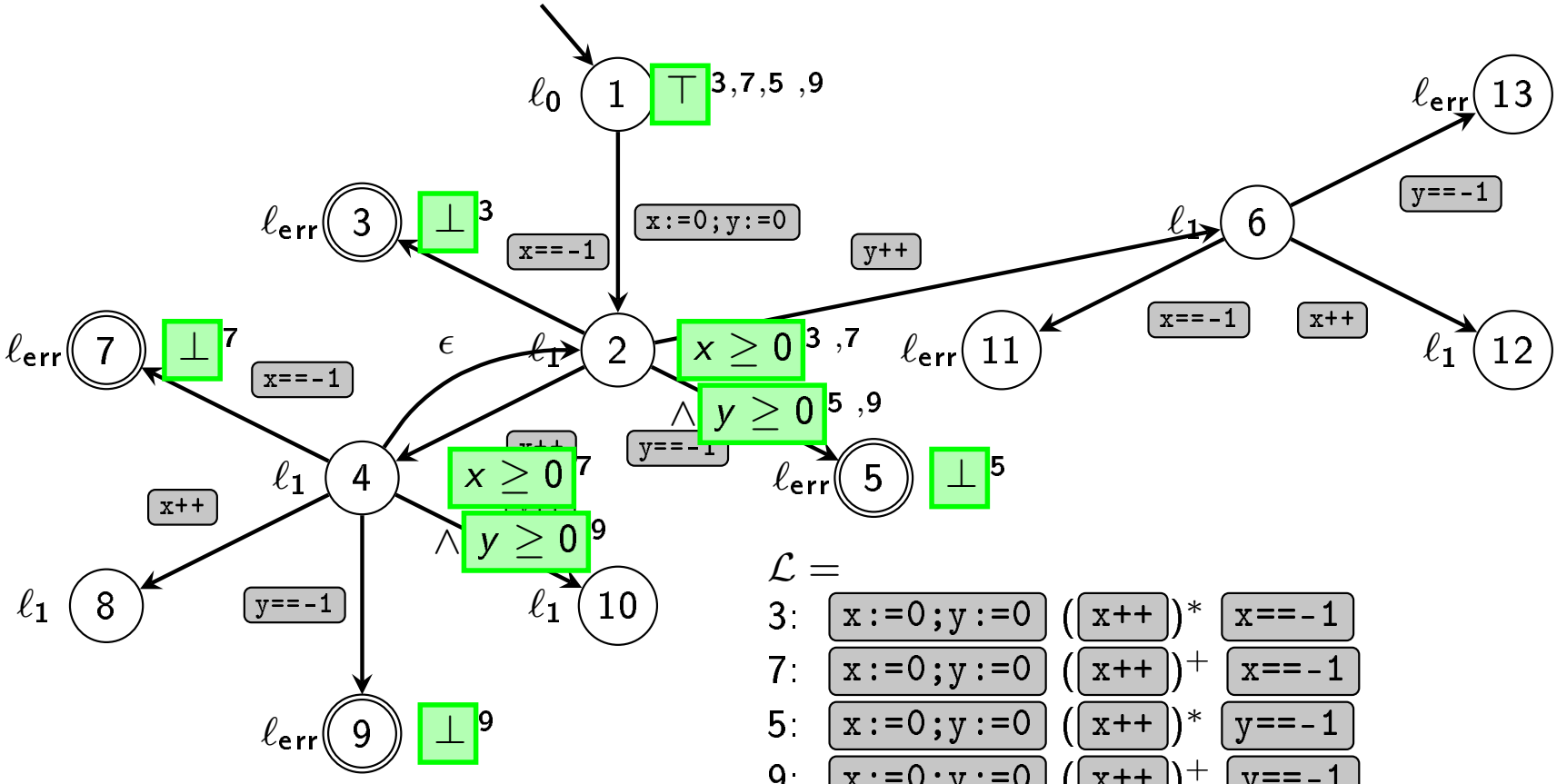
3: x:=0;y:=0 (x++)* x==-1

7: x:=0;y:=0 (x++)+ x==-1

5: x:=0;y:=0 (x++)* y==-1

9: x:=0;y:=0 (x++)+ y==-1

Unwindings als Automaten



$\mathcal{L} =$

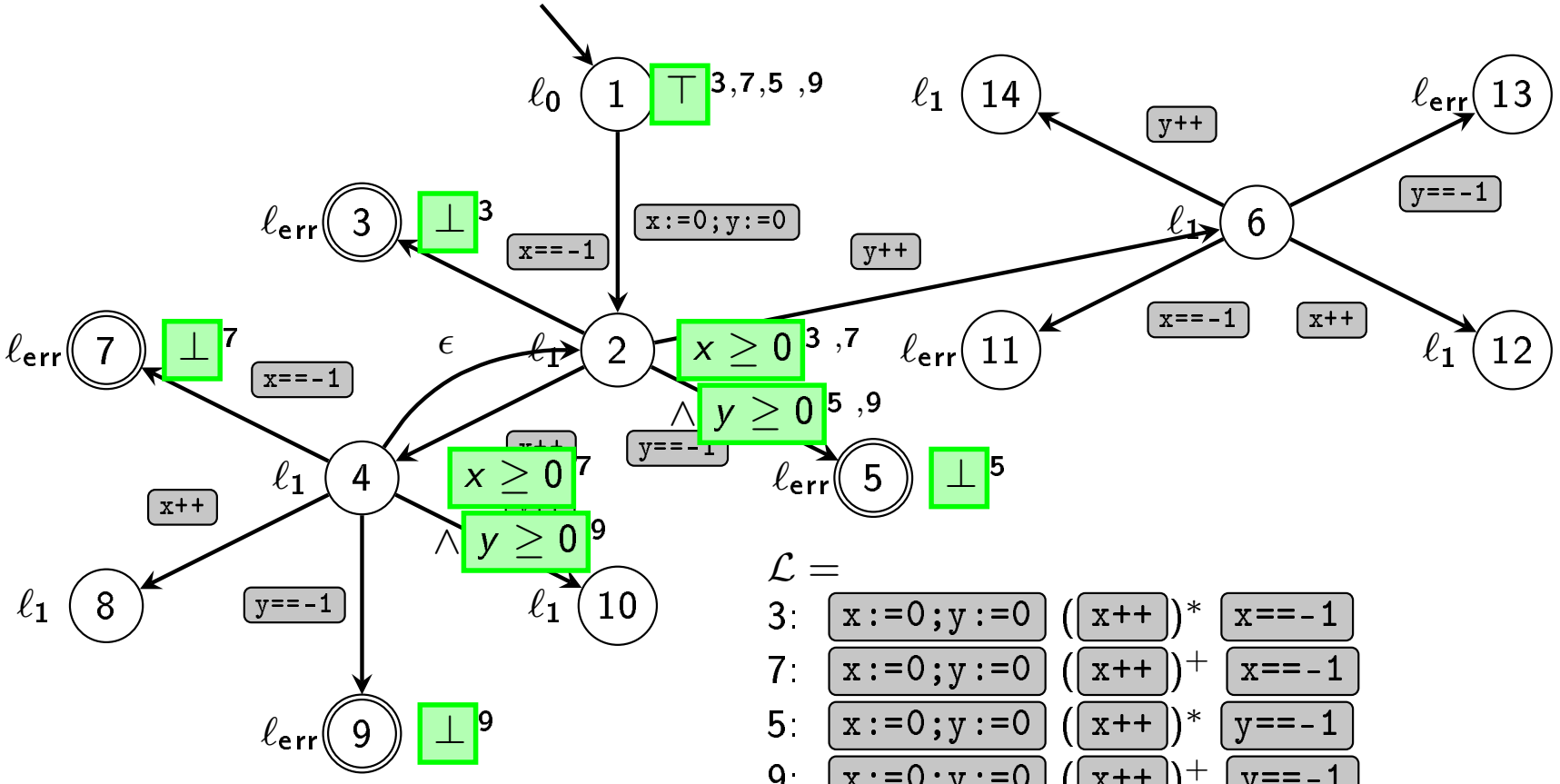
3: `x:=0;y:=0` (`x++`)* `x==-1`

7: `x:=0;y:=0` (`x++`)+ `x==-1`

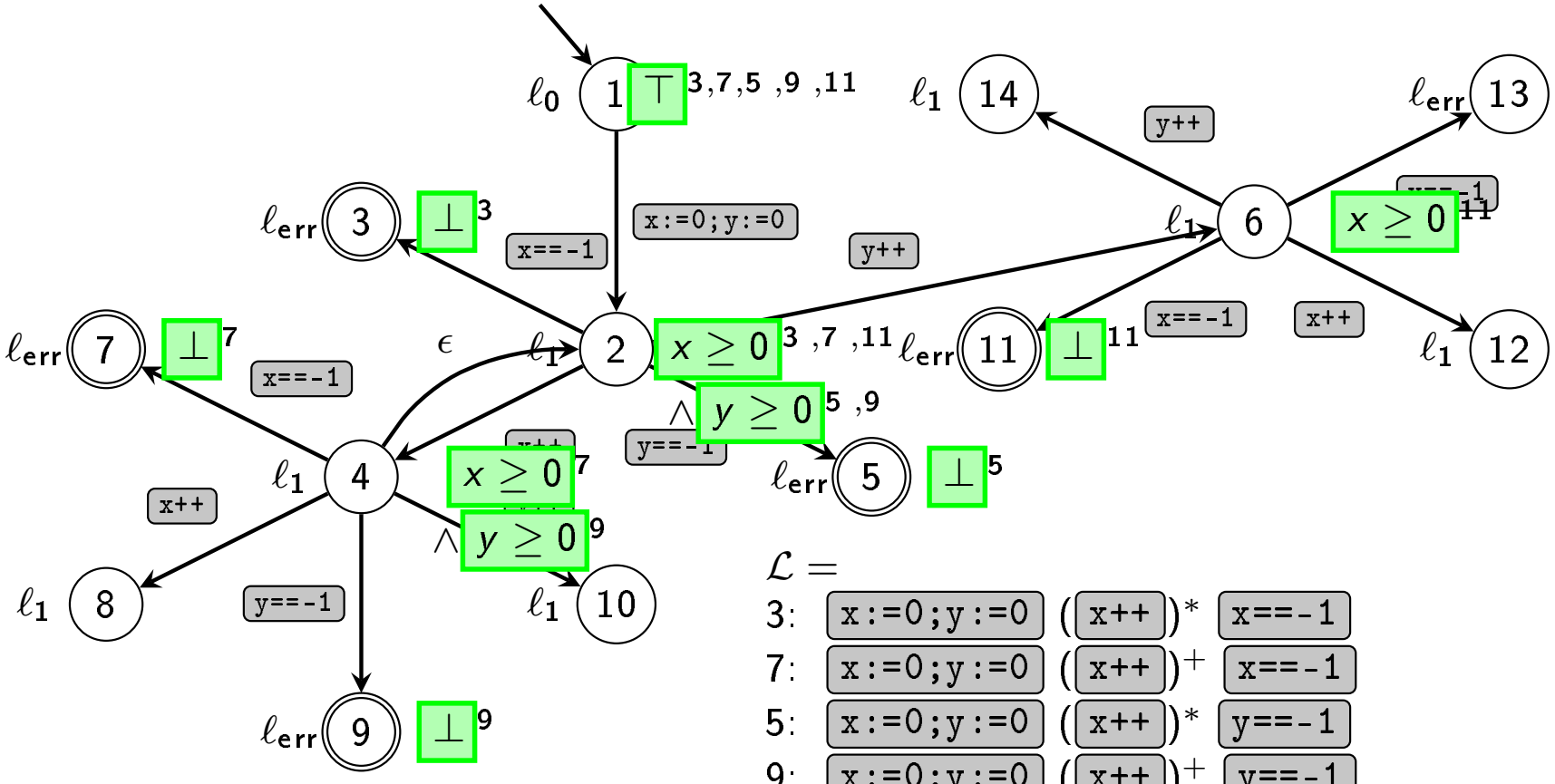
5: `x:=0;y:=0` (`x++`)* `y==-1`

9: `x:=0;y:=0` (`x++`)+ `y==-1`

Unwindings als Automaten



Unwindings als Automaten



$\mathcal{L} =$

3: x:=0;y:=0 (x++)* x==-1

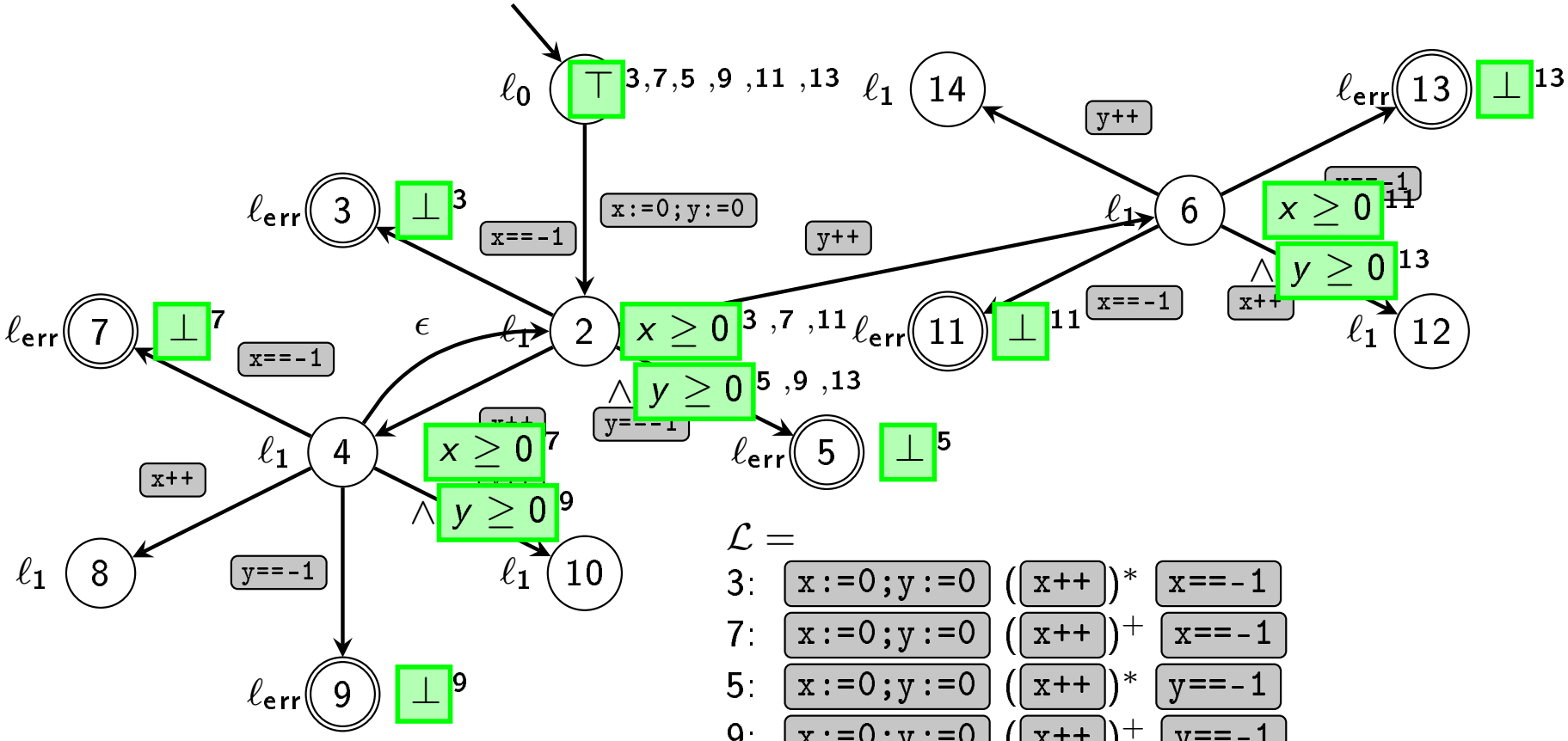
7: x:=0;y:=0 (x++)+ x==-1

5: x:=0;y:=0 (x++)* y==-1

9: x:=0;y:=0 (x++)+ y==-1

11: x:=0;y:=0 (x++)* y++ x==-1

Unwindings als Automaten



$\mathcal{L} =$

3: x:=0;y:=0 (x++)* x==-1

7: x:=0;y:=0 (x++)+ x==-1

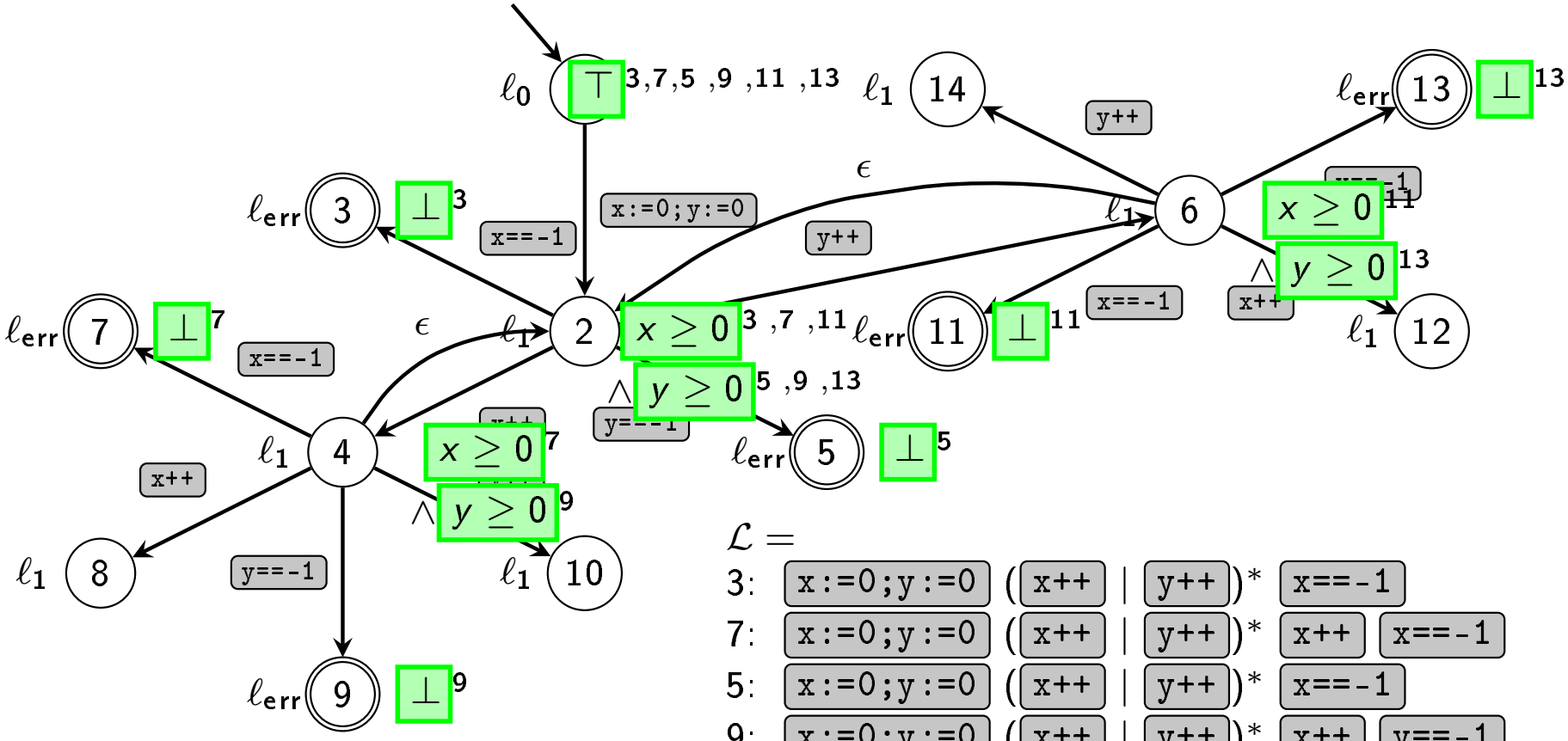
5: x:=0;y:=0 (x++)* y==-1

9: x:=0;y:=0 (x++)+ y==-1

11: x:=0;y:=0 (x++)* y++ x==-1

13: x:=0;y:=0 (x++)* y++ y==-1

Unwindings als Automaten



$\mathcal{L} =$

3: x:=0;y:=0 (x++ | y++)* x==-1

7: x:=0;y:=0 (x++ | y++)* x++ x==-1

5: x:=0;y:=0 (x++ | y++)* x==-1

9: x:=0;y:=0 (x++ | y++)* x++ y==-1

11: x:=0;y:=0 (x++ | y++)* y++ x==-1

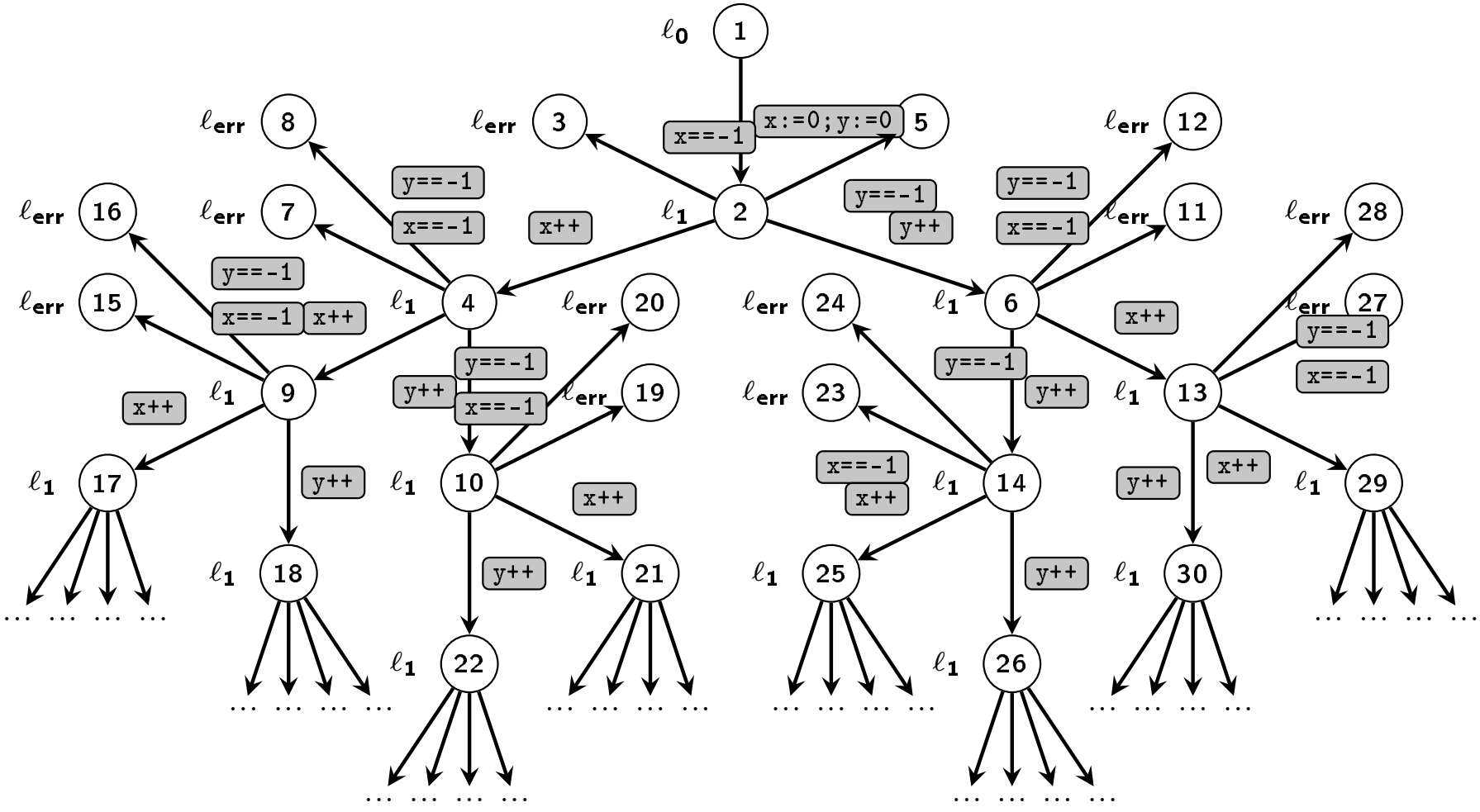
13: x:=0;y:=0 (x++ | y++)* y++ y==-1

Trace Abstraction auf Unwindings

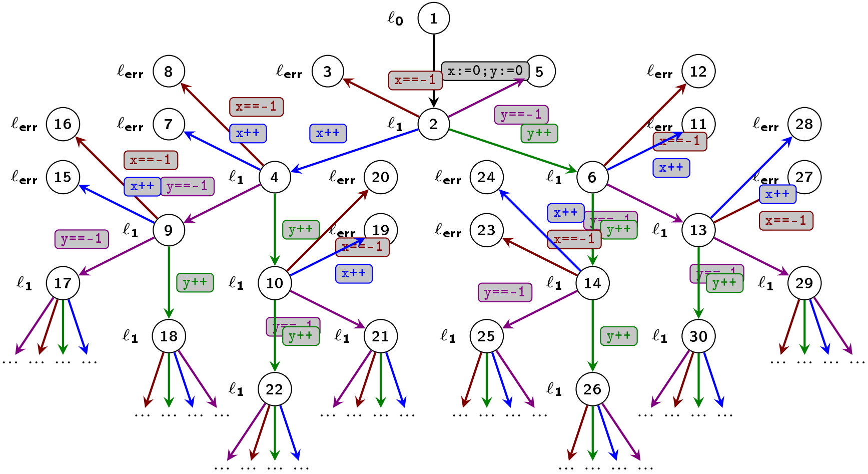
Die Unwindings von IMPACT entsprechen teilweisen Abwicklungen des Kontrollflussgraphen.

- Stelle den Kontrollflussgraphen als Unwinding dar.
- Streiche für jeden Iterationsschritt der Trace abstraction die Error locations, die als infeasible erkannt wurden.

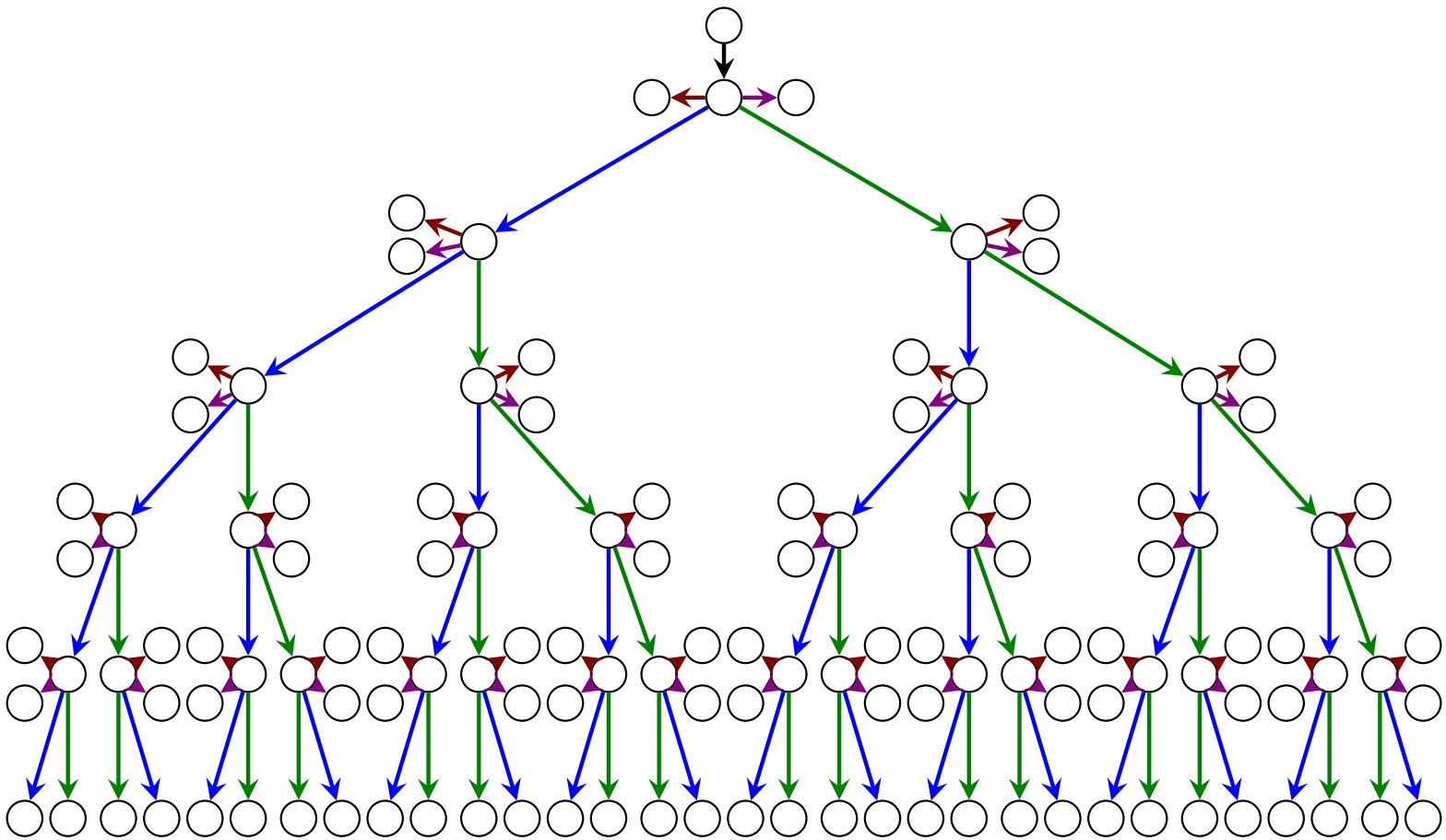
Trace Abstraction auf Unwindings



Trace Abstraction auf Unwindings

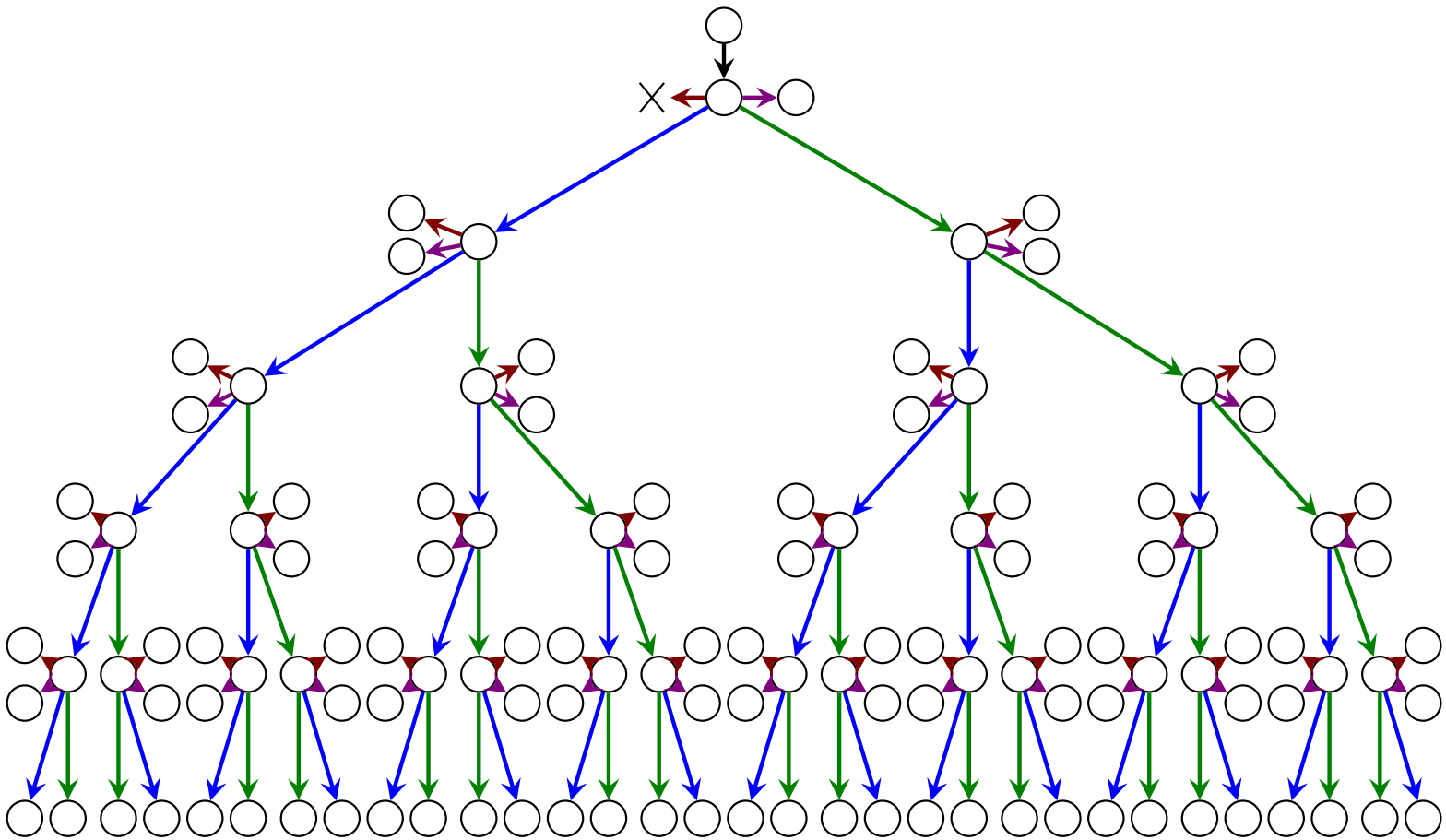


$\cup \mathcal{L}(A_I) =$
$\emptyset$



$\cup \mathcal{L}(A_I) =$

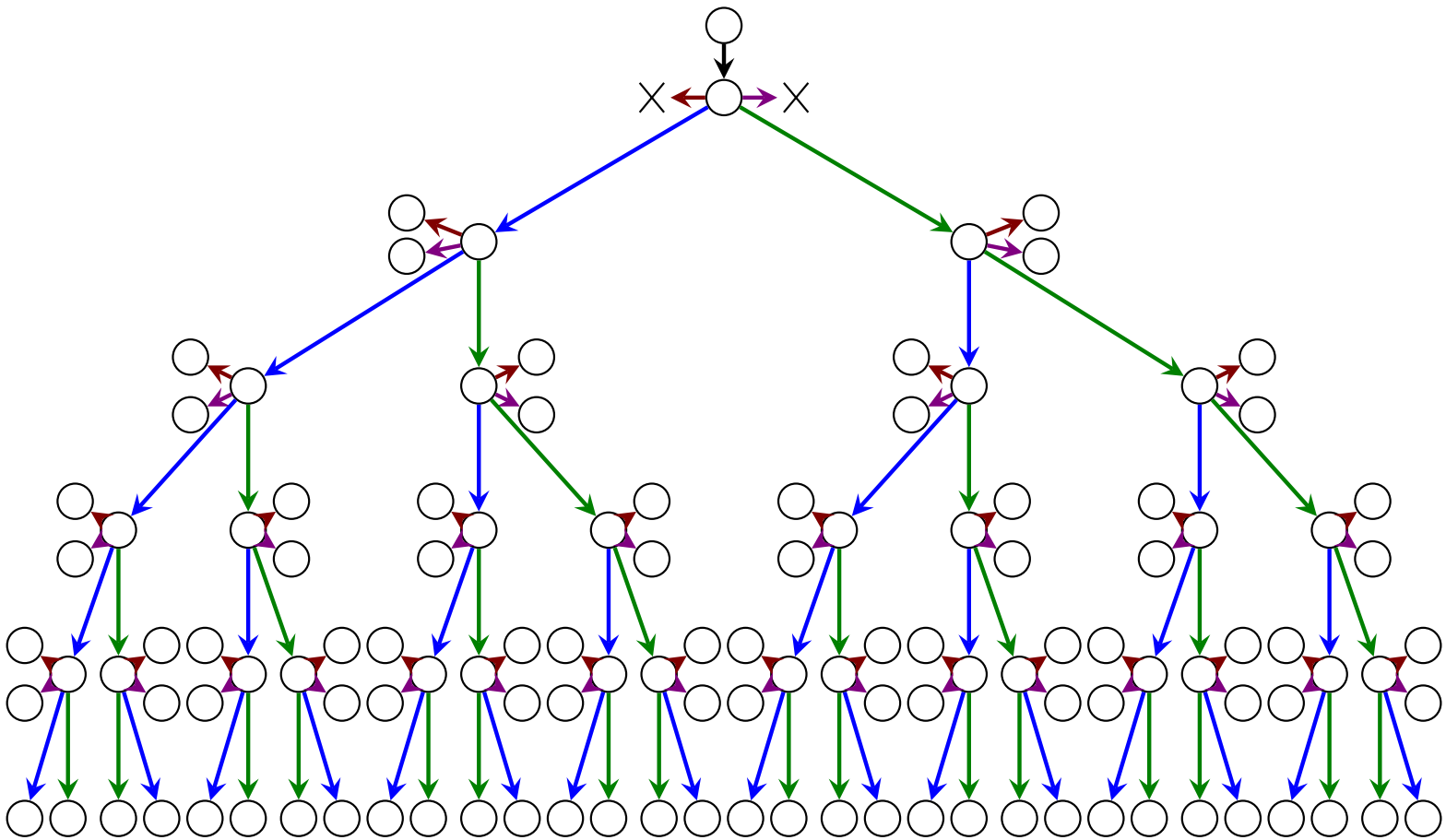
1: init x==-1



$\cup\mathcal{L}(A_I) =$

1: init x==-1

2: init y==-1

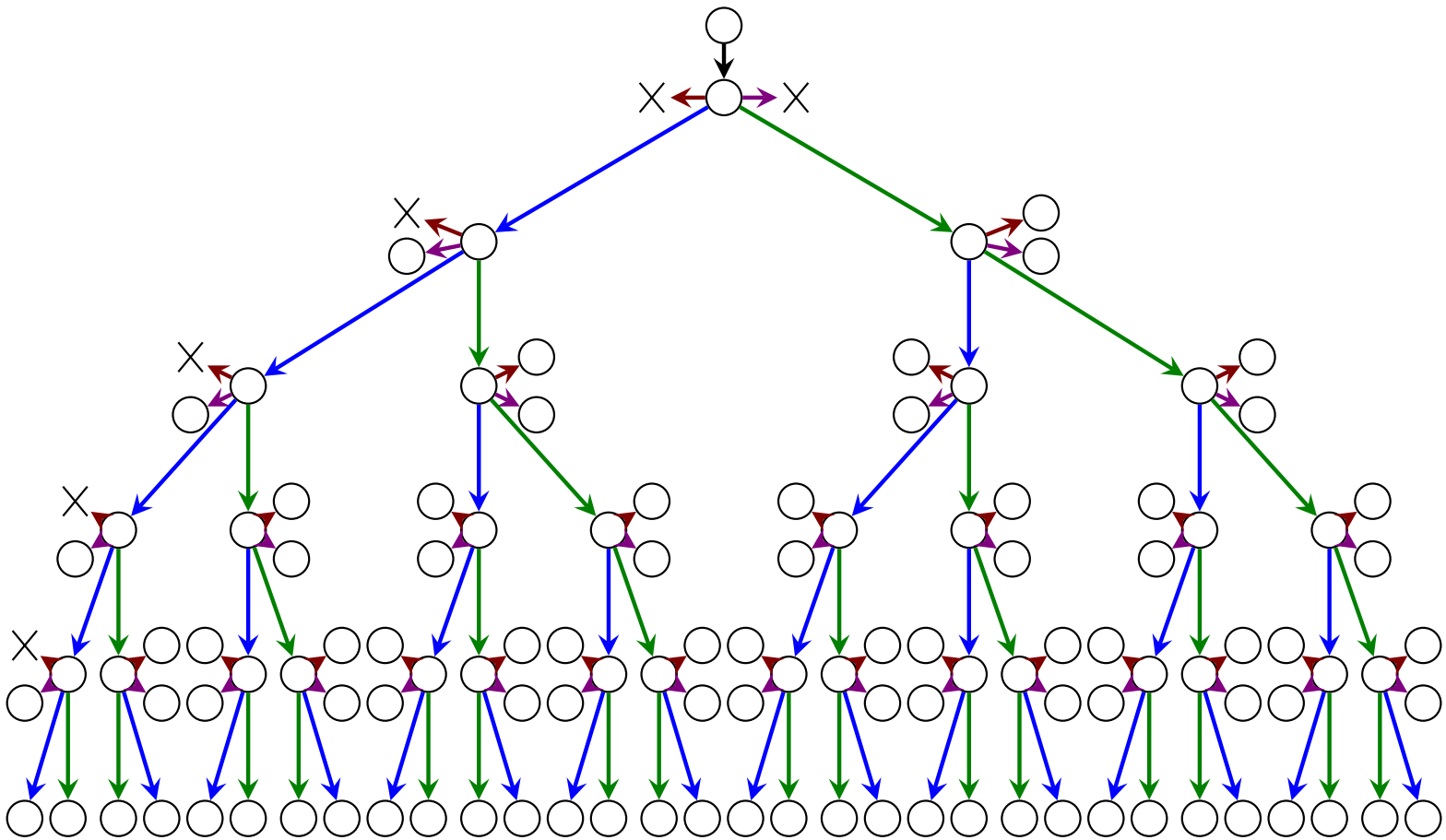


$\cup \mathcal{L}(A_I) =$

1: init x==-1

2: init y==-1

3: init (x++)$^+$ x==-1



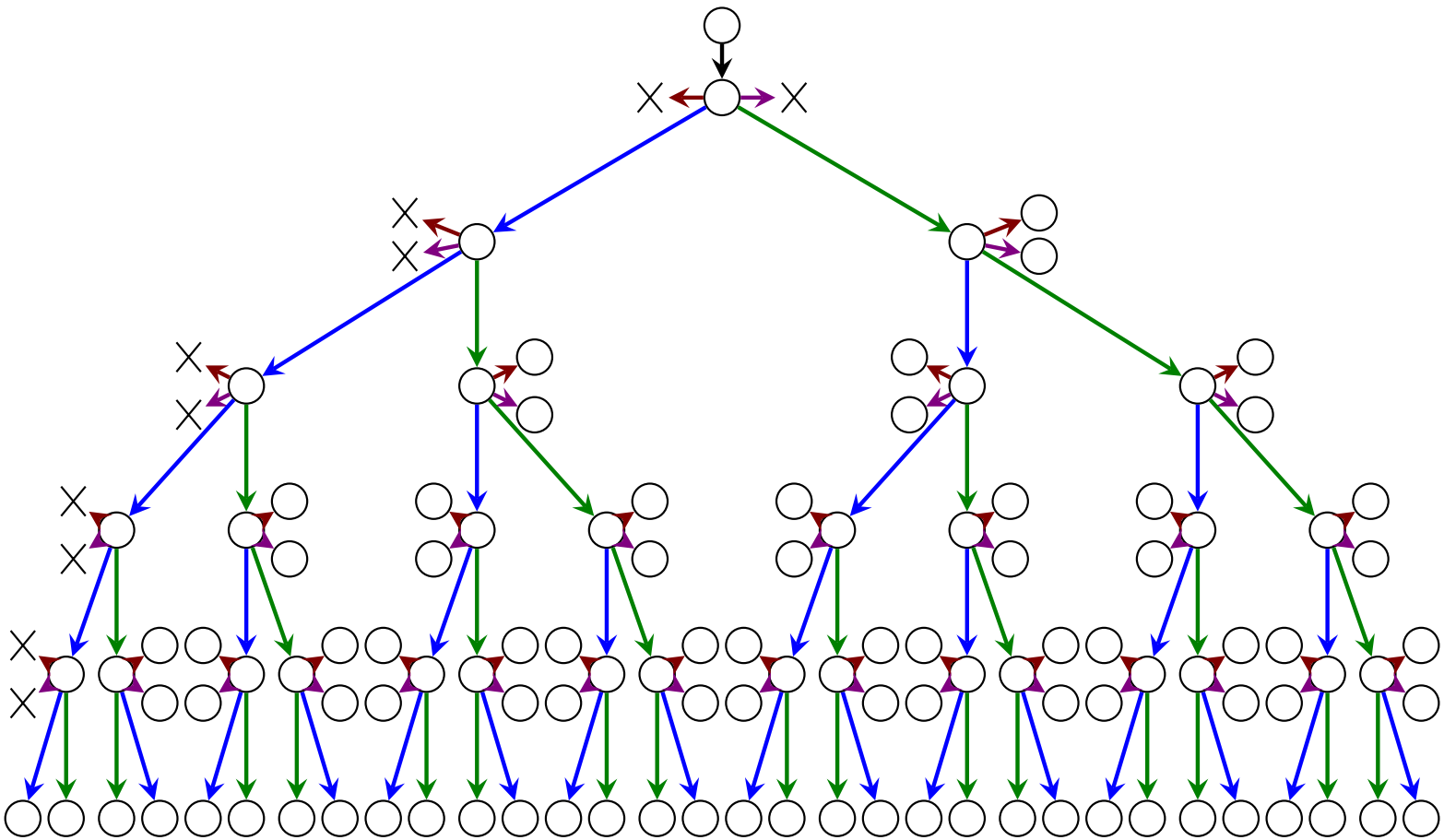
$\cup \mathcal{L}(A_I) =$

1: init x==-1

2: init y==-1

3: init (x++)$^+$ x==-1

4: init (x++)$^+$ y==-1



$\cup \mathcal{L}(A_I) =$

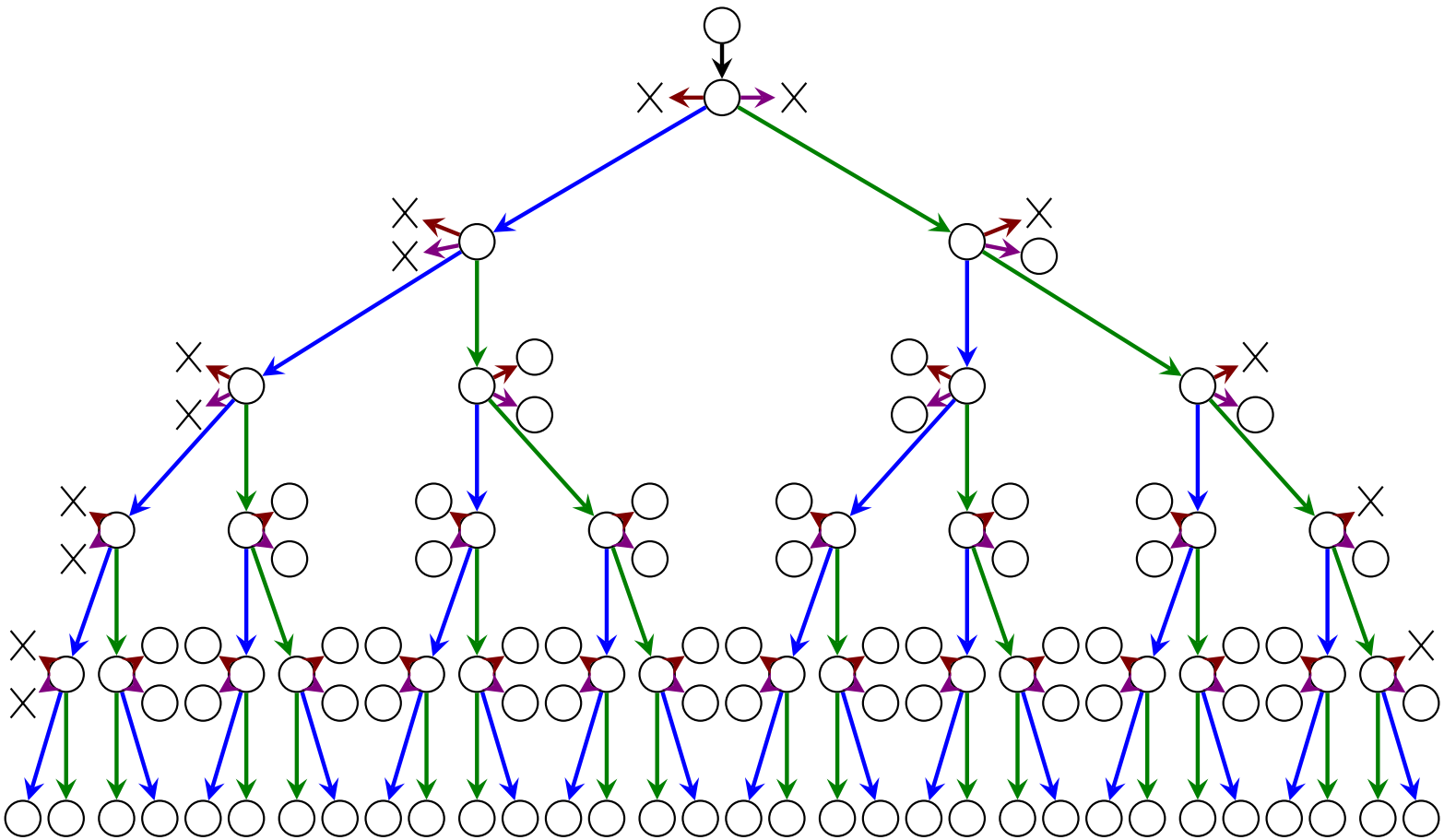
1: init x==-1

2: init y==-1

3: init (x++)$^+$ x==-1

4: init (x++)$^+$ y==-1

5: init (y++)$^+$ x==-1



$\cup \mathcal{L}(A_I) =$

1: init x==-1

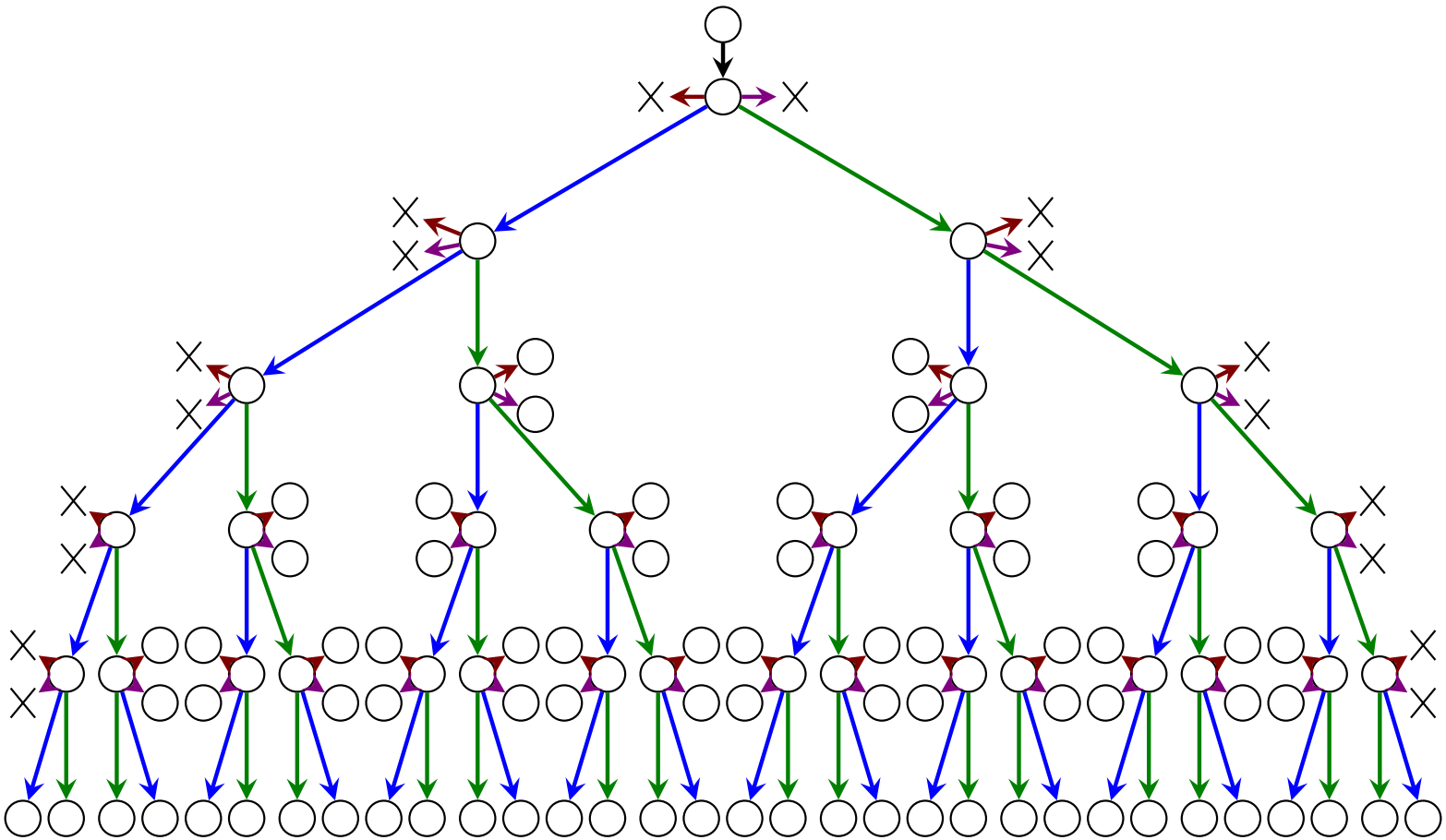
2: init y==-1

3: init (x++)$^+$ x==-1

4: init (x++)$^+$ y==-1

5: init (y++)$^+$ x==-1

6: init (y++)$^+$ y==-1



$\cup \mathcal{L}(A_I) =$

1: init x==-1

2: init y==-1

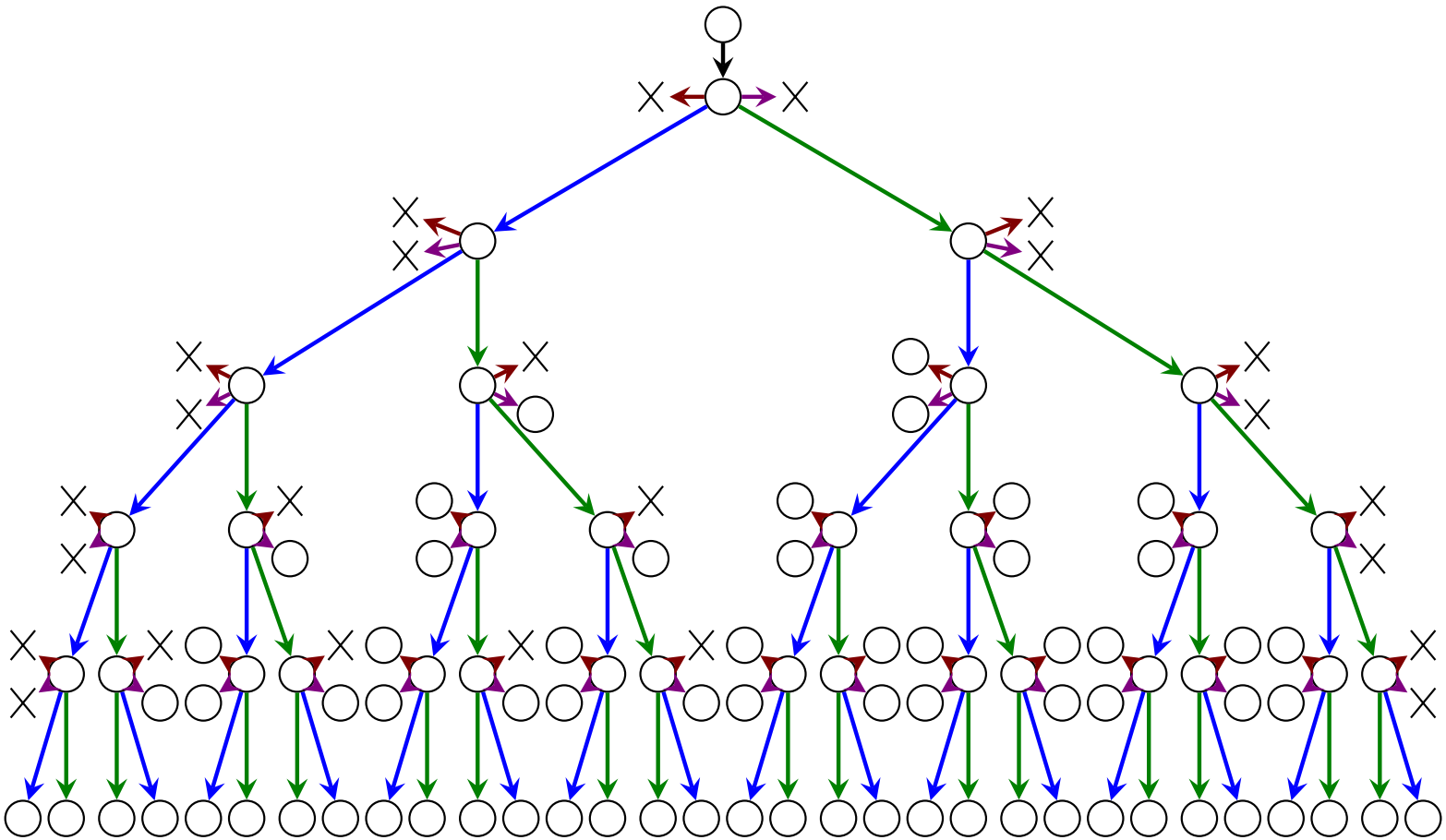
3: init (x++)$^+$ x==-1

4: init (x++)$^+$ y==-1

5: init (y++)$^+$ x==-1

6: init (y++)$^+$ y==-1

7: init ((x++)$^+$ (y++)$^+$)$^+$ x==-1



$\cup \mathcal{L}(A_I) =$

1: init x==-1

2: init y==-1

3: init (x++)$^+$ x==-1

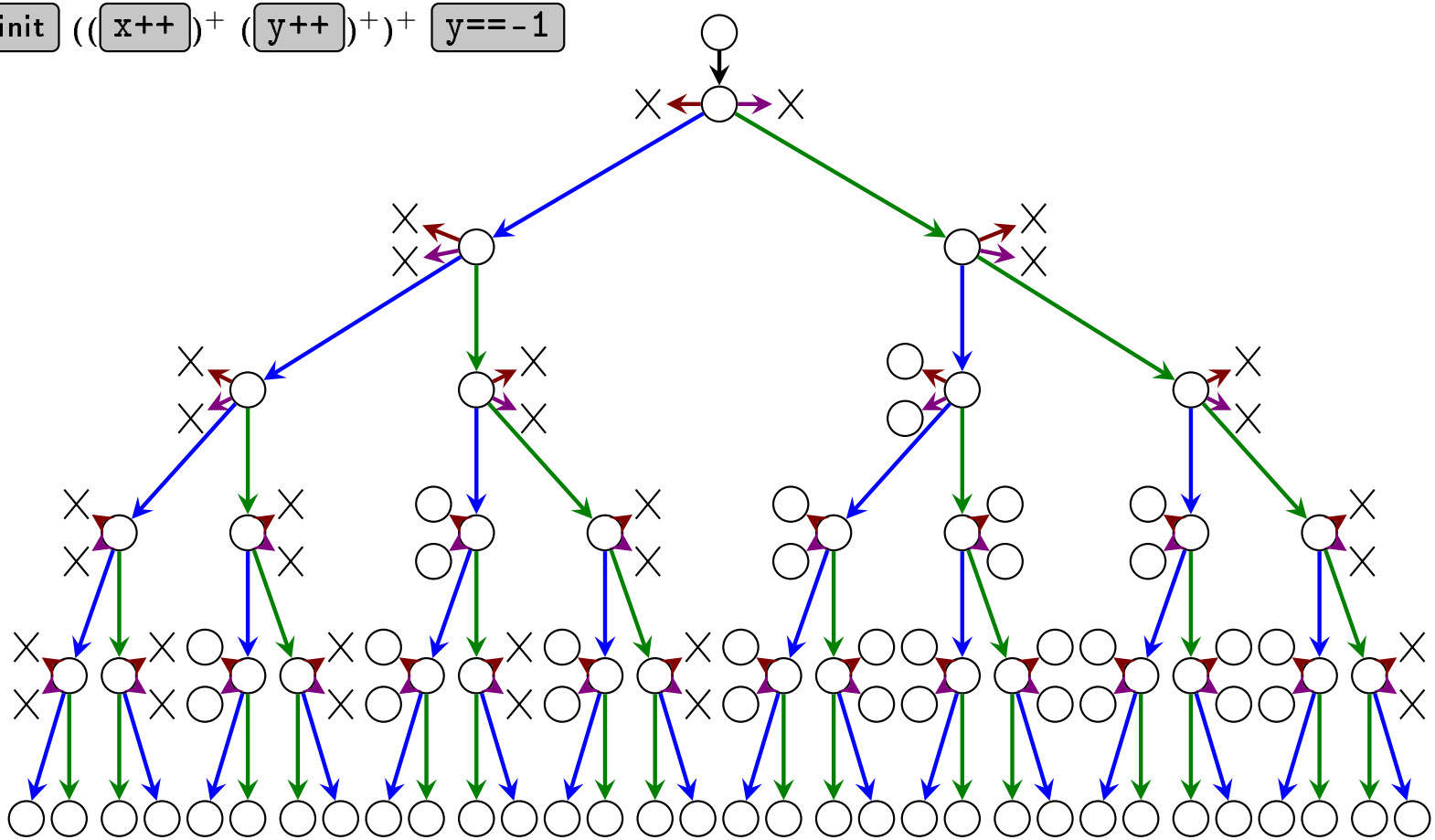
4: init (x++)$^+$ y==-1

5: init (y++)$^+$ x==-1

6: init (y++)$^+$ y==-1

7: init ((x++)$^+$ (y++)$^+$)$^+$ x==-1

8: init ((x++)$^+$ (y++)$^+$)$^+$ y==-1



$\cup \mathcal{L}(A_I) =$

1: init x==-1

2: init y==-1

3: init (x++)$^+$ x==-1

4: init (x++)$^+$ y==-1

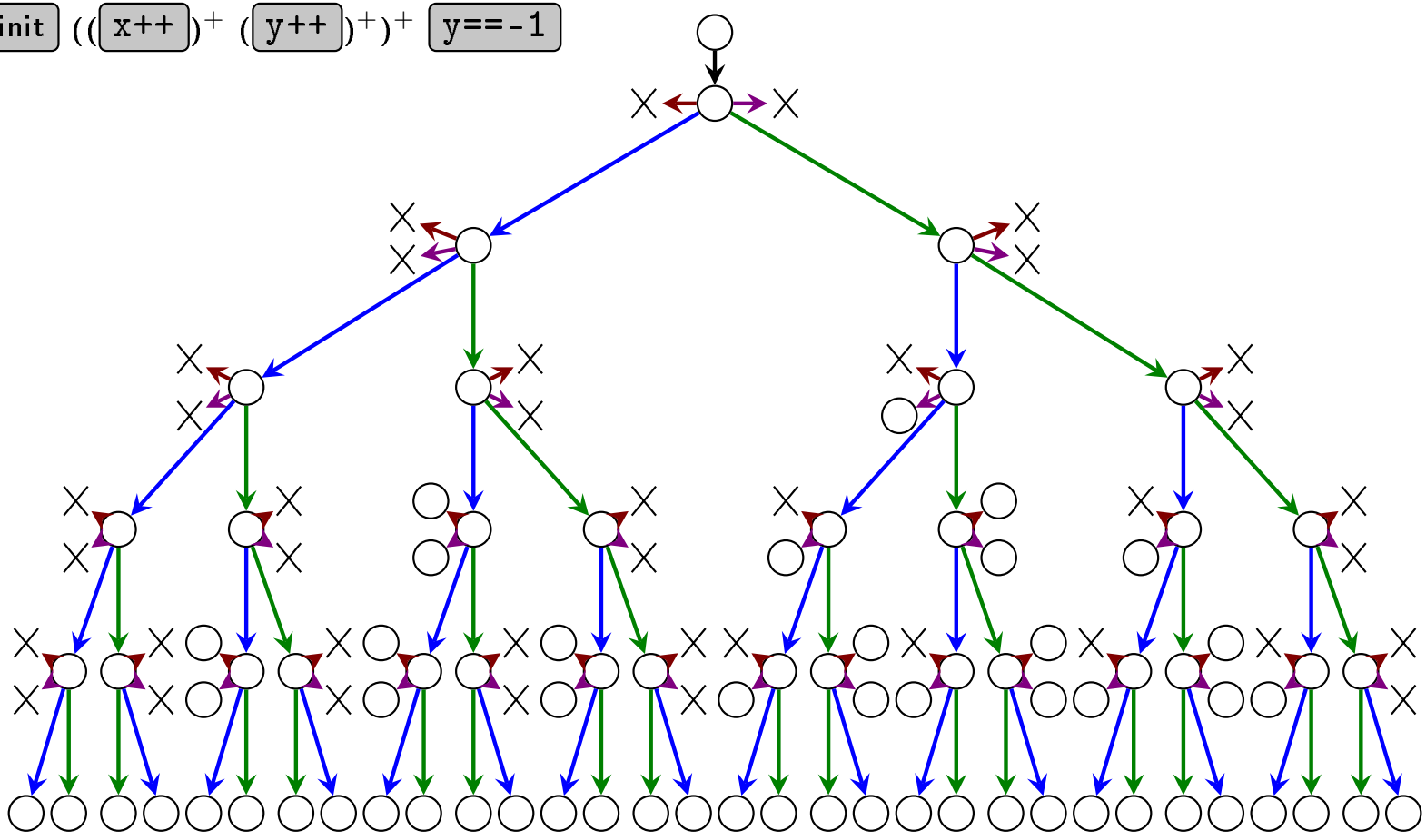
5: init (y++)$^+$ x==-1

6: init (y++)$^+$ y==-1

7: init ((x++)$^+$ (y++)$^+$)$^+$ x==-1

8: init ((x++)$^+$ (y++)$^+$)$^+$ y==-1

9: init ((y++)$^+$ (x++)$^+$)$^+$ x==-1



$\cup \mathcal{L}(A_I) =$

1: init x==-1

2: init y==-1

3: init (x++)$^+$ x==-1

4: init (x++)$^+$ y==-1

5: init (y++)$^+$ x==-1

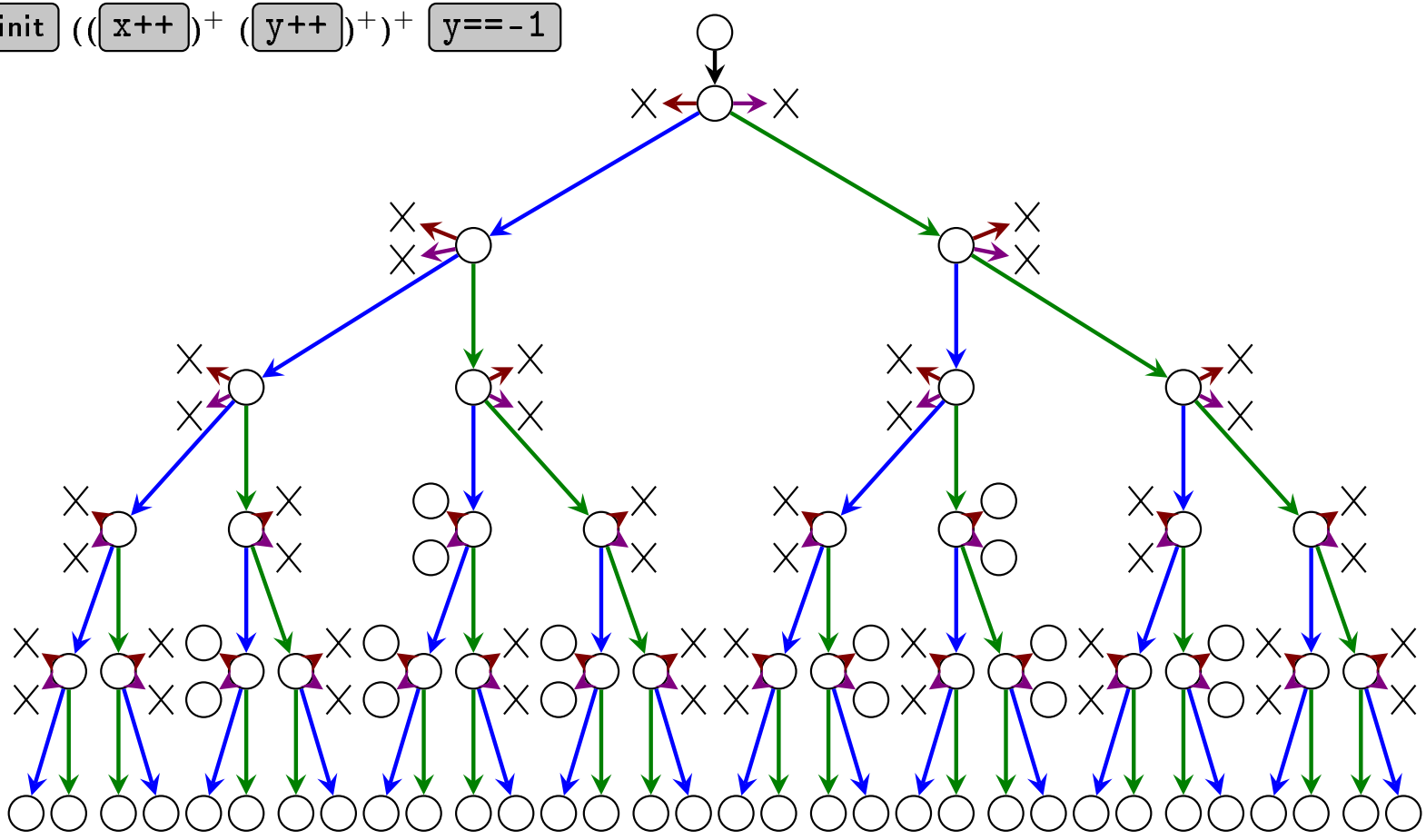
6: init (y++)$^+$ y==-1

7: init ((x++)$^+$ (y++)$^+$)$^+$ x==-1

8: init ((x++)$^+$ (y++)$^+$)$^+$ y==-1

9: init ((y++)$^+$ (x++)$^+$)$^+$ x==-1

10: init ((y++)$^+$ (x++)$^+$)$^+$ y==-1



$\cup \mathcal{L}(A_I) =$

1: init x==-1

2: init y==-1

3: init (x++)$^+$ x==-1

4: init (x++)$^+$ y==-1

5: init (y++)$^+$ x==-1

6: init (y++)$^+$ y==-1

7: init ((x++)$^+$ (y++)$^+$)$^+$ x==-1

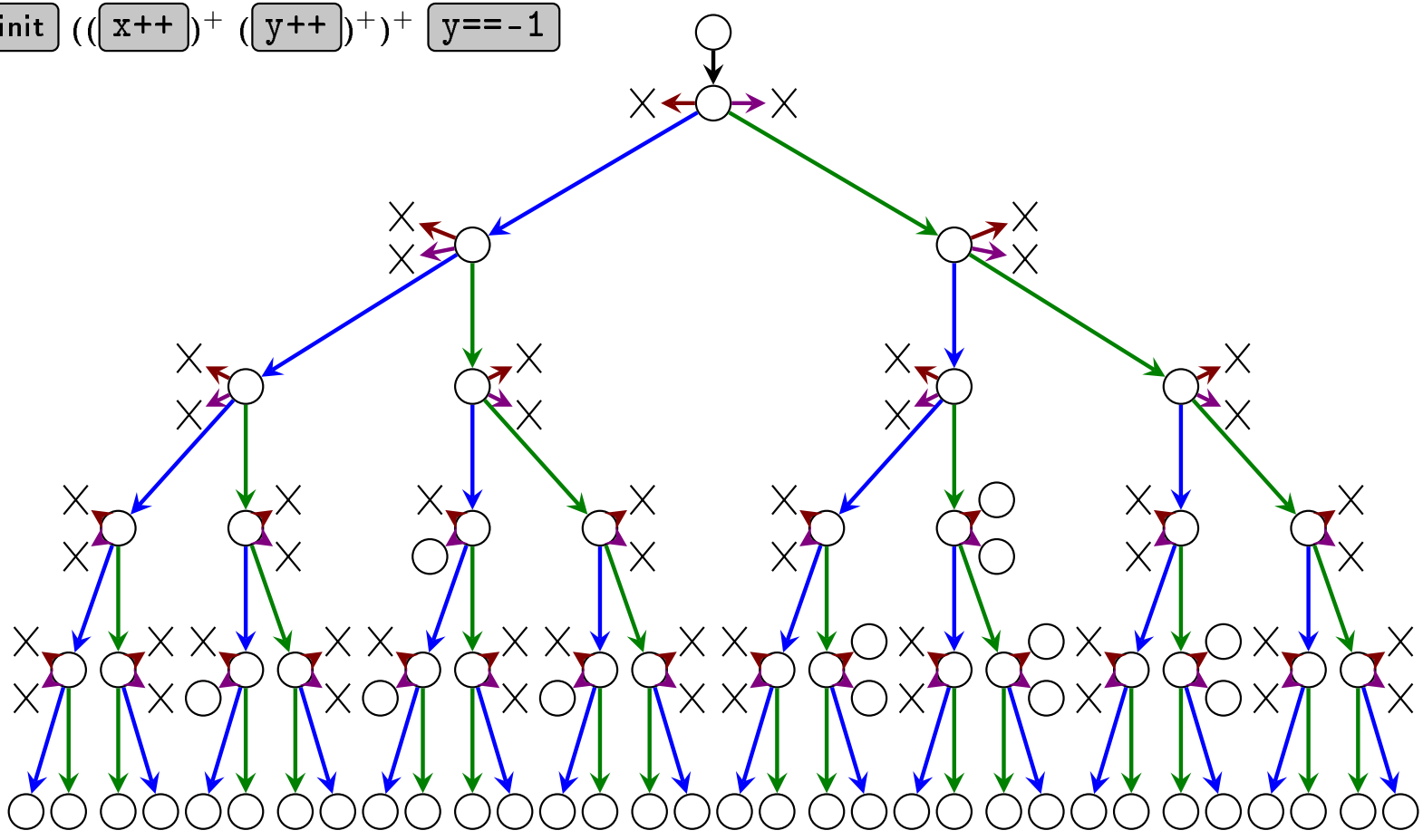
8: init ((x++)$^+$ (y++)$^+$)$^+$ y==-1

9: init ((y++)$^+$ (x++)$^+$)$^+$ x==-1

10: init ((y++)$^+$ (x++)$^+$)$^+$ y==-1

11: init ((x++)$^+$ ((y++)$^+$ (x++)$^+$)$^+$)$^+$ x==-1

init (((x++)$^+$ ((y++)$^+$)$^+$ (x++)$^+$)$^+$ x==-1



$\cup \mathcal{L}(A_I) =$

1: init x==-1

2: init y==-1

3: init (x++)$^+$ x==-1

4: init (x++)$^+$ y==-1

5: init (y++)$^+$ x==-1

6: init (y++)$^+$ y==-1

7: init ((x++)$^+$ (y++)$^+$)$^+$ x==-1

8: init ((x++)$^+$ (y++)$^+$)$^+$ y==-1

9: init ((y++)$^+$ (x++)$^+$)$^+$ x==-1

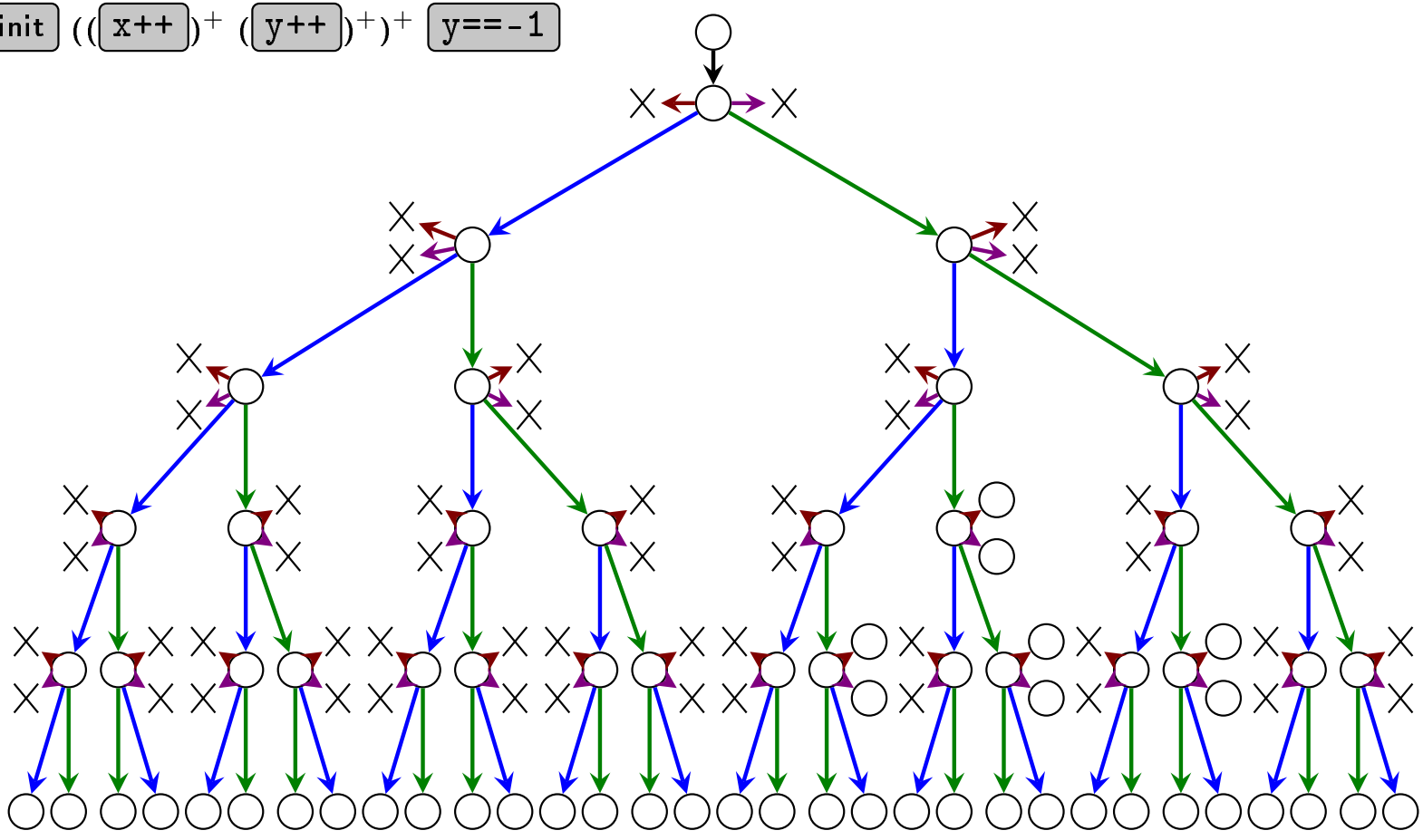
10: init ((y++)$^+$ (x++)$^+$)$^+$ y==-1

11: init ((x++)$^+$ ((y++)$^+$ (x++)$^+$)$^+$)$^+$ x==-1

init (((x++)$^+$ ((y++)$^+$)$^+$ (x++)$^+$)$^+$ x==-1

12: init ((x++)$^+$ ((y++)$^+$ (x++)$^+$)$^+$)$^+$ y==-1

init (((x++)$^+$ ((y++)$^+$)$^+$ (x++)$^+$)$^+$ y==-1



$\cup \mathcal{L}(A_I) =$

1: init x==-1

2: init y==-1

3: init (x++)$^+$ x==-1

4: init (x++)$^+$ y==-1

5: init (y++)$^+$ x==-1

6: init (y++)$^+$ y==-1

7: init ((x++)$^+$ (y++)$^+$)$^+$ x==-1

8: init ((x++)$^+$ (y++)$^+$)$^+$ y==-1

9: init ((y++)$^+$ (x++)$^+$)$^+$ x==-1

10: init ((y++)$^+$ (x++)$^+$)$^+$ y==-1

11: init ((x++)$^+$ ((y++)$^+$ (x++)$^+$)$^+$)$^+$ x==-1

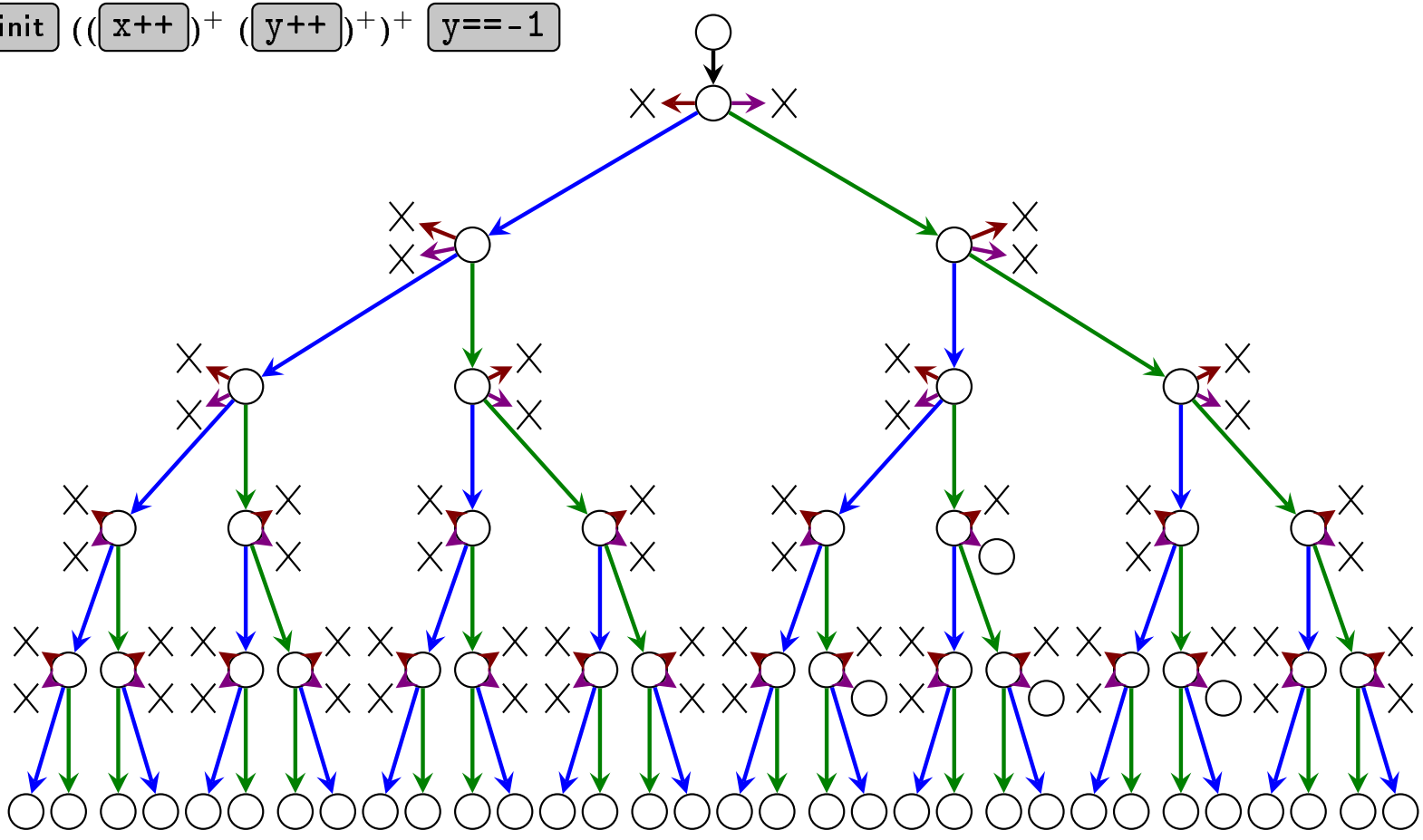
init (((x++)$^+$ ((y++)$^+$)$^+$ (x++)$^+$)$^+$ x==-1

12: init ((x++)$^+$ ((y++)$^+$ (x++)$^+$)$^+$)$^+$ y==-1

init (((x++)$^+$ ((y++)$^+$)$^+$ (x++)$^+$)$^+$ y==-1

13: init ((y++)$^+$ ((x++)$^+$ (y++)$^+$)$^+$)$^+$ x==-1

init (((y++)$^+$ ((x++)$^+$)$^+$ (y++)$^+$)$^+$ x==-1



$\cup \mathcal{L}(A_I) =$

1: init x==-1

2: init y==-1

3: init (x++)$^+$ x==-1

4: init (x++)$^+$ y==-1

5: init (y++)$^+$ x==-1

6: init (y++)$^+$ y==-1

7: init ((x++)$^+$ (y++)$^+$)$^+$ x==-1

8: init ((x++)$^+$ (y++)$^+$)$^+$ y==-1

9: init ((y++)$^+$ (x++)$^+$)$^+$ x==-1

10: init ((y++)$^+$ (x++)$^+$)$^+$ y==-1

11: init ((x++)$^+$ ((y++)$^+$ (x++)$^+$)$^+$)$^+$ x==-1

init (((x++)$^+$ ((y++)$^+$)$^+$ (x++)$^+$)$^+$ x==-1

12: init ((x++)$^+$ ((y++)$^+$ (x++)$^+$)$^+$)$^+$ y==-1

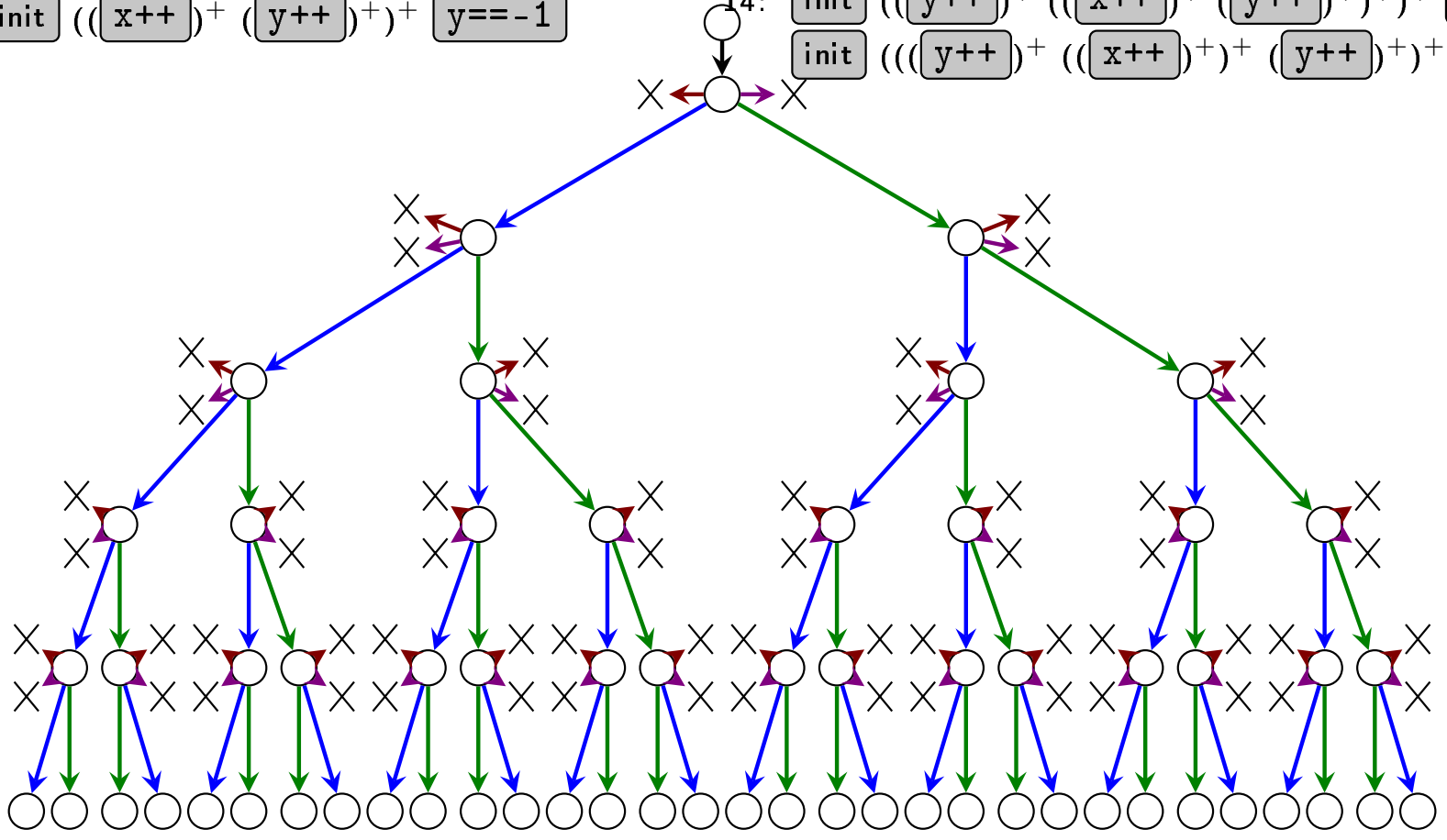
init (((x++)$^+$ ((y++)$^+$)$^+$ (x++)$^+$)$^+$ y==-1

13: init ((y++)$^+$ ((x++)$^+$ (y++)$^+$)$^+$)$^+$ x==-1

init (((y++)$^+$ ((x++)$^+$)$^+$ (y++)$^+$)$^+$ x==-1

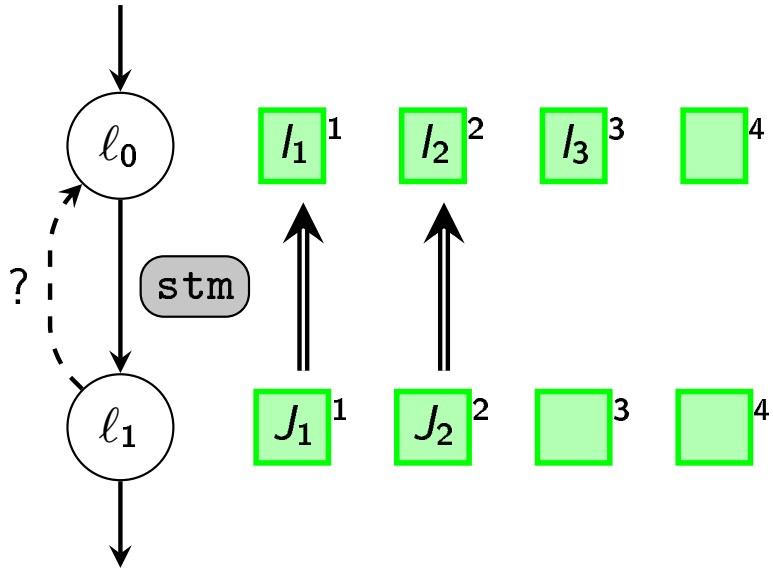
14: init ((y++)$^+$ ((x++)$^+$ (y++)$^+$)$^+$)$^+$ y==-1

init (((y++)$^+$ ((x++)$^+$)$^+$ (y++)$^+$)$^+$ y==-1



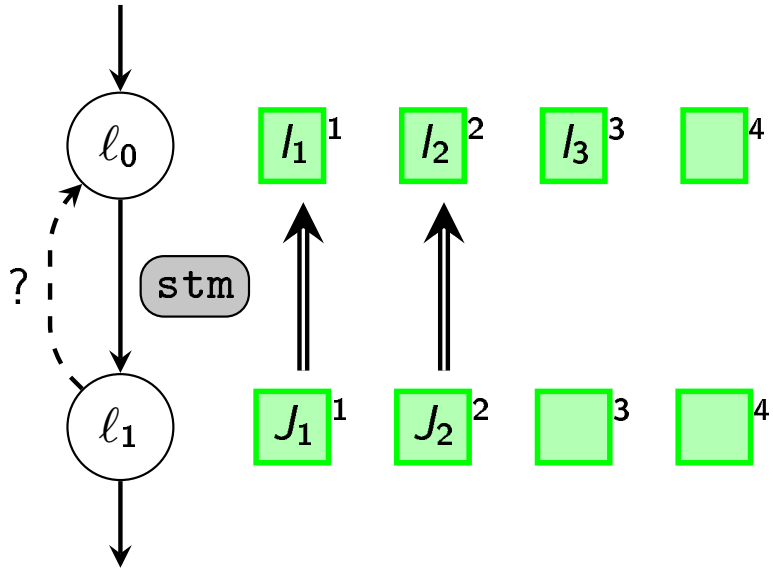
IMPACT-Variation 1

Idee: Implikations-Checks für Covering nur innerhalb einer Interpolantensequenz, bei nicht vorhandenen Interpolanten wird nichts überprüft

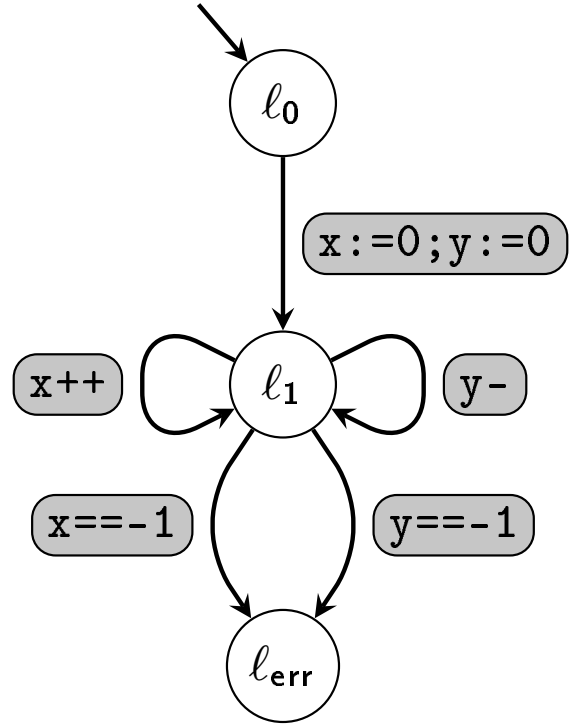


IMPACT-Variation 1

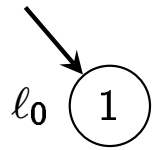
Idee: Implikations-Checks für Covering nur innerhalb einer Interpolantensequenz, bei nicht vorhandenen Interpolanten wird nichts überprüft



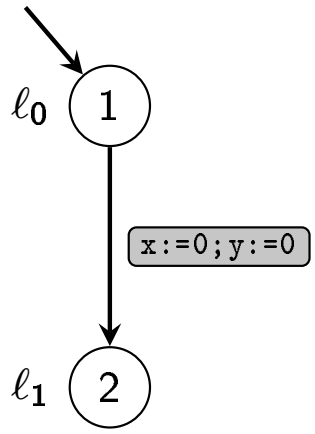
Beispiel für Inkorrektheit:



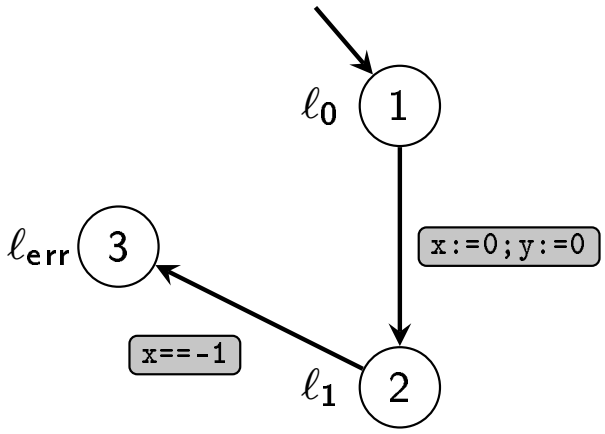
IMPACT-Variation 1



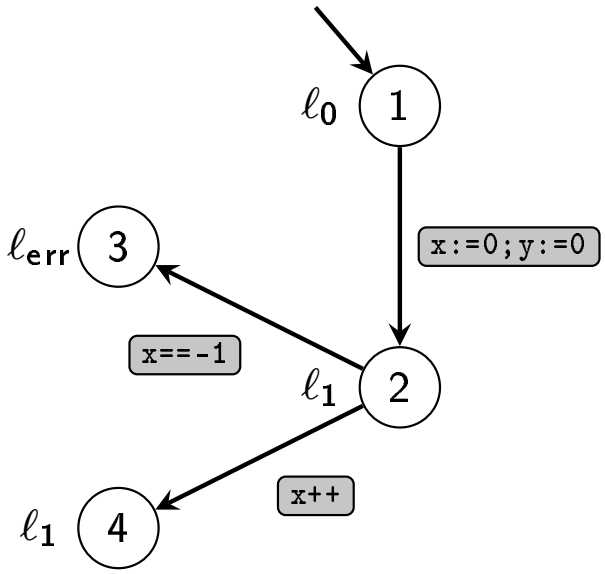
IMPACT-Variation 1



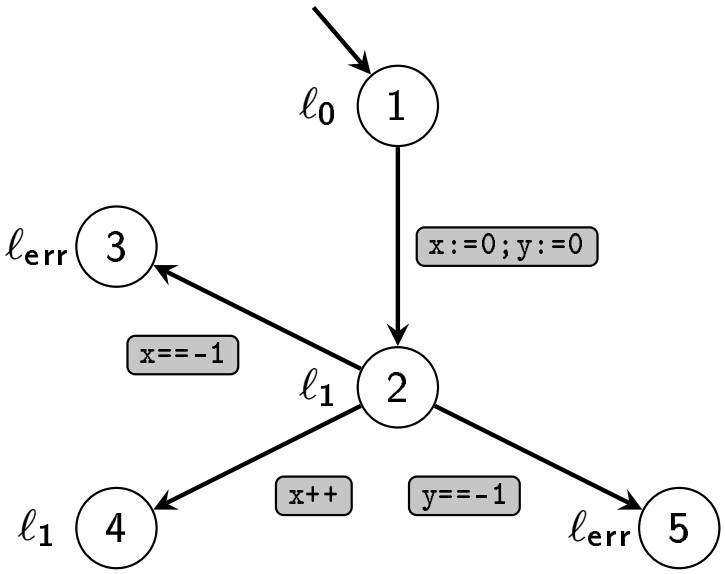
IMPACT-Variation 1



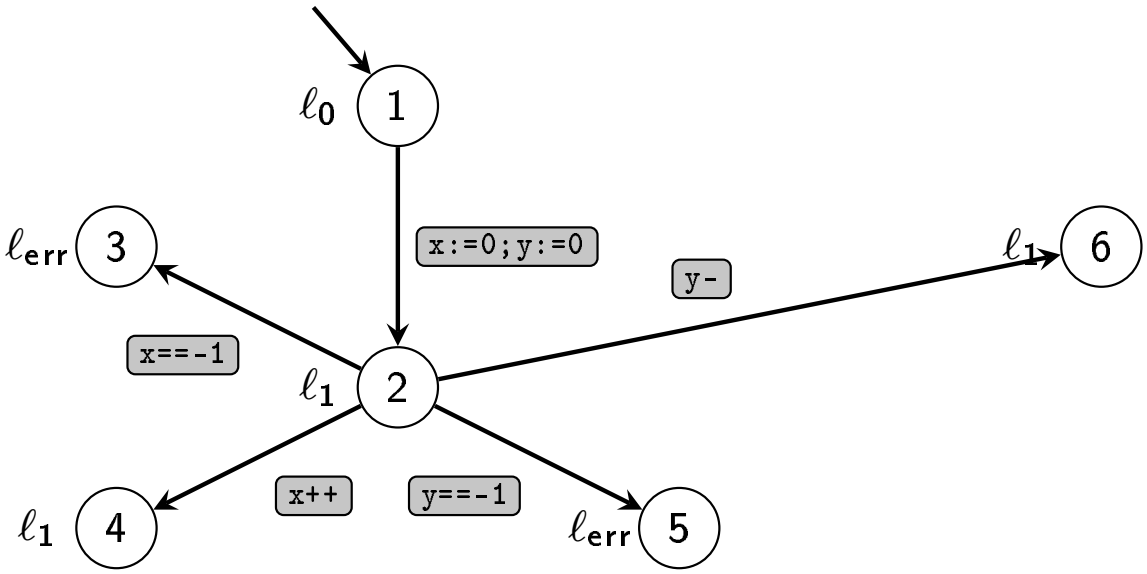
IMPACT-Variation 1



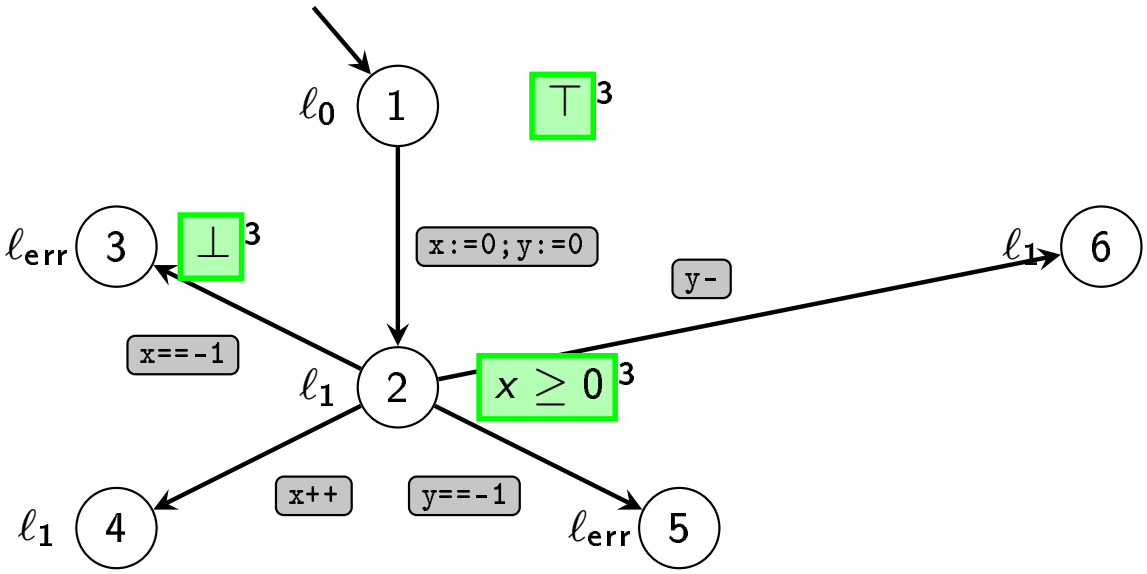
IMPACT-Variation 1



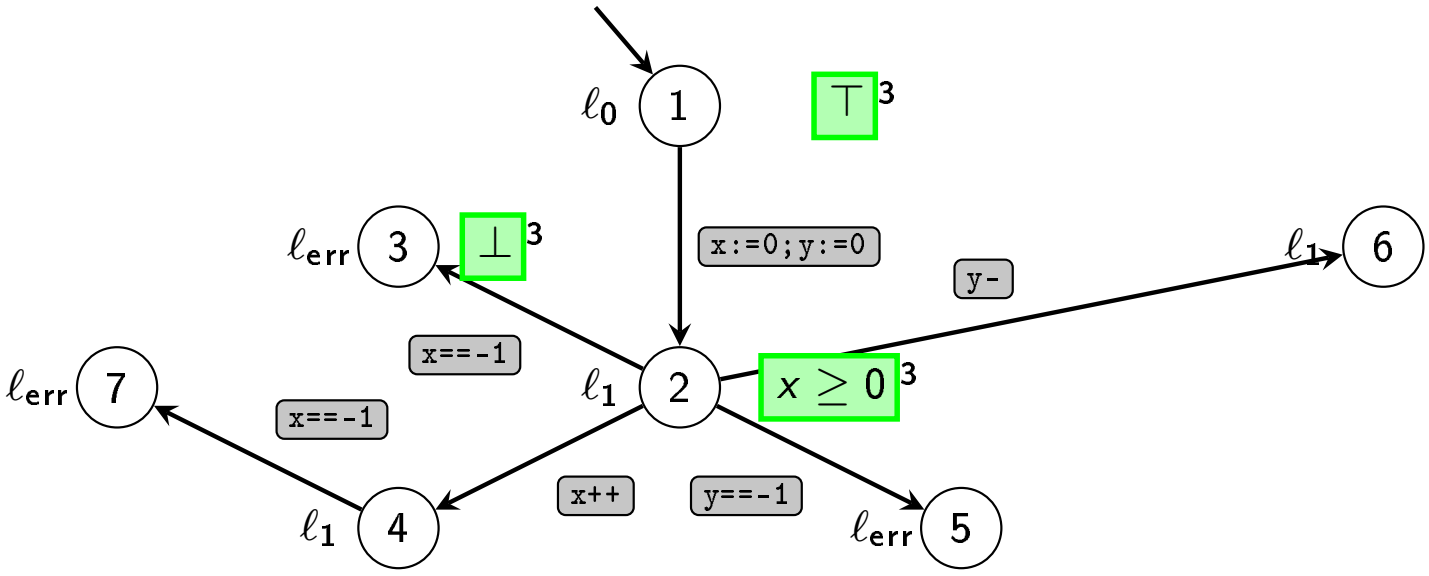
IMPACT-Variation 1



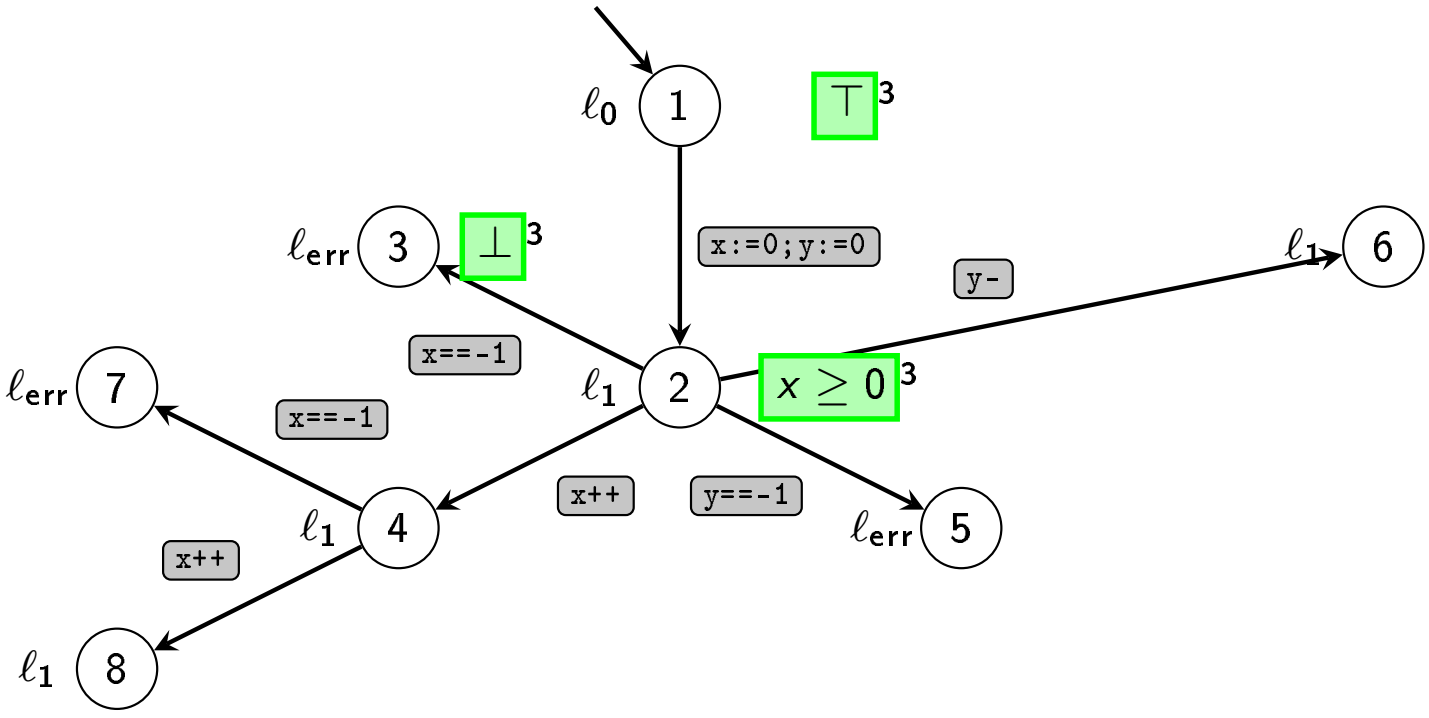
IMPACT-Variation 1



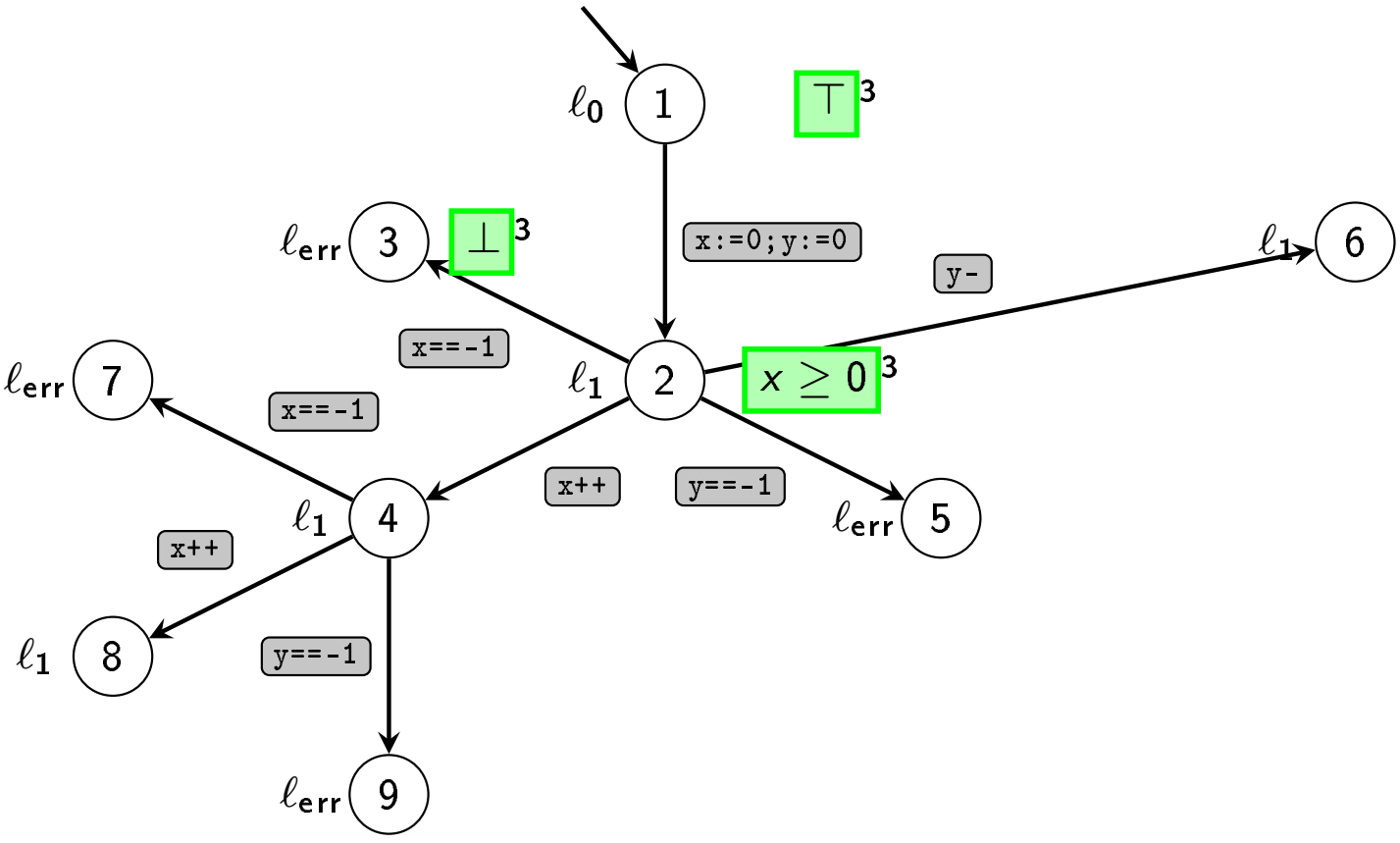
IMPACT-Variation 1



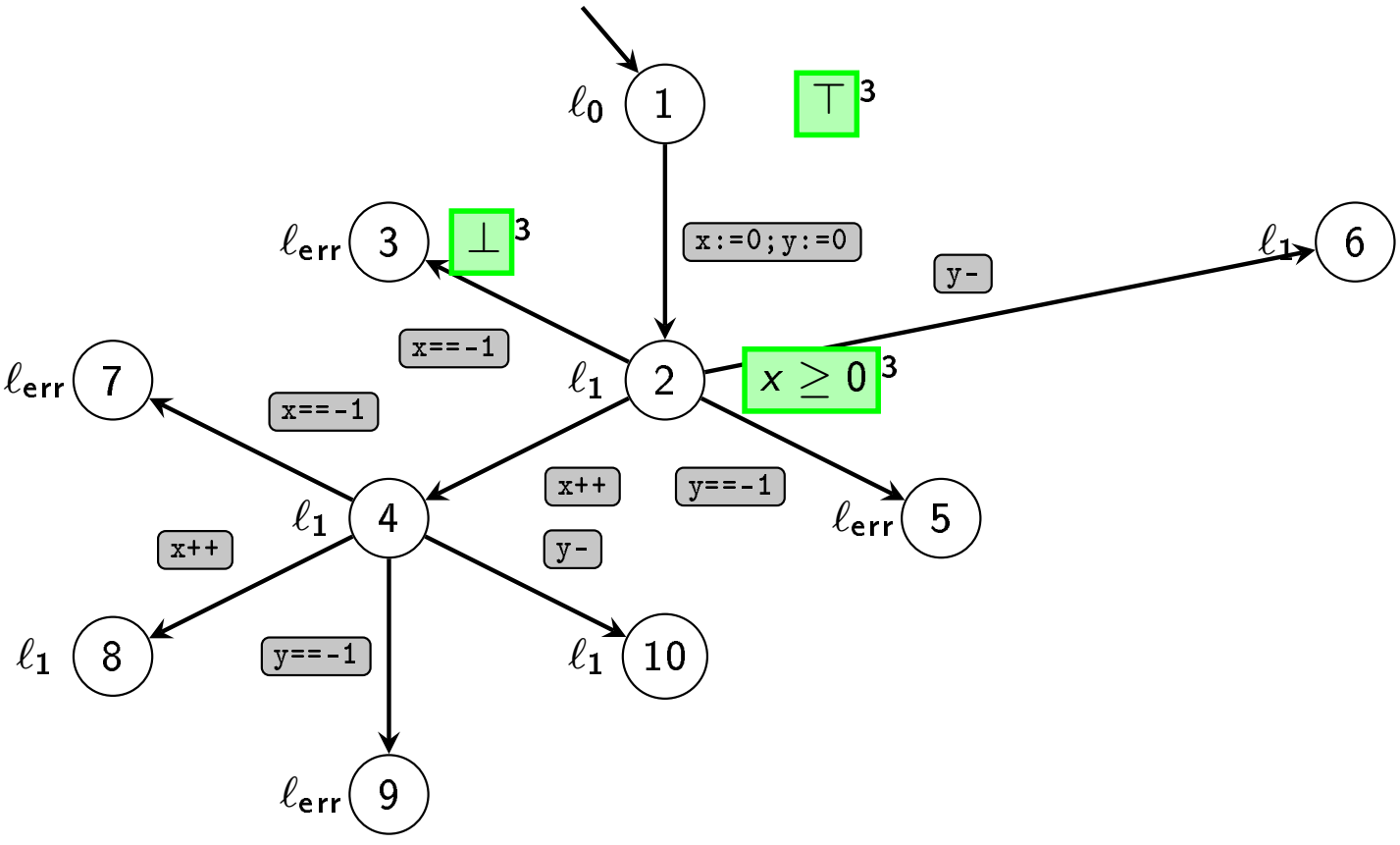
IMPACT-Variation 1



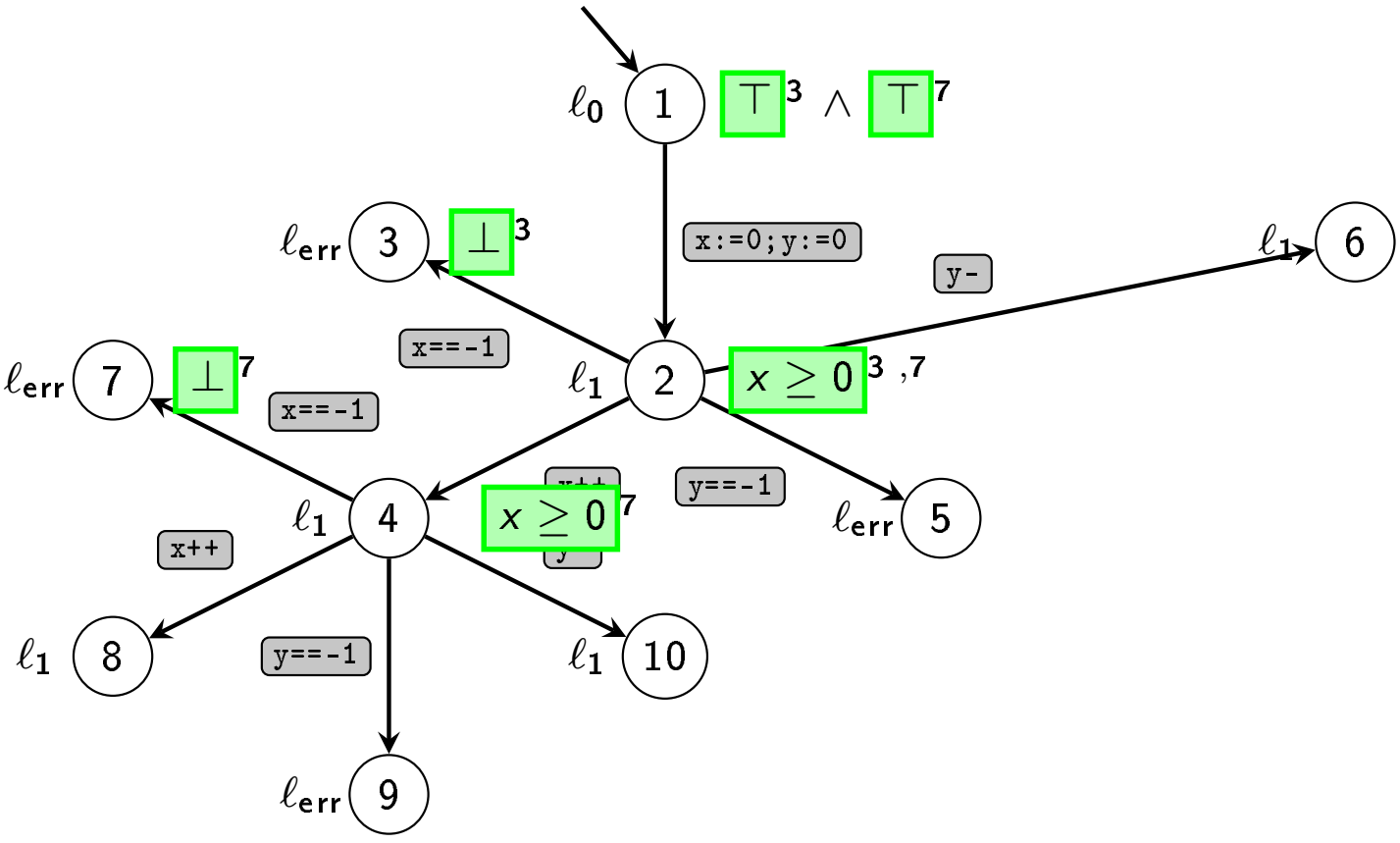
IMPACT-Variation 1



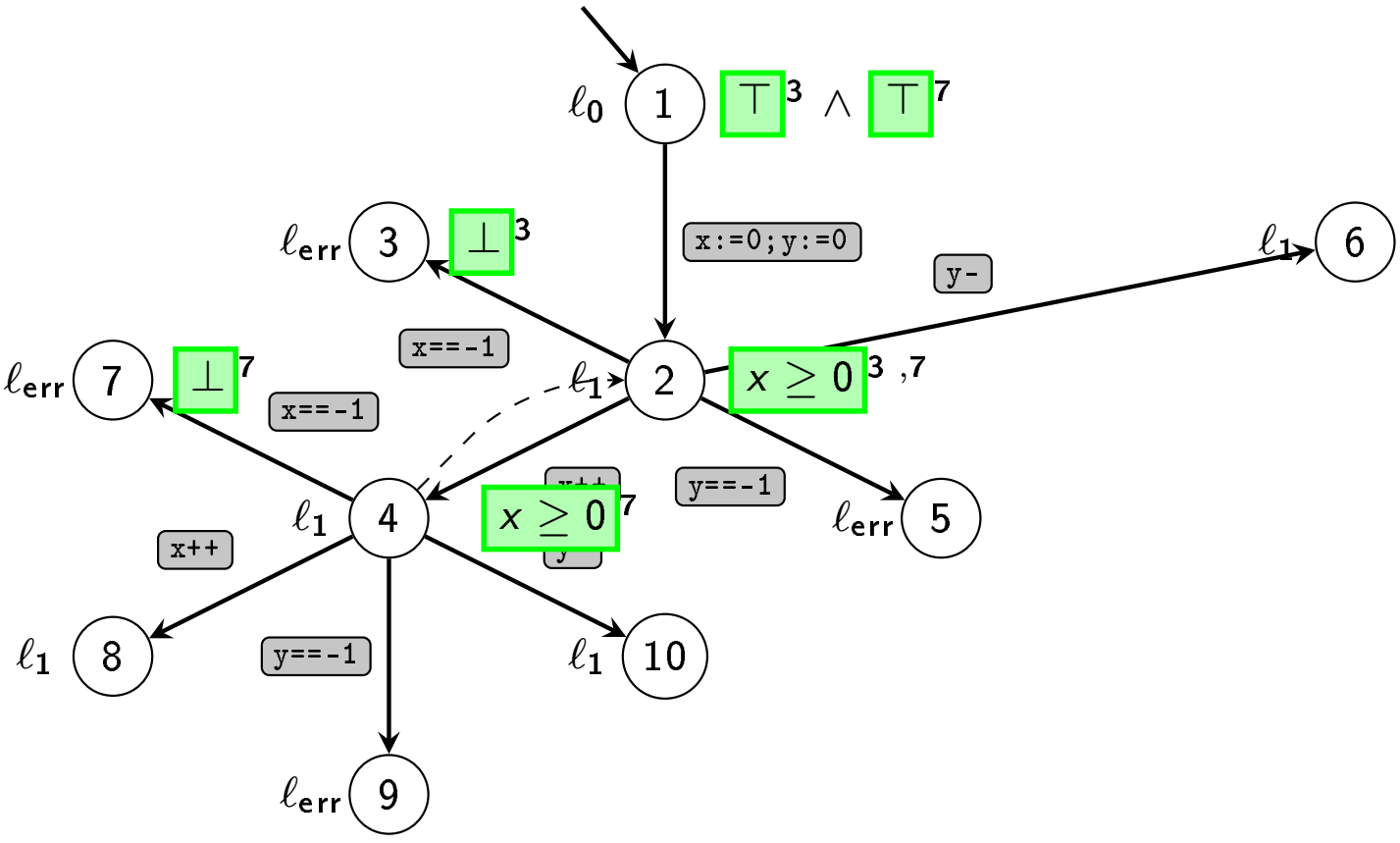
IMPACT-Variation 1



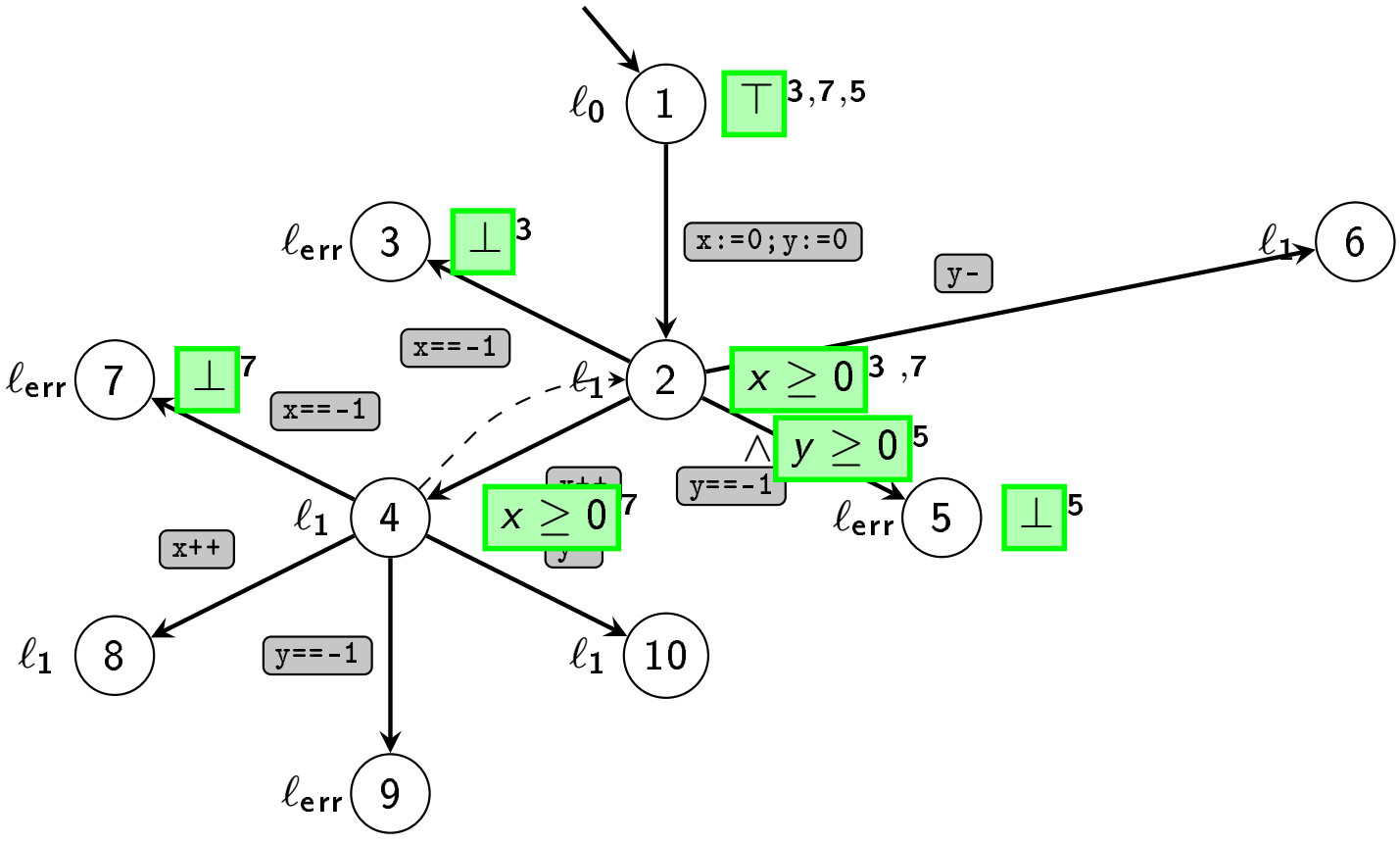
IMPACT-Variation 1



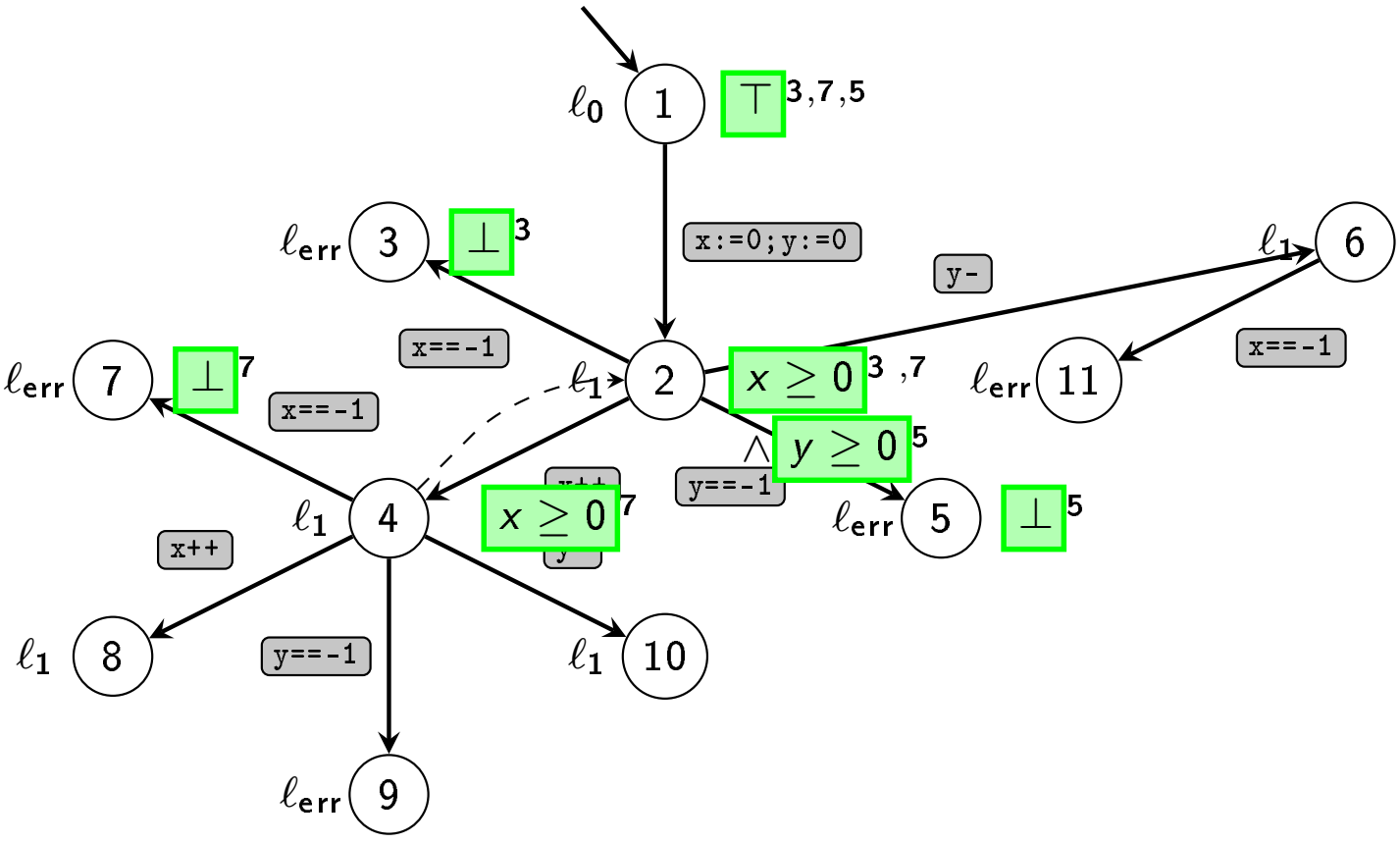
IMPACT-Variation 1



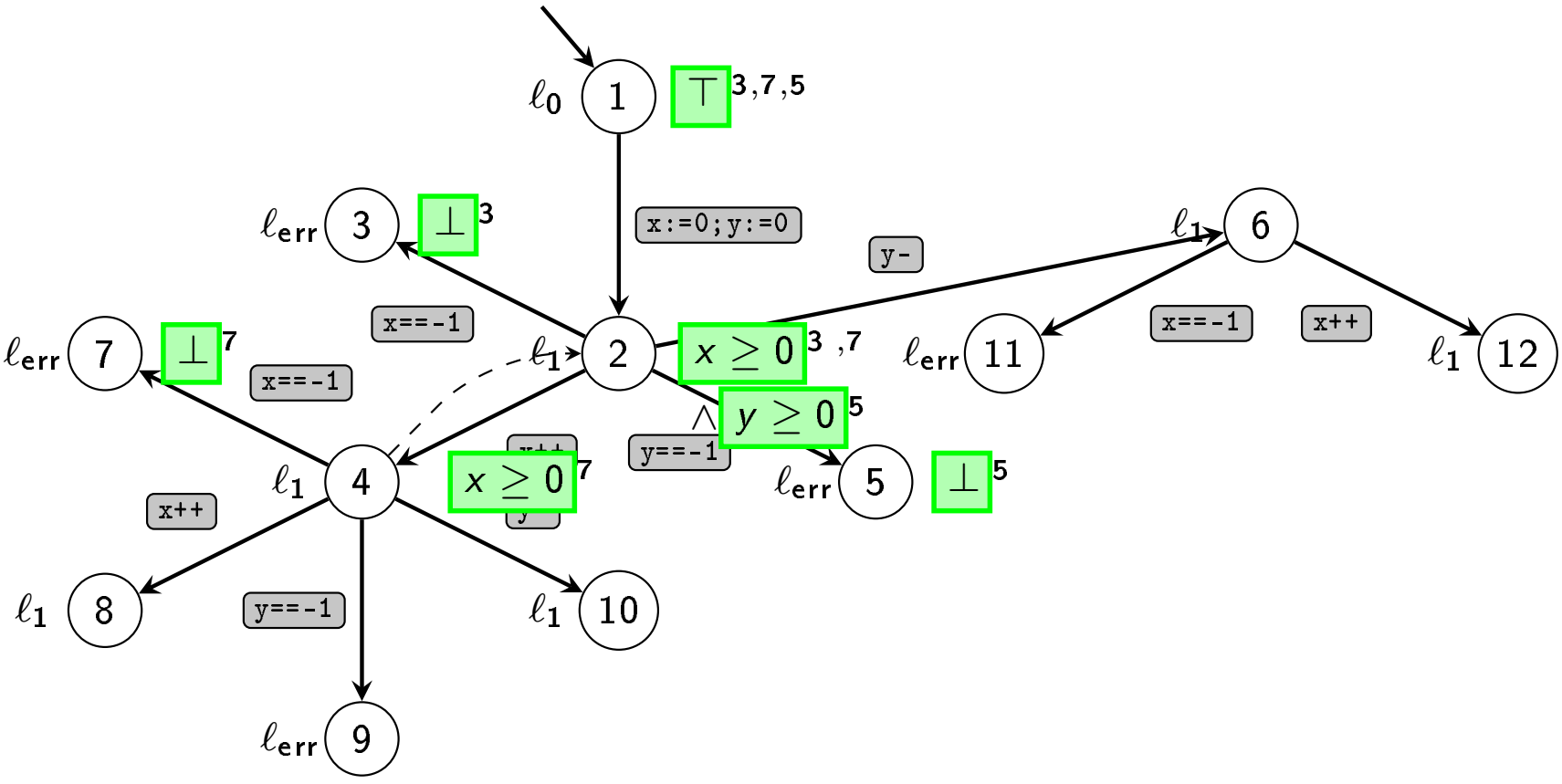
IMPACT-Variation 1



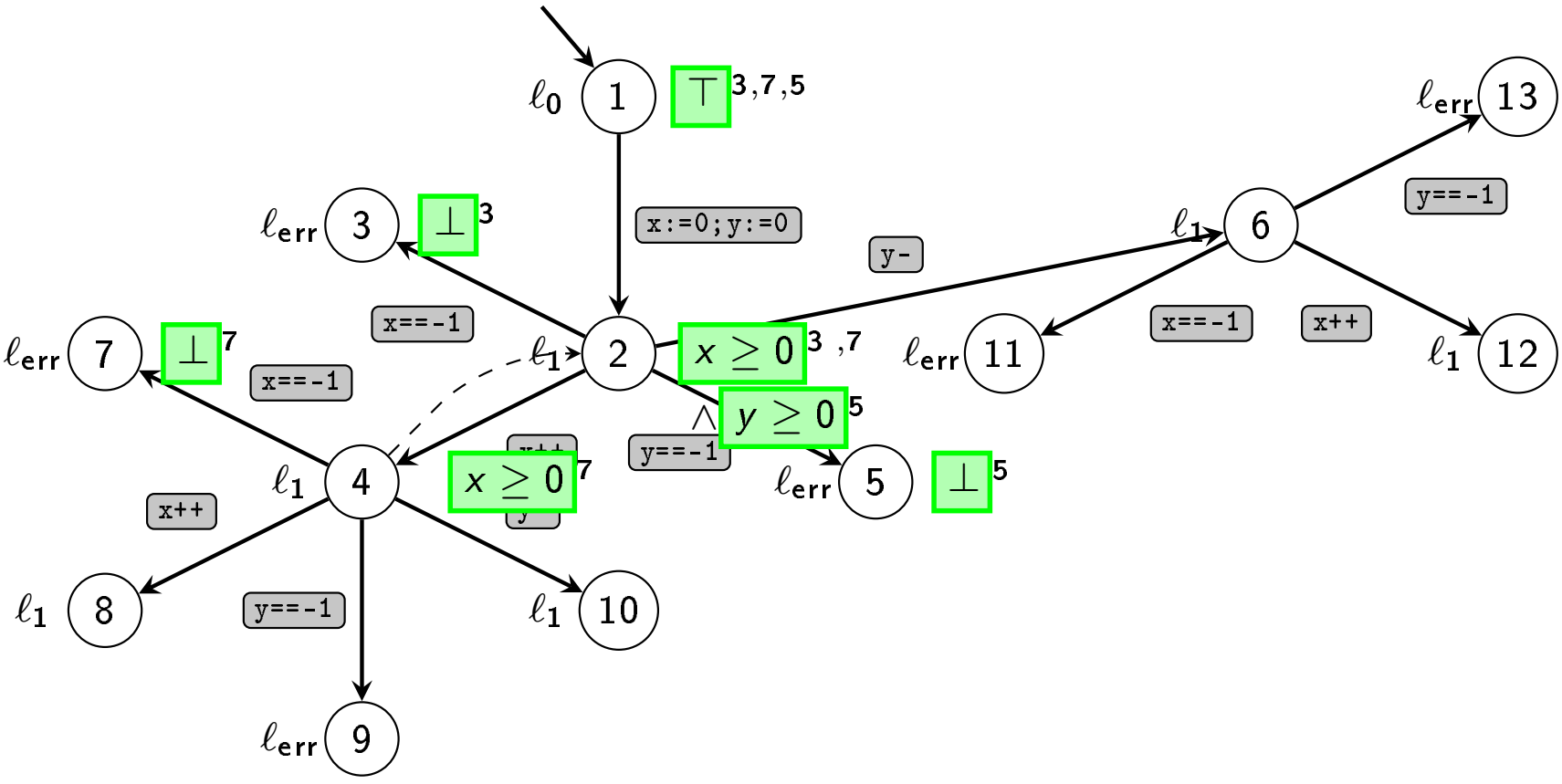
IMPACT-Variation 1



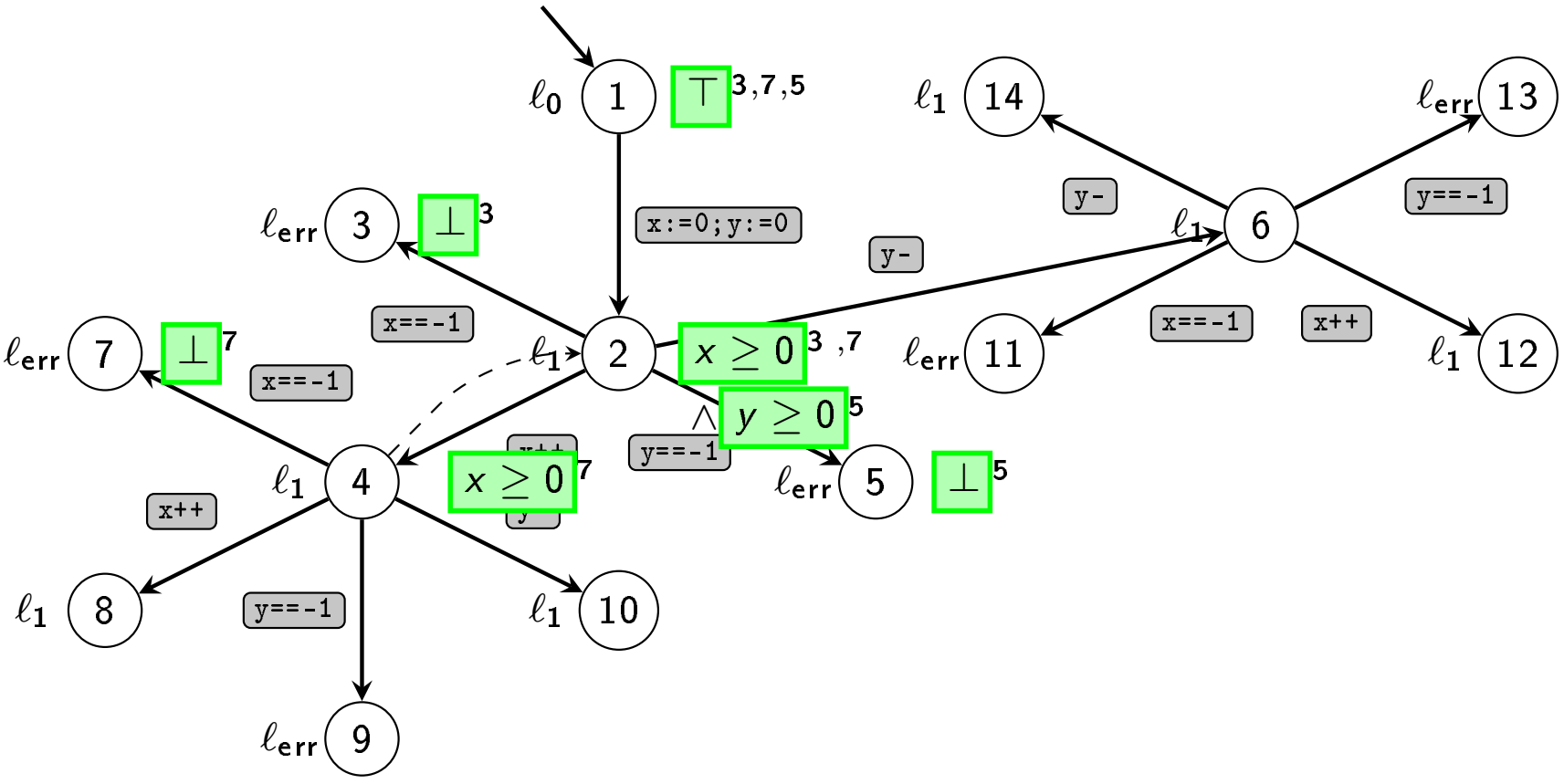
IMPACT-Variation 1



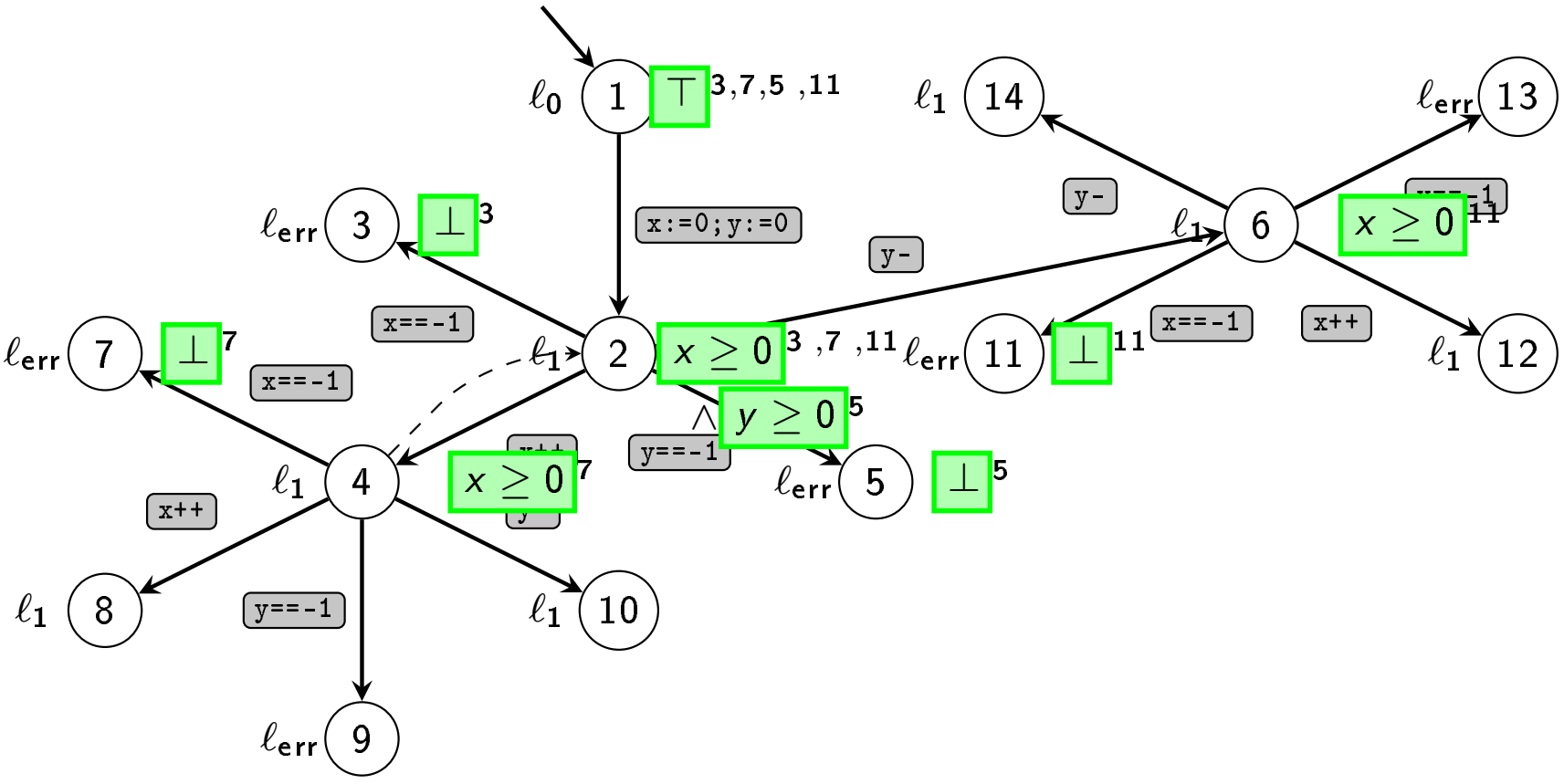
IMPACT-Variation 1



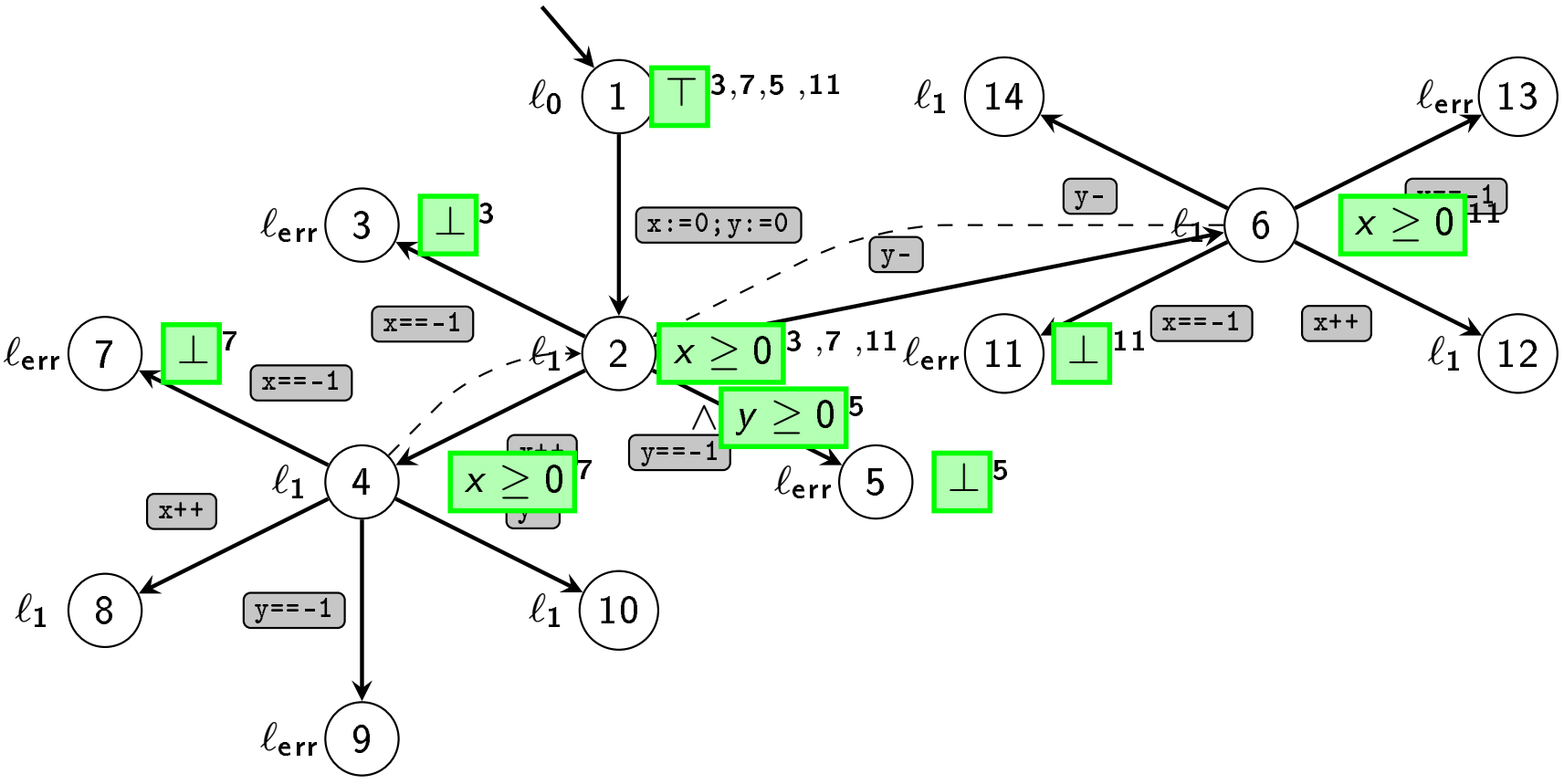
IMPACT-Variation 1



IMPACT-Variation 1

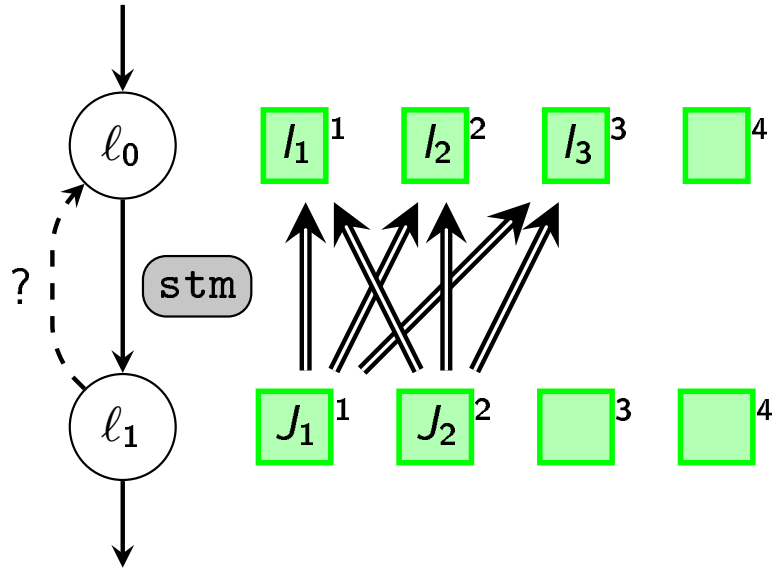


IMPACT-Variation 1



IMPACT-Variation 2

Idee: teste alle Implikationen zwischen Elementen beider Tupel



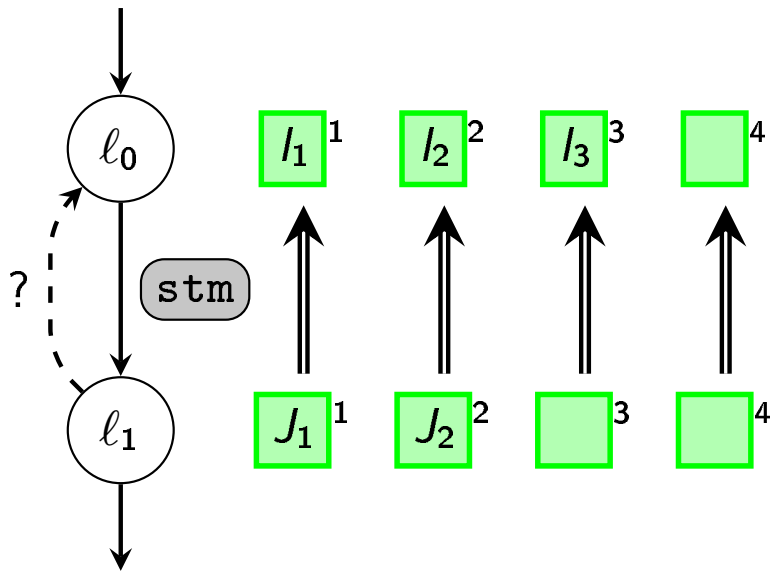
Fazit:

- Ist korrekt, da nur gecovert wird, wenn IMPACT auch covern würde
- Im Allgemeinen wird seltener gecovert, als bei IMPACT
- Keine große Ähnlichkeit zu Trace Abstraction
- Viele redundante Überprüfungen von Implikationen

IMPACT-Variation 3

Idee:

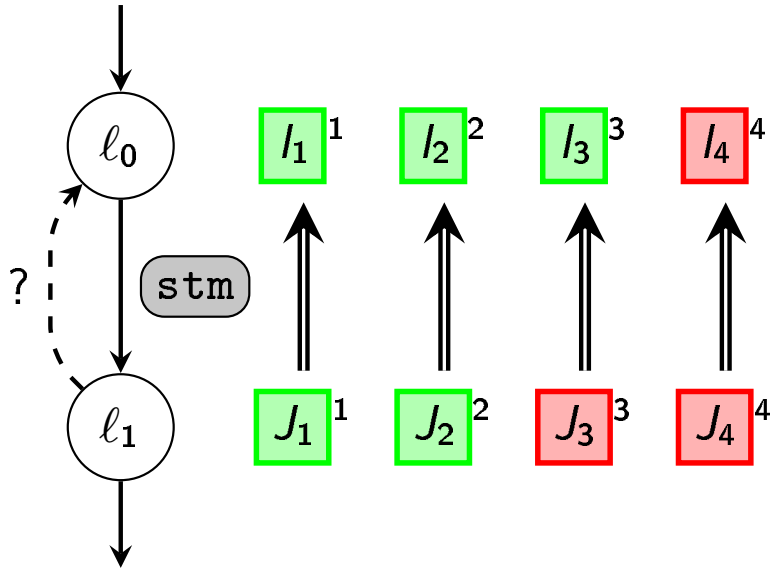
- Implikationsüberprüfung nur innerhalb der Interpolantensequenzen
- Ergänze fehlende Formeln mittels der Interpolantenautomaten aus der Trace Abstraction

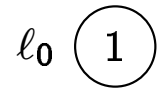


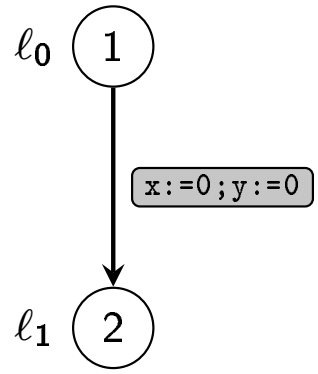
IMPACT-Variation 3

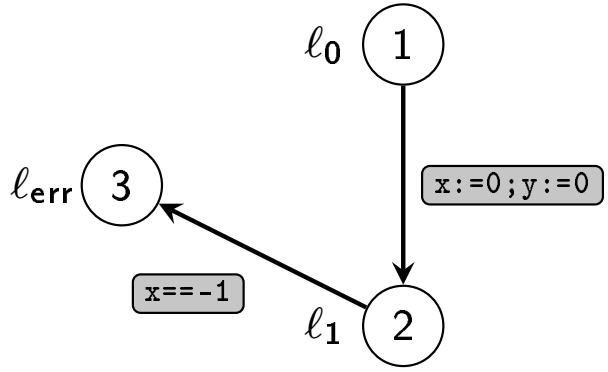
Idee:

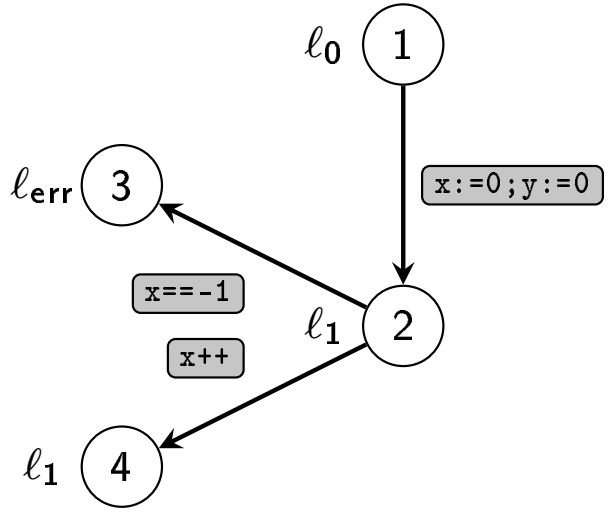
- Implikationsüberprüfung nur innerhalb der Interpolantensequenzen
- Ergänze fehlende Formeln mittels der Interpolantenautomaten aus der Trace Abstraction

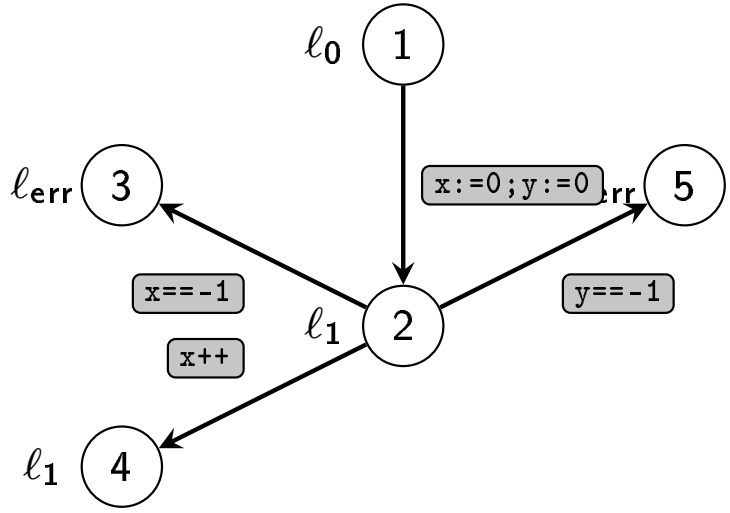


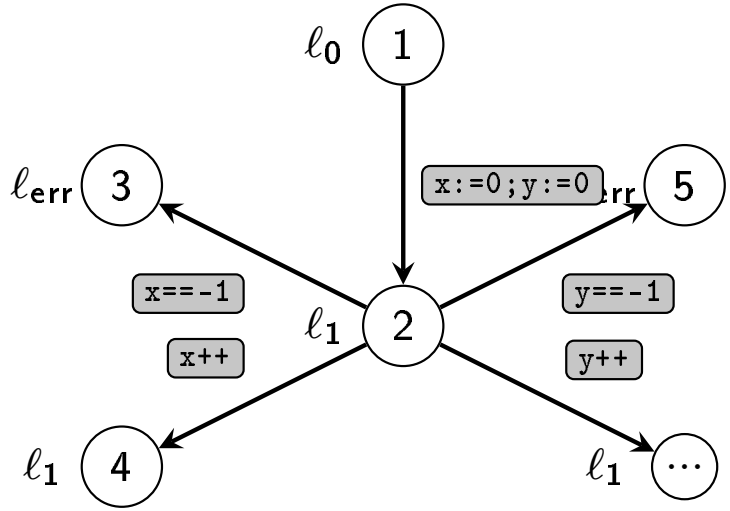

ℓ_0
1

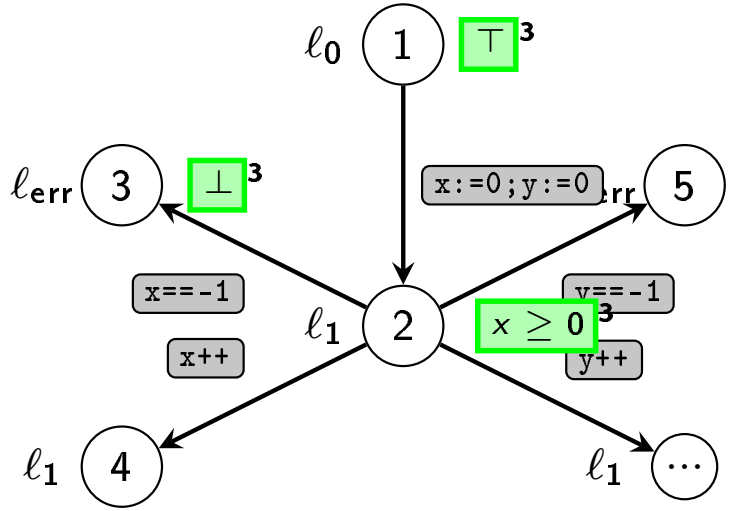

ℓ_0
1
x:=0;y:=0
ℓ_1
2

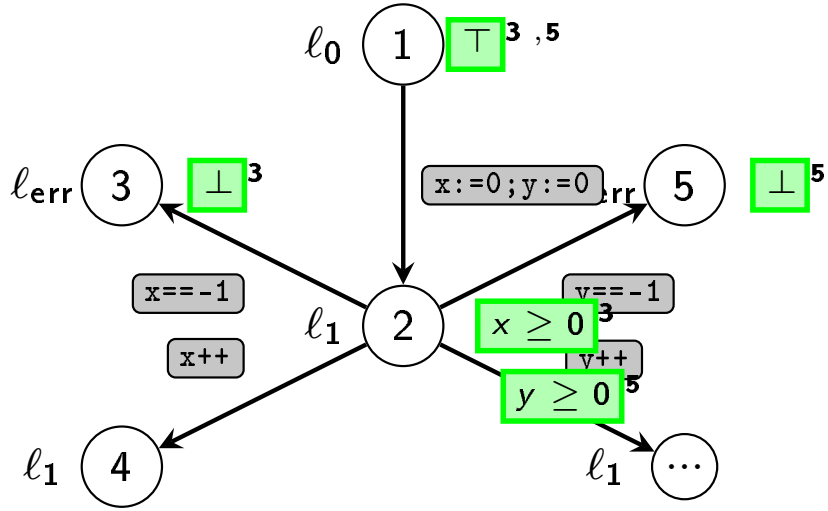

ℓ_0
1
x:=0;y:=0
ℓ_{err}
3
x==-1
ℓ_1
2

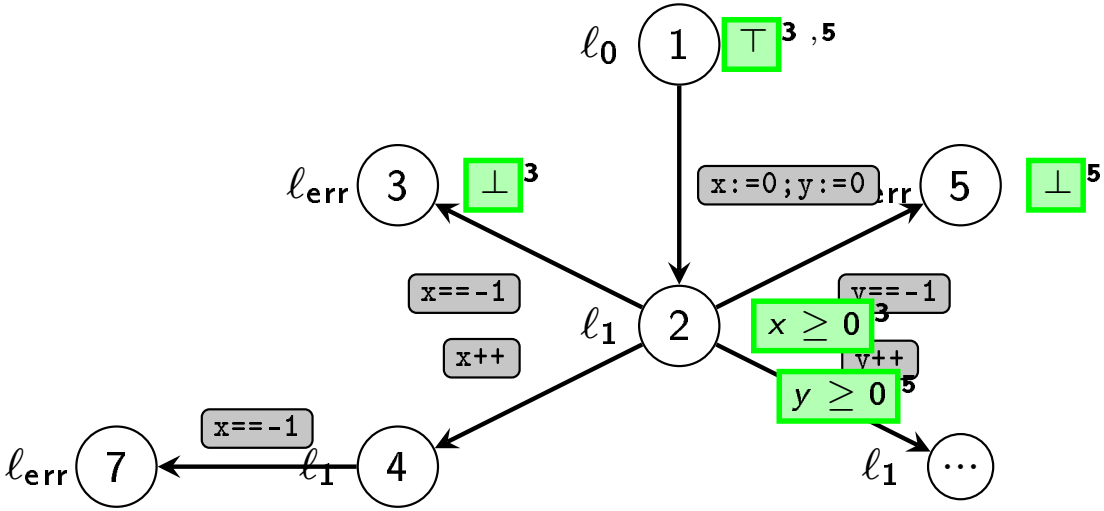

ℓ_0
1
x:=0;y:=0
ℓ_{err}
3
x==-1
ℓ_1
2
x++
ℓ_1
4

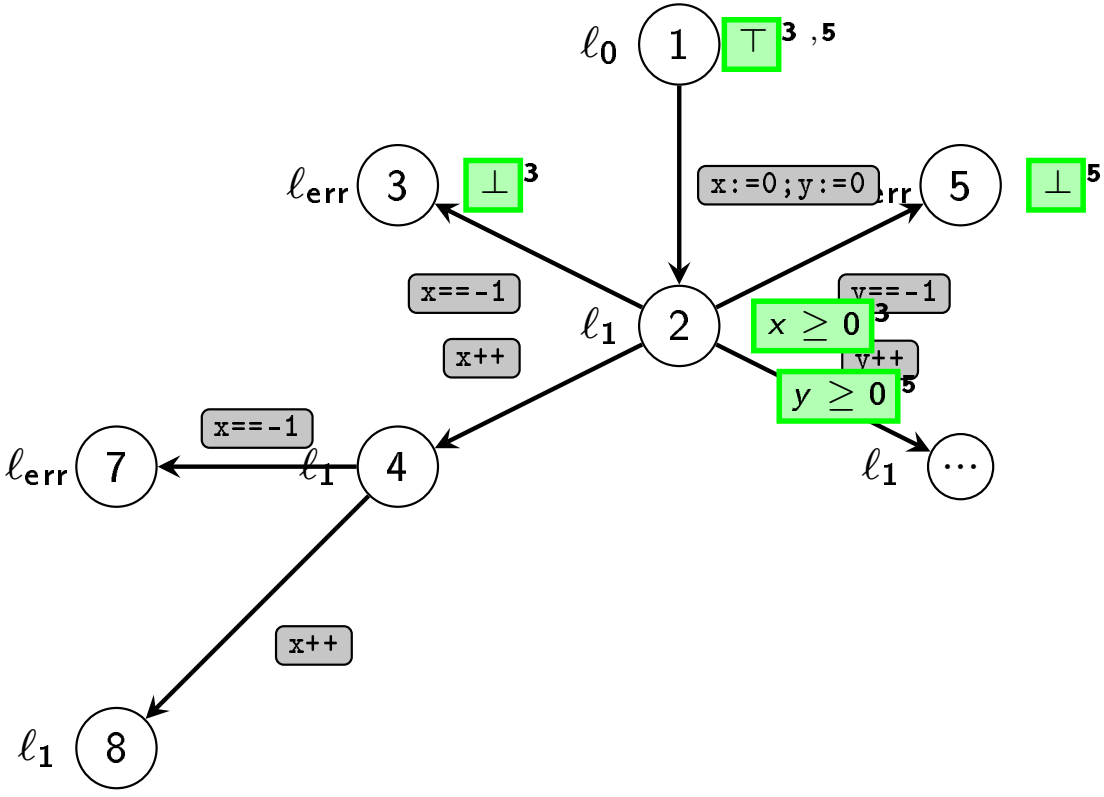

ℓ_0
1
ℓ_{err}
3
x:=0;y:=0
5
x==-1
y==-1
ℓ_1
2
x++
ℓ_1
4

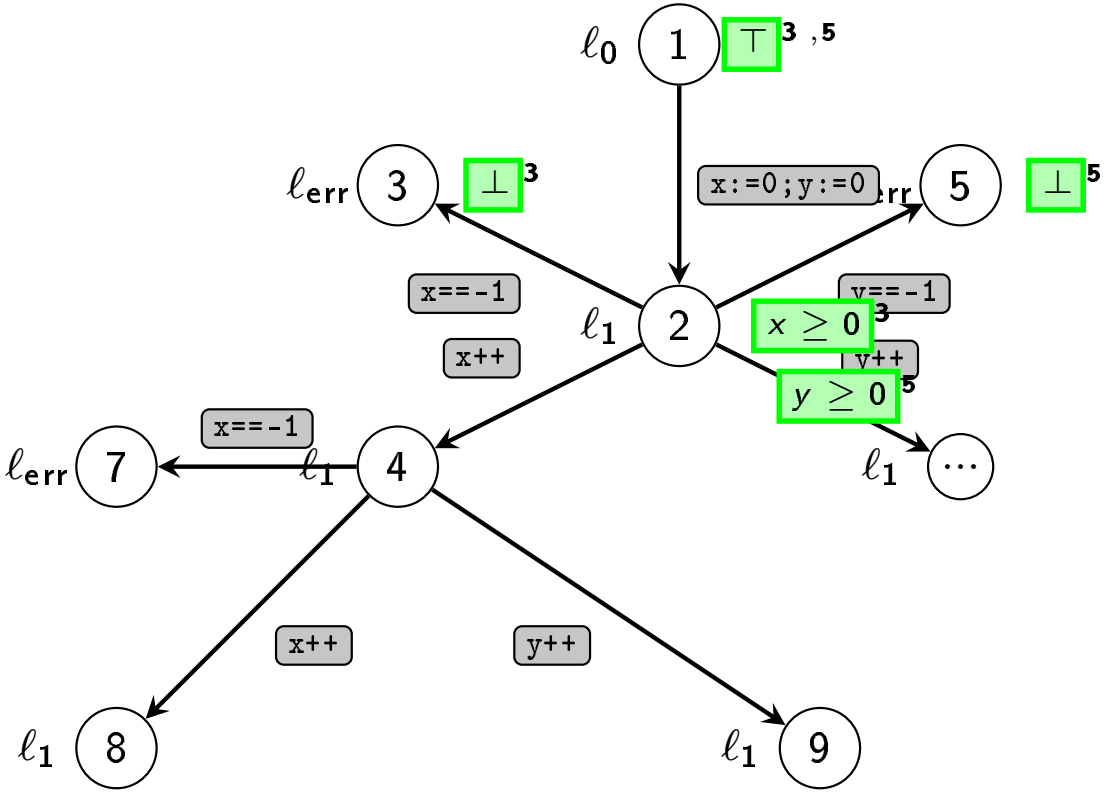

ℓ_0
1
x:=0;y:=0
ℓ_{err}
3
5
x==-1
y==-1
ℓ_1
2
x++
y++
ℓ_1
4
ℓ_1
...

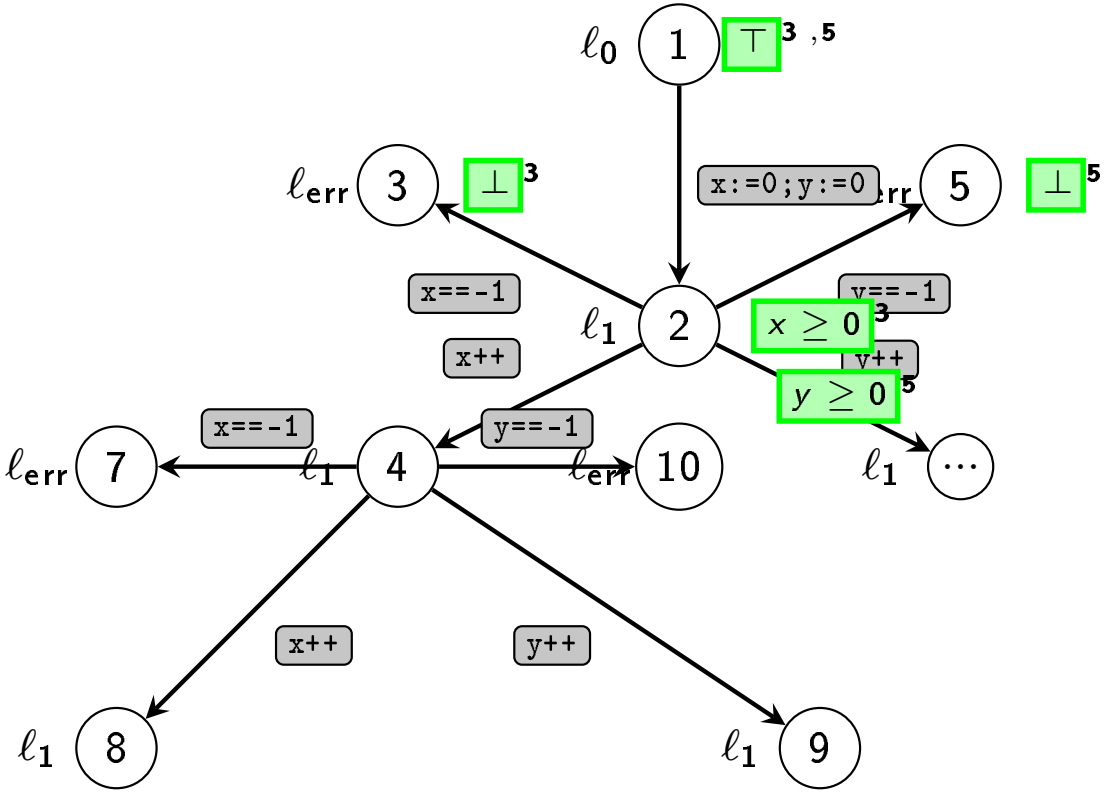

ℓ_0
1
$\top$
3
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
x==-1
x++
ℓ_1
2
$x \geq 0$
3
y==-1
y++
ℓ_1
4
ℓ_1
...

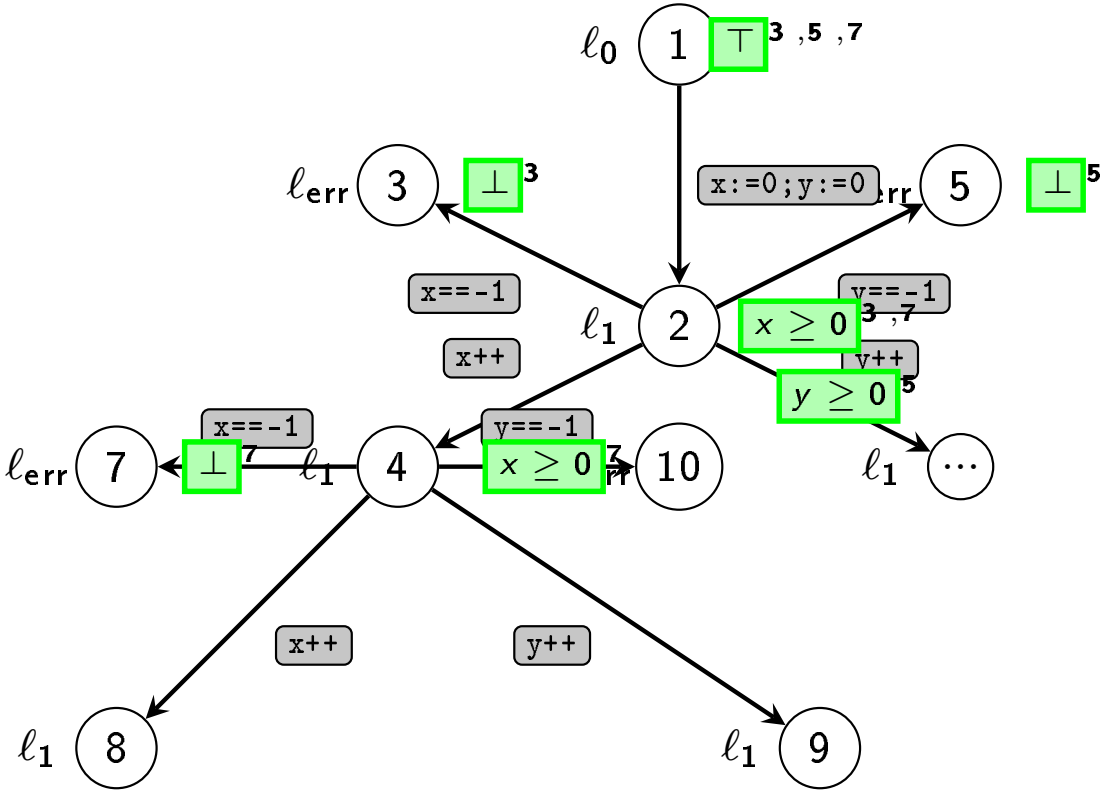

ℓ_0
1
$\top$ 3 ,5
ℓ_{err}
3
$\bot$ 3
x:=0;y:=0
5
$\bot$ 5
x==-1
x++
ℓ_1
2
y==-1
$x \geq 0$ 3
y++
$y \geq 0$ 5
ℓ_1
4
ℓ_1
...

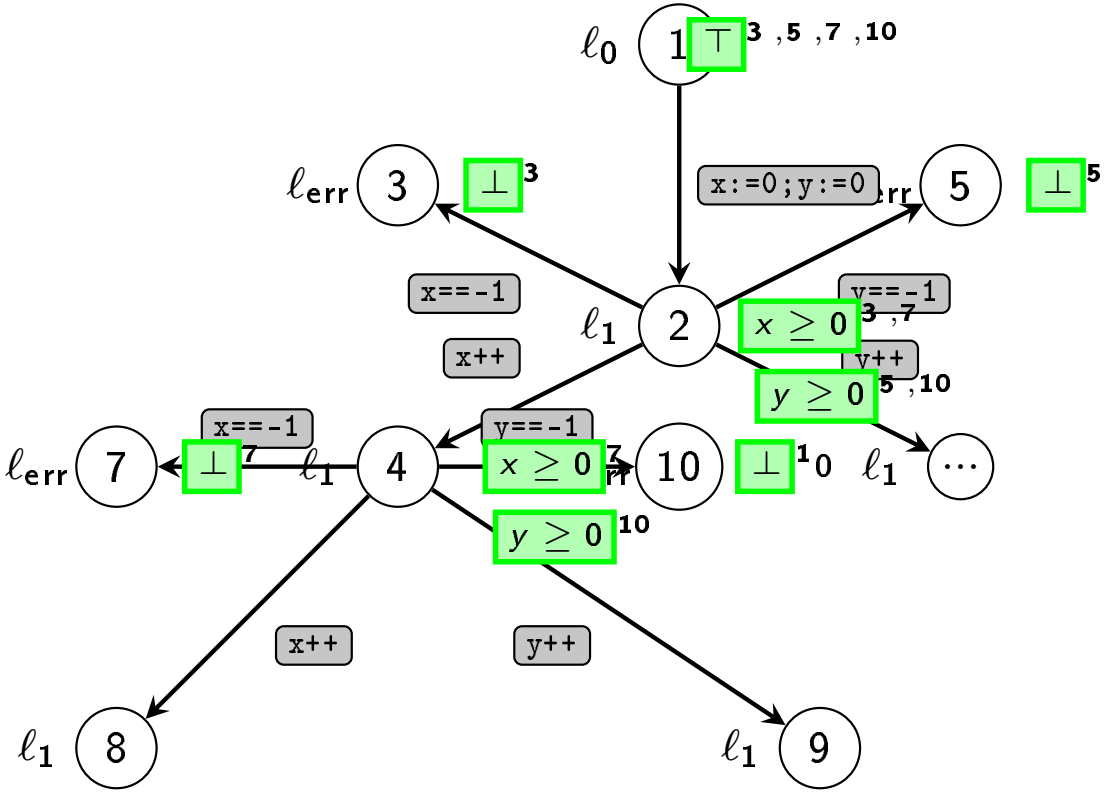

ℓ_0
1
$\top$ 3 ,5
ℓ_{err}
3
$\bot$ 3
x:=0;y:=0
5
$\bot$ 5
x==-1
ℓ_1
2
y==-1
$x \geq 0$ 3
x++
y++
$y \geq 0$ 5
x==-1
ℓ_{err}
7
ℓ_1
4
ℓ_1
...

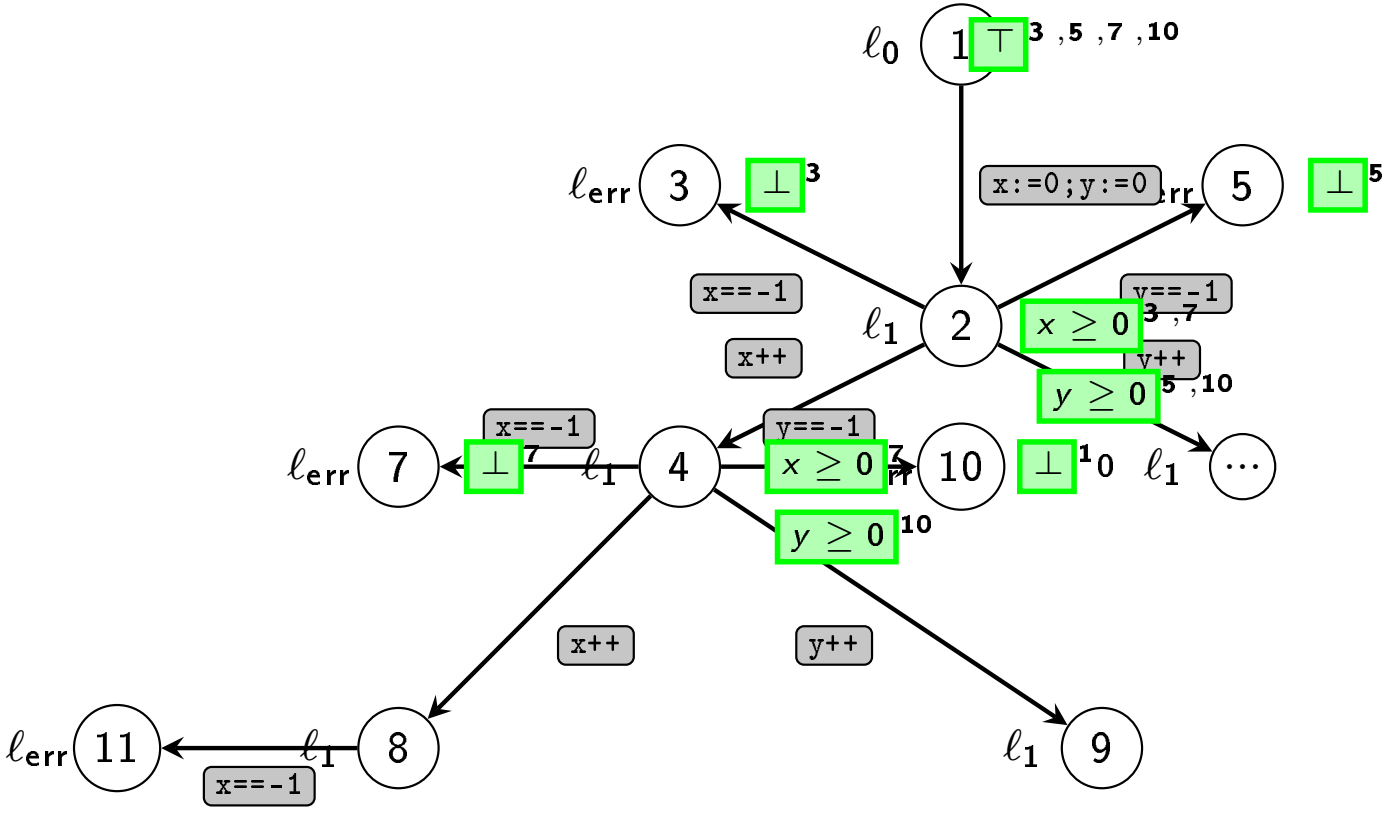

ℓ_0
1
$\top^{3,5}$
ℓ_{err}
3
$\bot^3$
x:=0;y:=0
5
$\bot^5$
x==-1
x++
ℓ_1
2
y==-1
$x \geq 0^3$
y++
$y \geq 0^5$
x==-1
ℓ_{err}
7
ℓ_1
4
ℓ_1
…
x++
ℓ_1
8

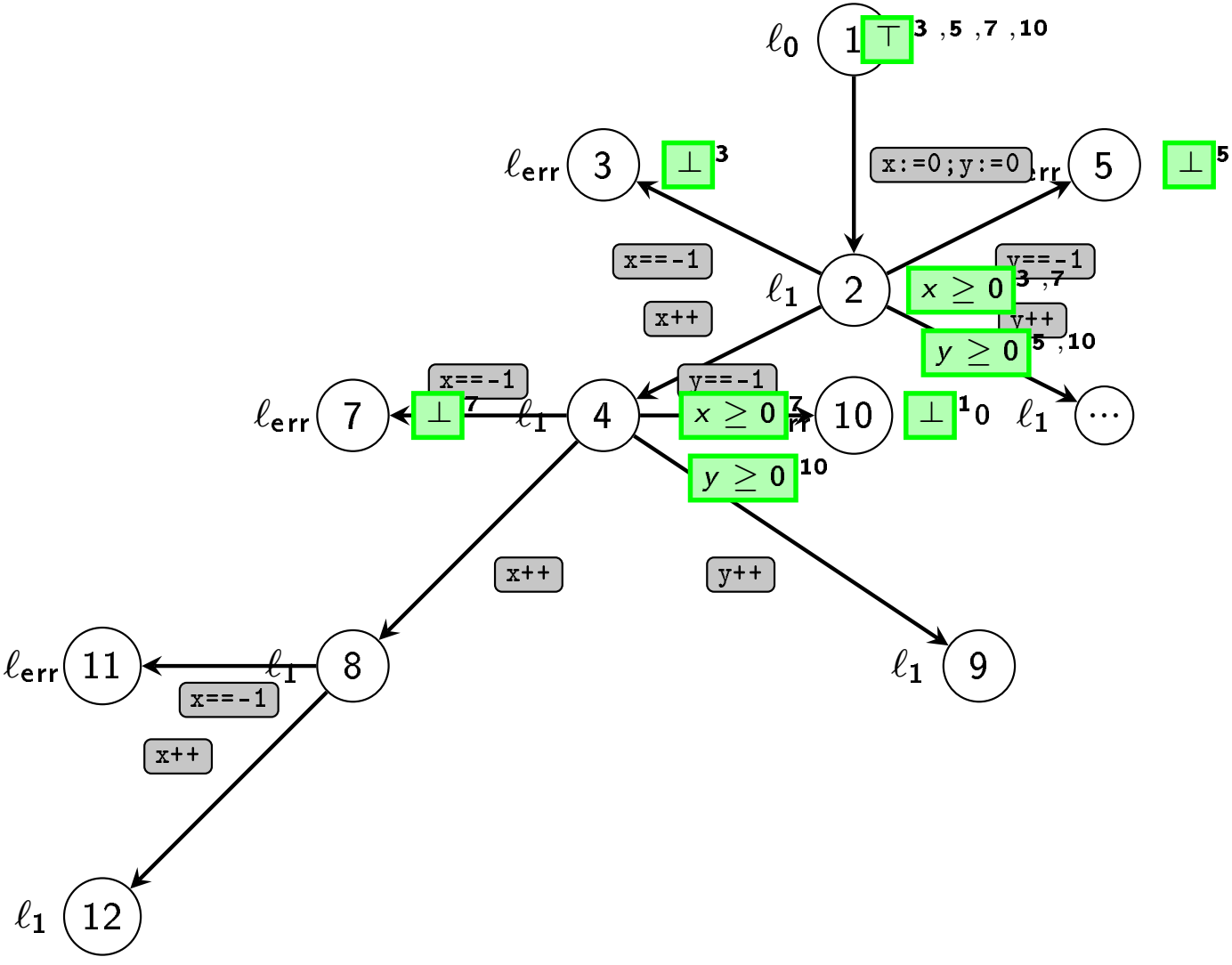

ℓ_0
1
$\top$ 3 ,5
ℓ_{err}
3
$\bot$ 3
x:=0;y:=0
5
$\bot$ 5
x==-1
ℓ_1
2
y==-1
$x \geq 0$ 3
x++
y++
$y \geq 0$ 5
x==-1
ℓ_{err}
7
ℓ_1
4
ℓ_1
…
x++
y++
ℓ_1
8
ℓ_1
9

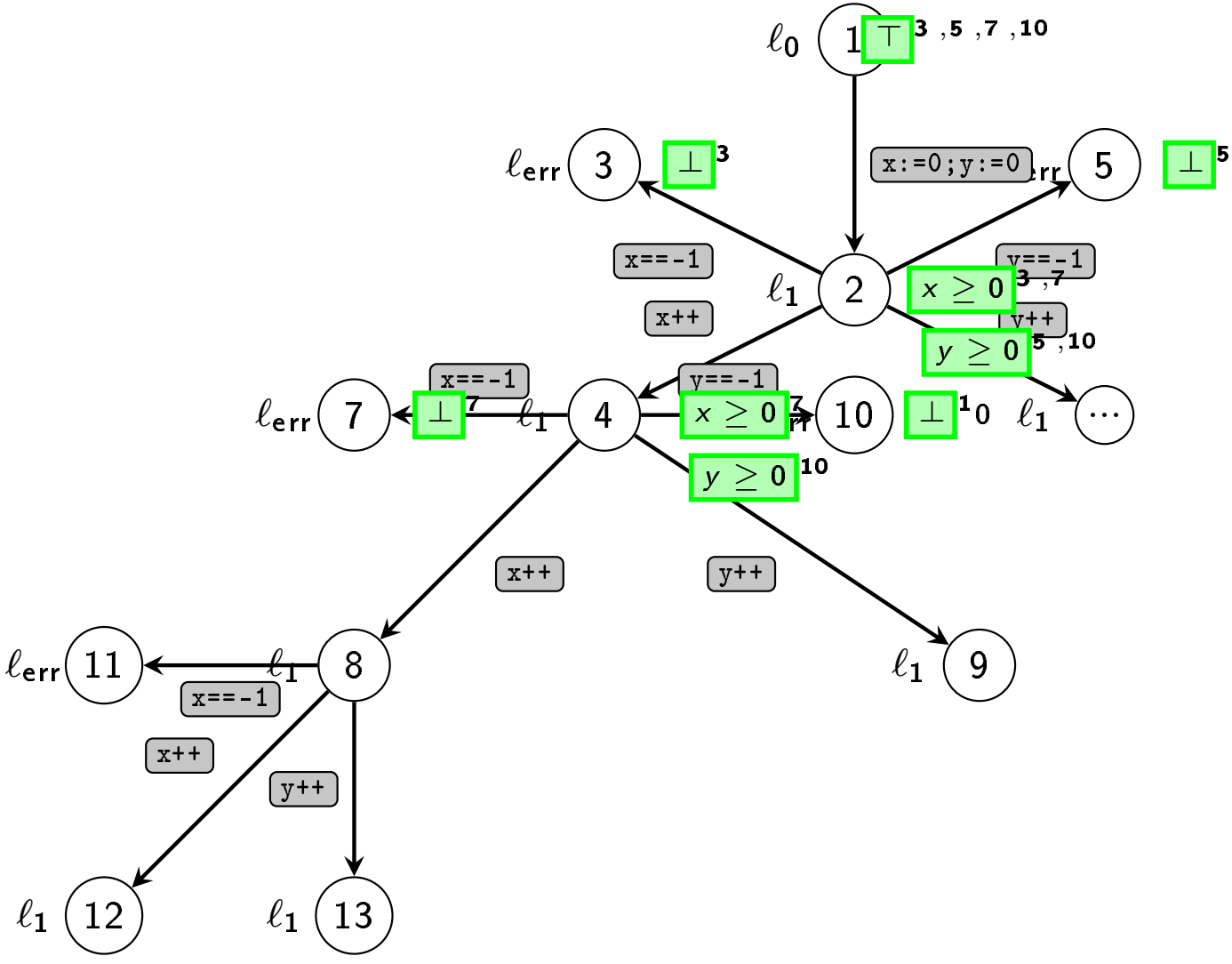

ℓ_0
1
$\top$ 3 ,5
ℓ_{err}
3
$\bot$ 3
x:=0;y:=0
5
$\bot$ 5
x==-1
ℓ_1
2
y==-1
$x \geq 0$ 3
x++
y++
$y \geq 0$ 5
x==-1
y==-1
ℓ_{err}
7
ℓ_1
4
ℓ_{err}
10
ℓ_1
...
x++
y++
ℓ_1
8
ℓ_1
9

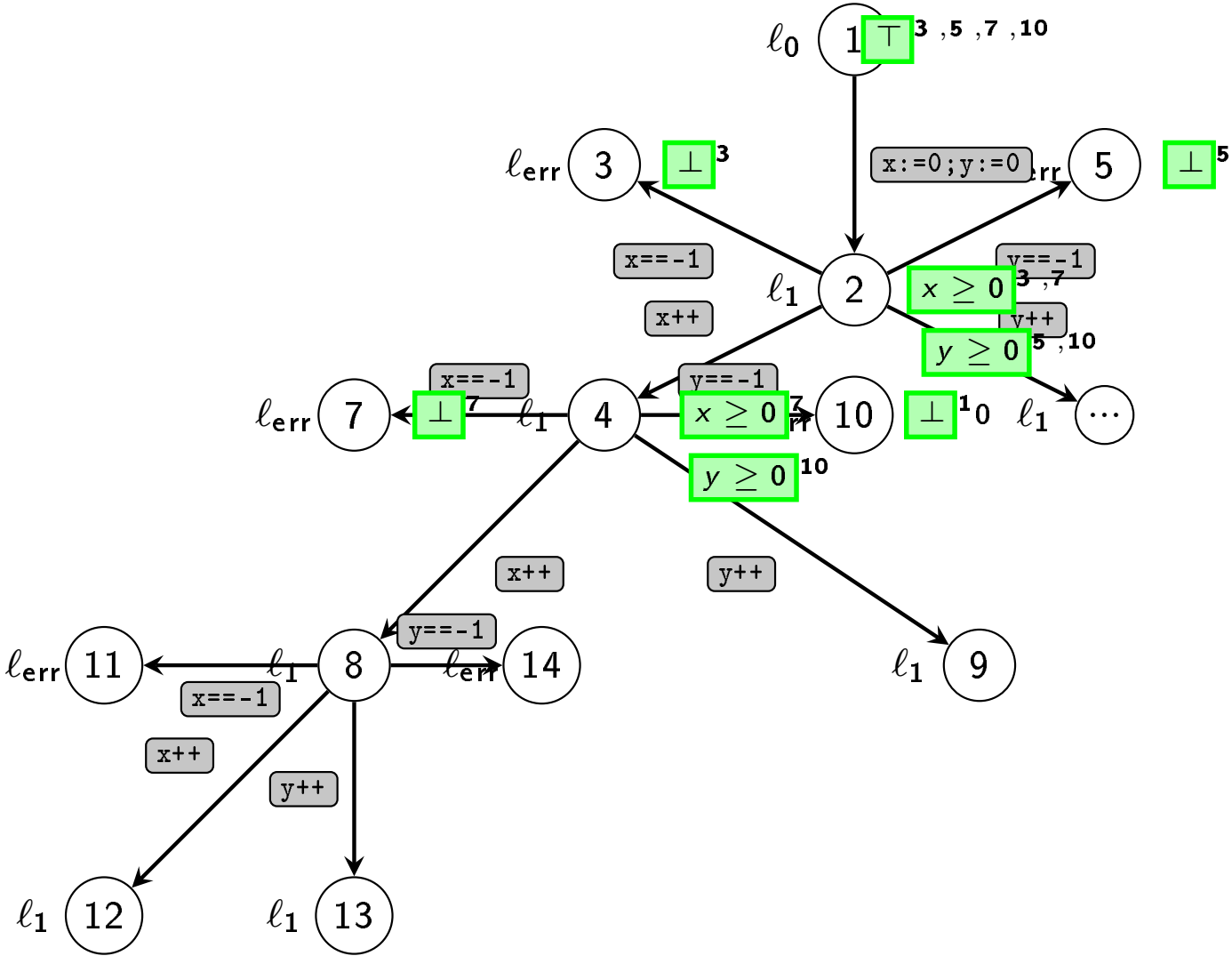

ℓ_0
1
$\top$
3 ,5 ,7
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
x==-1
ℓ_1
2
y==-1
$x \geq 0$
3 ,7
x++
y++
$y \geq 0$
5
x==-1
y==-1
ℓ_{err}
7
$\bot$
7
ℓ_1
4
$x \geq 0$
7
10
ℓ_1
…
x++
y++
ℓ_1
8
ℓ_1
9

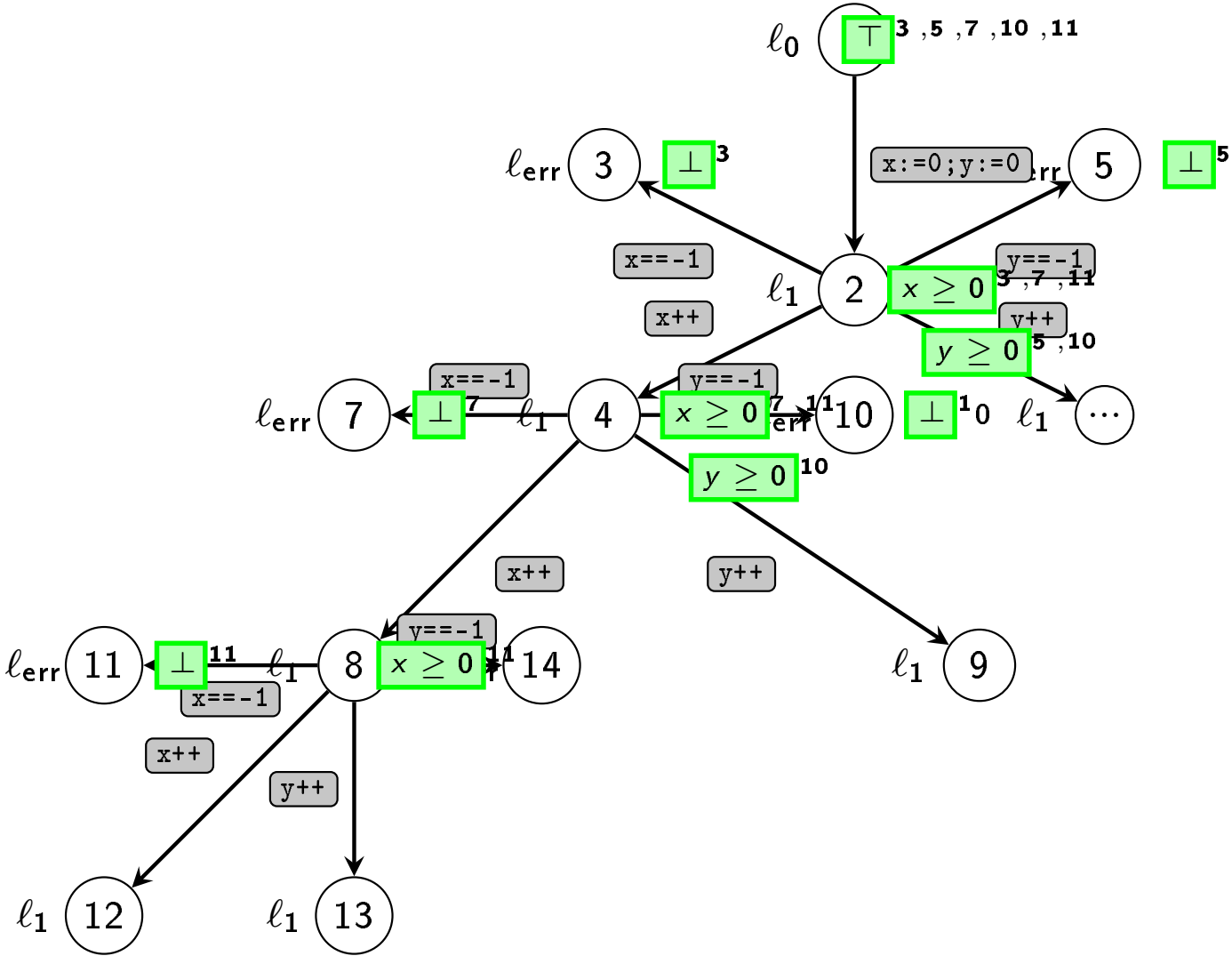

ℓ_0
1
$\top$
3 ,5 ,7 ,10
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
x==-1
ℓ_1
2
$x \geq 0$
3 ,7
y==-1
x++
y++
5 ,10
$y \geq 0$
x==-1
y==-1
ℓ_{err}
7
$\bot$
7
ℓ_1
4
$x \geq 0$
7
10
$\bot$
10
ℓ_1
...
$y \geq 0$
10
x++
y++
ℓ_1
8
ℓ_1
9

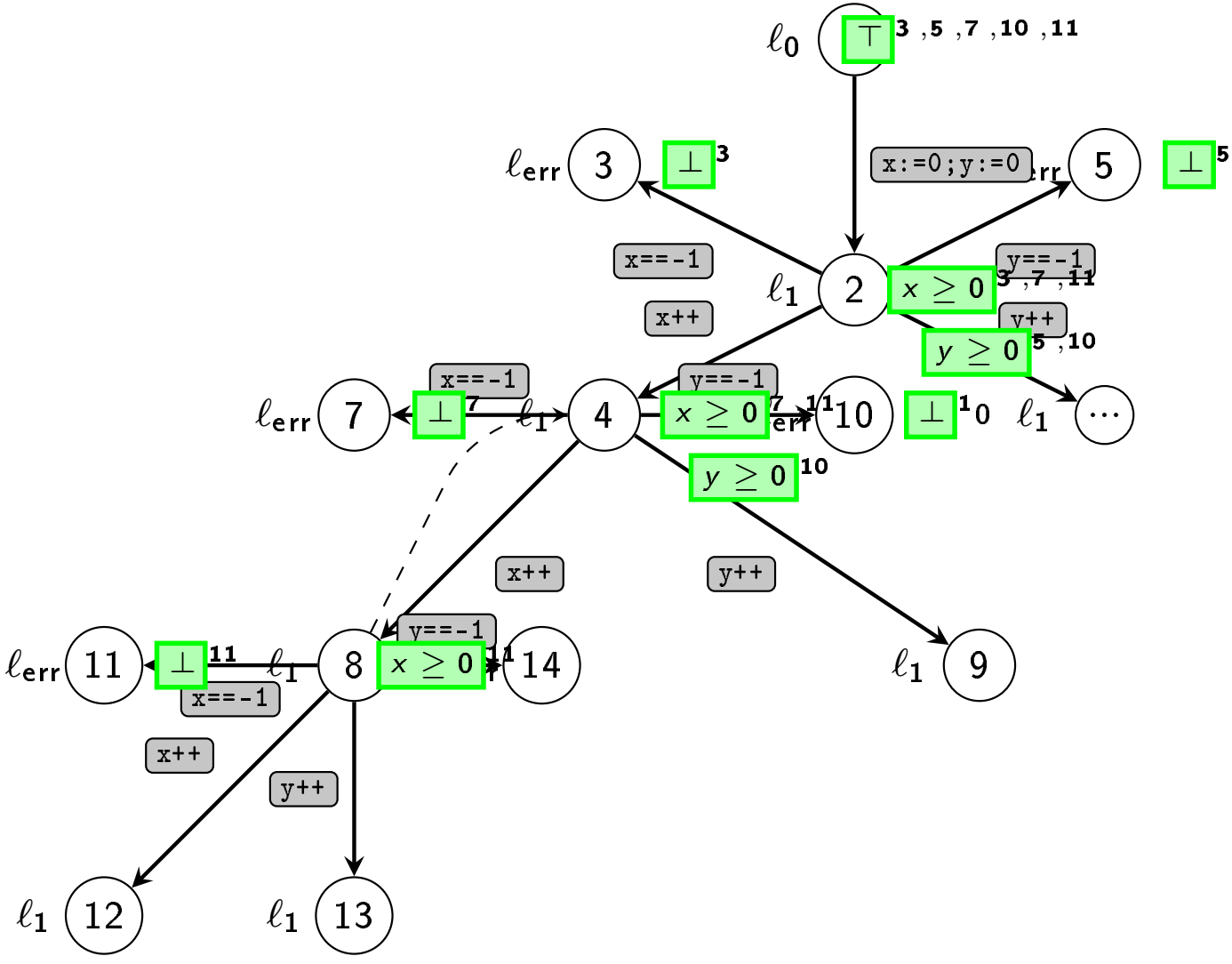

ℓ_0
1
$\top$
3 ,5 ,7 ,10
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
x==-1
ℓ_1
2
y==-1
3 ,7
$x \geq 0$
x++
y++
5 ,10
$y \geq 0$
x==-1
y==-1
ℓ_{err}
7
$\bot$
7
ℓ_1
4
$x \geq 0$
7
10
$\bot$
10
ℓ_1
$\cdots$
$y \geq 0$
10
x++
y++
ℓ_{err}
11
ℓ_1
8
ℓ_1
9
x==-1

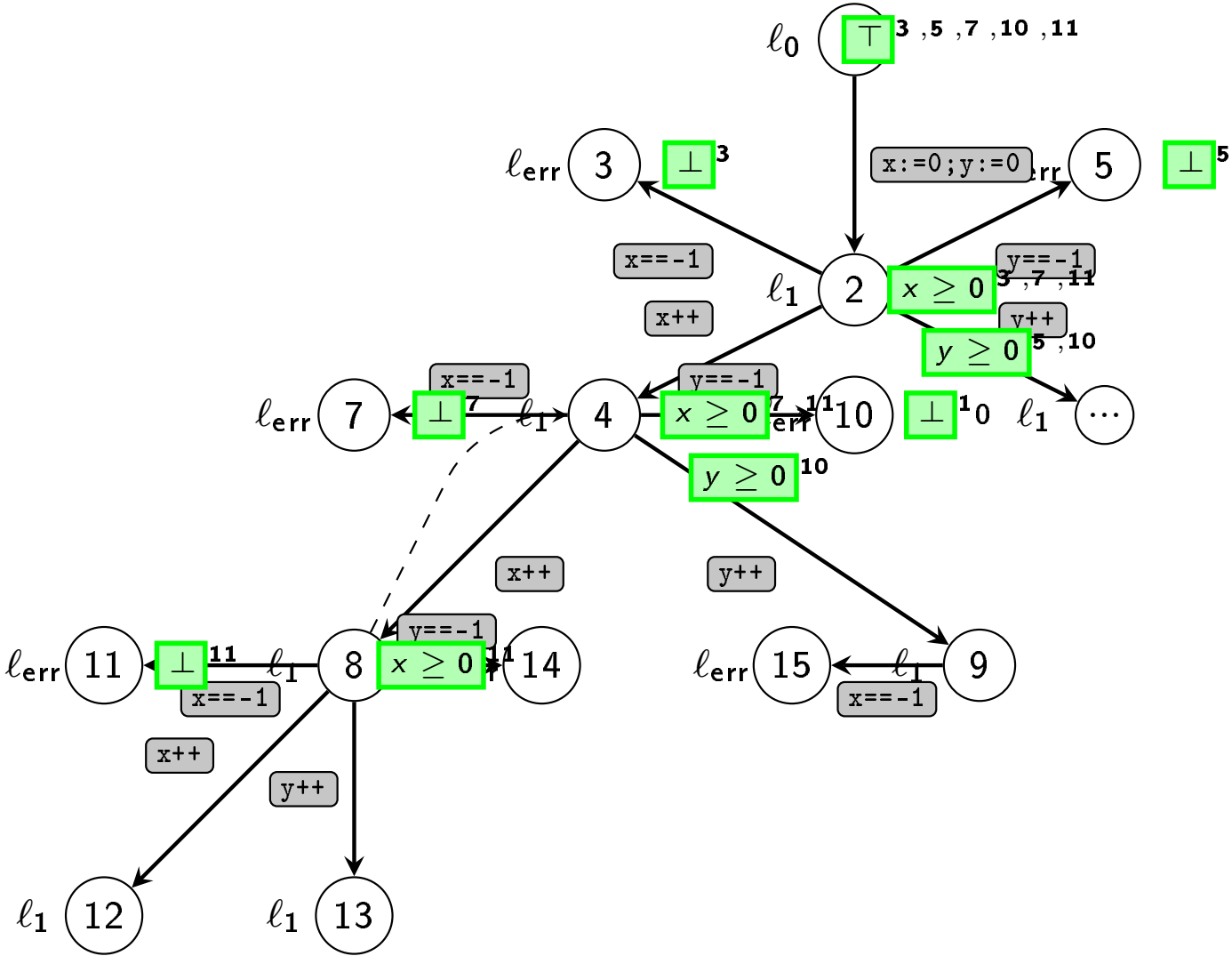

ℓ_0
1
$\top$ 3 ,5 ,7 ,10
ℓ_{err}
3
$\bot$ 3
x:=0;y:=0
5
$\bot$ 5
x==-1
ℓ_1
2
y==-1
$x \geq 0$ 3 ,7
x++
y++
$y \geq 0$ 5 ,10
x==-1
y==-1
ℓ_{err}
7
$\bot$ 7
ℓ_1
4
$x \geq 0$ 7
10
$\bot$ 10
ℓ_1
…
$y \geq 0$ 10
x++
y++
ℓ_{err}
11
ℓ_1
8
ℓ_1
9
x==-1
x++
ℓ_1
12

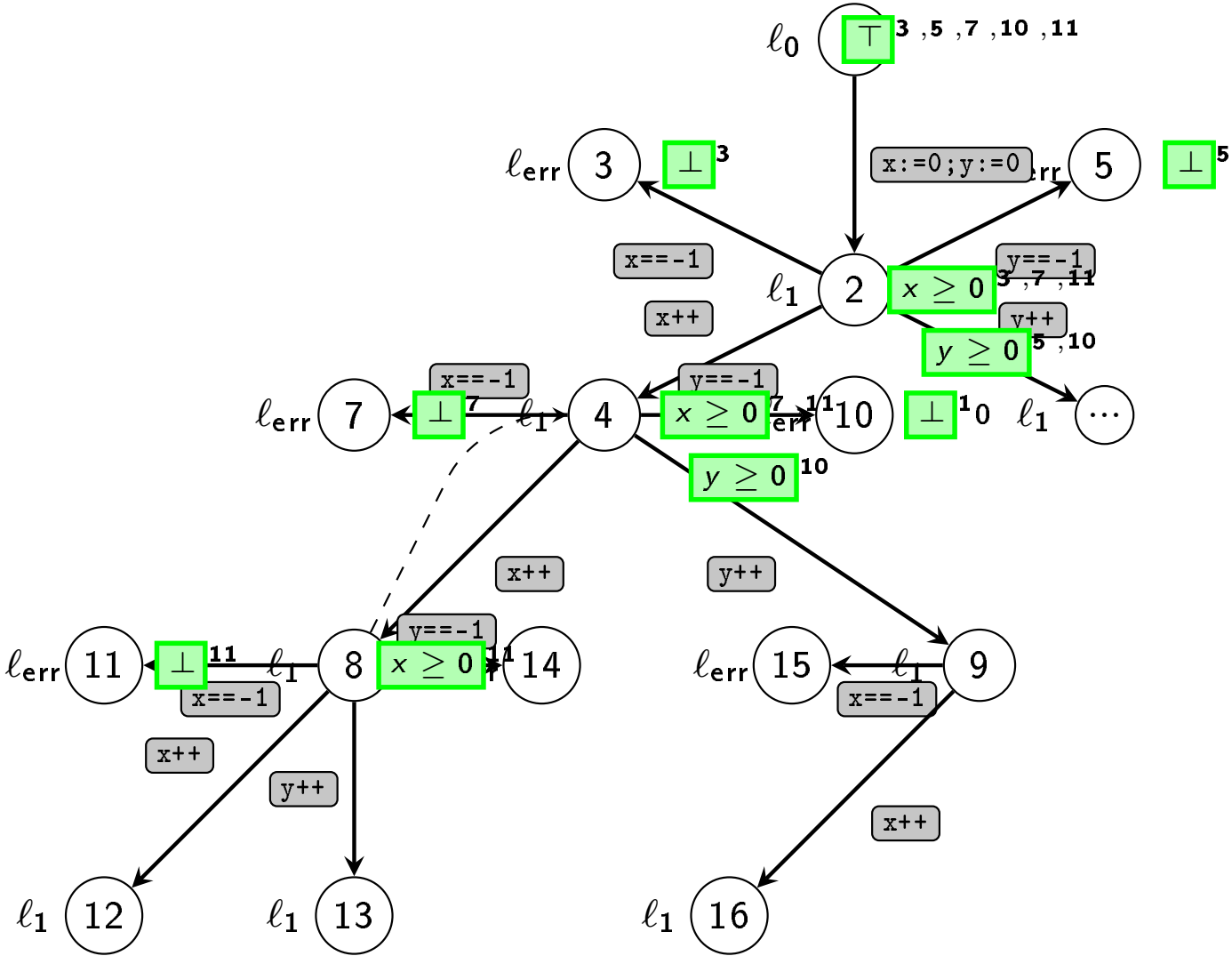

ℓ_0
1
$\top$
3 ,5 ,7 ,10
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
x==-1
ℓ_1
2
$x \geq 0$
3 ,7
x++
y==-1
y++
5 ,10
$y \geq 0$
ℓ_{err}
7
x==-1
$\bot$
7
ℓ_1
4
y==-1
$x \geq 0$
7
10
$\bot$
10
ℓ_1
$\cdots$
$y \geq 0$
10
x++
y++
ℓ_{err}
11
ℓ_1
8
ℓ_1
9
x==-1
x++
y++
ℓ_1
12
ℓ_1
13

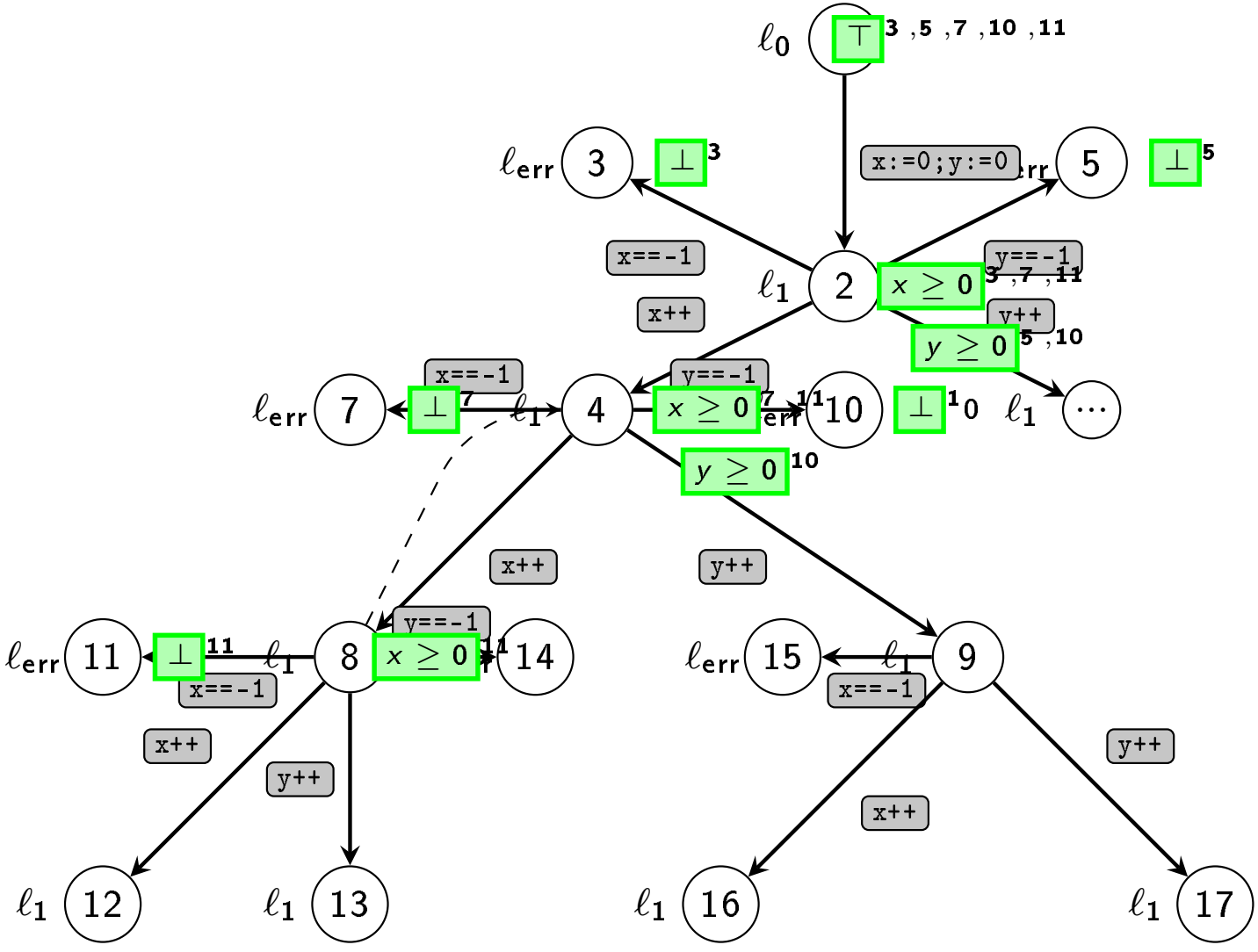

ℓ_0
1
$\top$
3 ,5 ,7 ,10
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
x==-1
ℓ_1
2
$x \geq 0$
3 ,7
y==-1
x++
y++
5 ,10
$y \geq 0$
x==-1
y==-1
ℓ_{err}
7
$\bot$
7
ℓ_1
4
$x \geq 0$
7
10
$\bot$
10
ℓ_1
…
$y \geq 0$
10
x++
y++
y==-1
ℓ_{err}
11
ℓ_1
8
ℓ_{err}
14
ℓ_1
9
x==-1
x++
y++
ℓ_1
12
ℓ_1
13

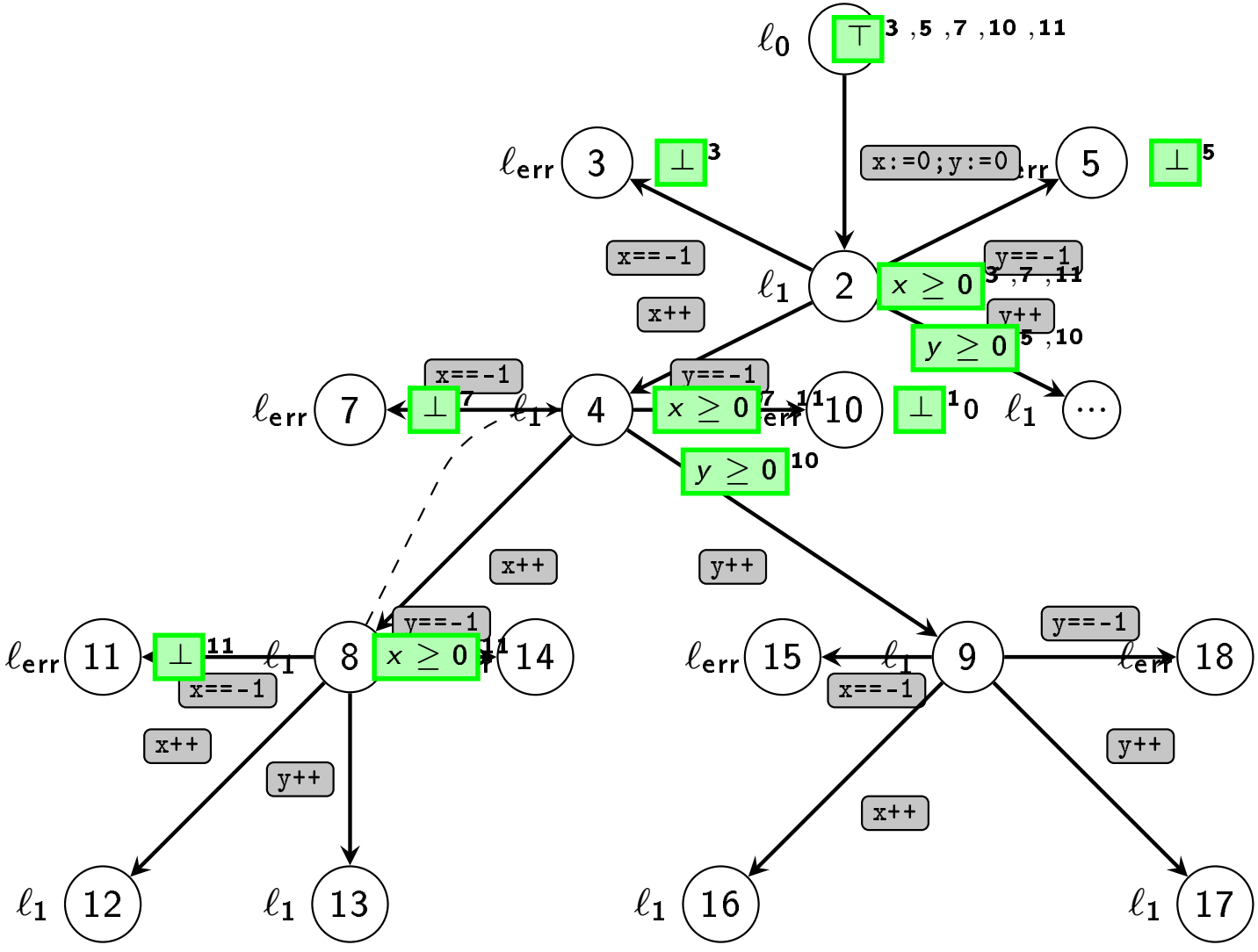

ℓ_0
$\top$ 3 ,5 ,7 ,10 ,11
ℓ_{err}
3
$\bot$ 3
x:=0;y:=0
5
$\bot$ 5
x==-1
ℓ_1
2
$x \geq 0$ 3 ,7 ,11
y==-1
x++
y++
$y \geq 0$ 5 ,10
x==-1
y==-1
ℓ_{err}
7
$\bot$ 7
ℓ_1
4
$x \geq 0$ 7 ,11
10
$\bot$ 10
ℓ_1
…
$y \geq 0$ 10
x++
y++
y==-1
ℓ_{err}
11
$\bot$ 11
ℓ_1
8
$x \geq 0$ 11
14
ℓ_1
9
x==-1
x++
y++
ℓ_1
12
ℓ_1
13

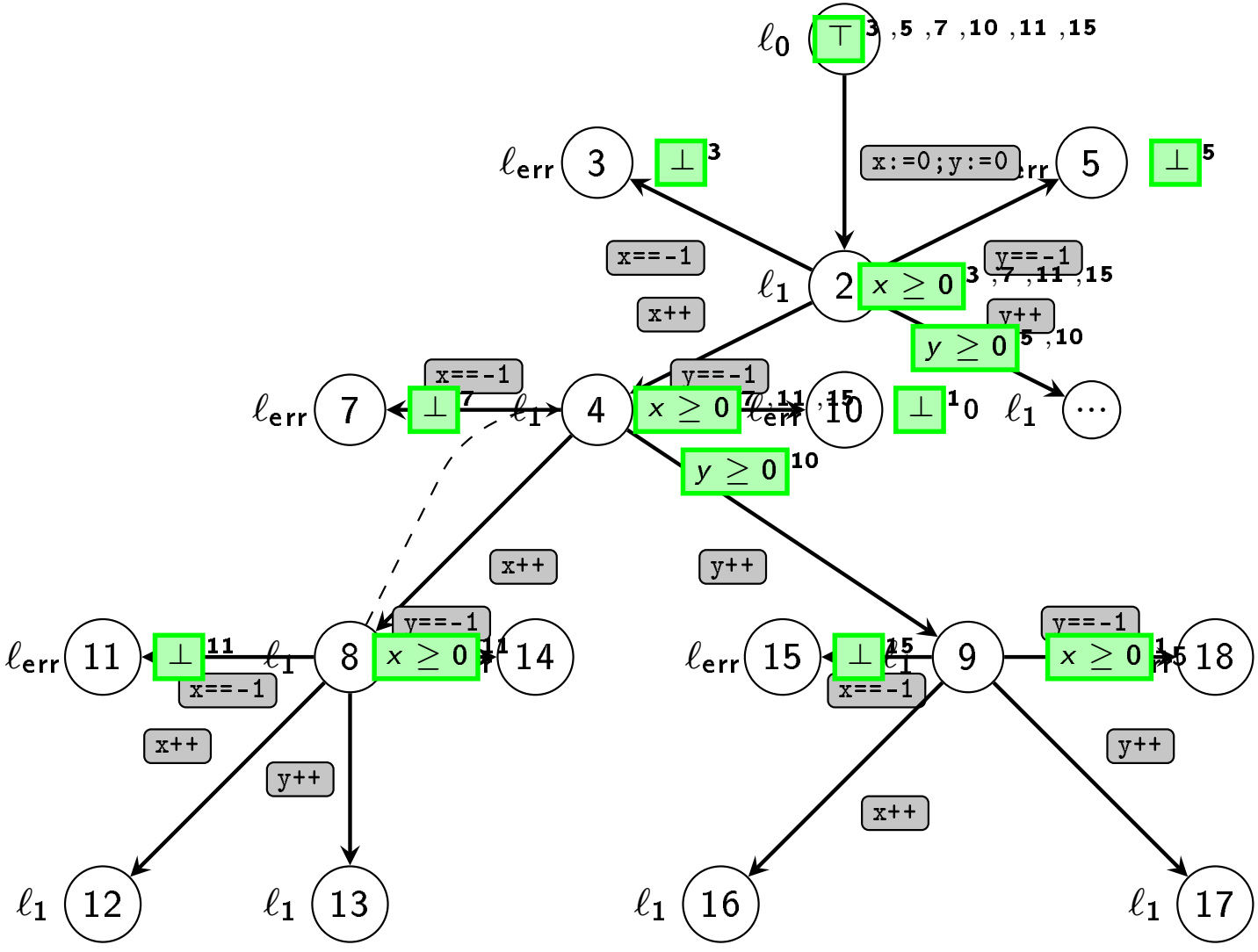

ℓ_0
$\top$ 3 ,5 ,7 ,10 ,11
ℓ_{err} 3 $\bot$ 3
x:=0;y:=0
5 $\bot$ 5
x==-1
ℓ_1 2 $x \geq 0$ 3 ,7 ,11
y==-1
x++
y++
$y \geq 0$ 5 ,10
x==-1
y==-1
ℓ_{err} 7 $\bot$ 7 ℓ_1 4 $x \geq 0$ 7 ,11 10 $\bot$ 10 ℓ_1 ...
$y \geq 0$ 10
x++
y++
y==-1
ℓ_{err} 11 $\bot$ 11 ℓ_1 8 $x \geq 0$ 11 14
ℓ_1 9
x==-1
x++
y++
ℓ_1 12
ℓ_1 13

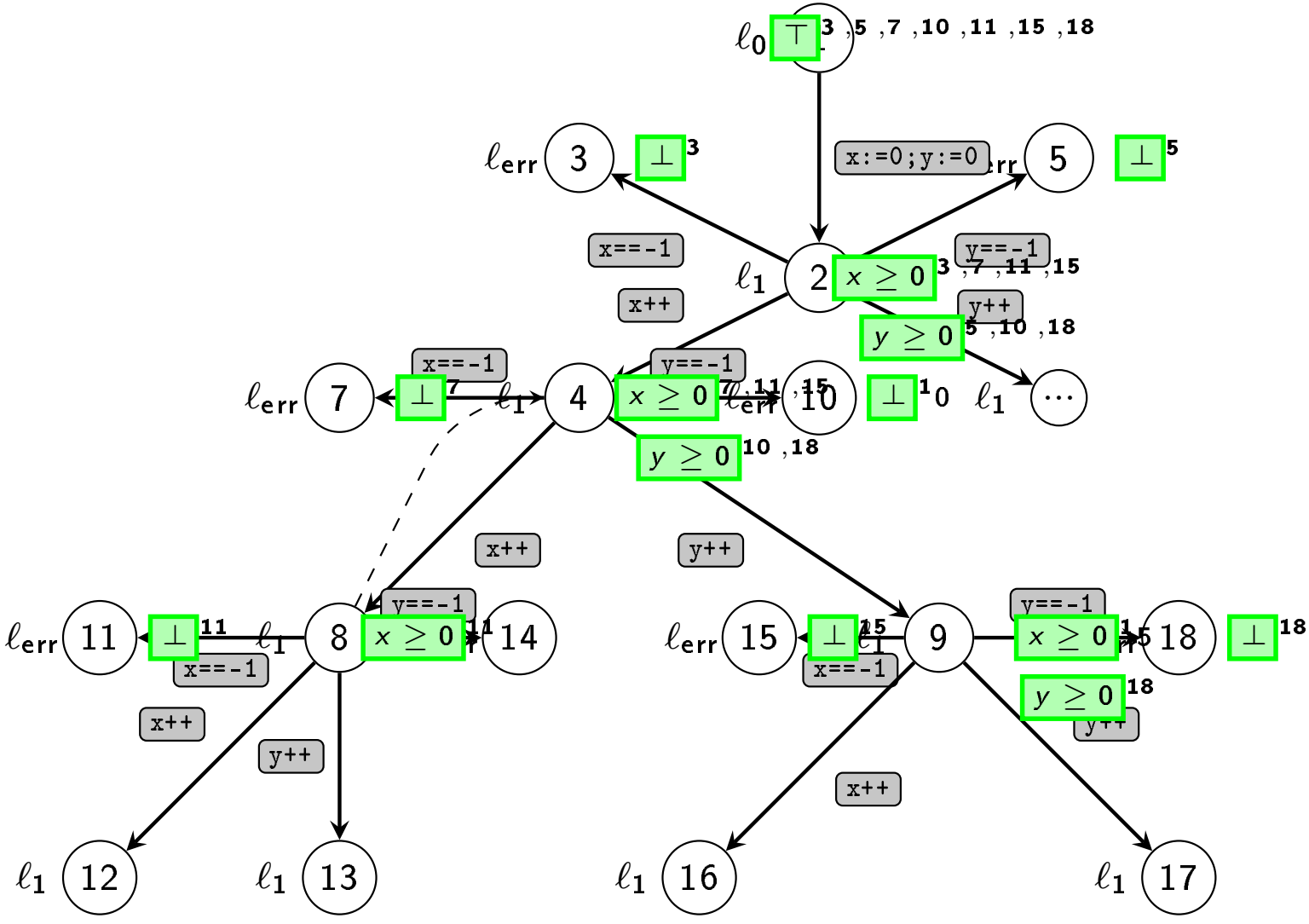

ℓ_0
$\top$ 3 ,5 ,7 ,10 ,11
ℓ_{err}
3
$\bot$ 3
x:=0;y:=0
5
$\bot$ 5
x==-1
ℓ_1
2
$x \geq 0$ 3 ,7 ,11
y==-1
x++
y++
$y \geq 0$ 5 ,10
ℓ_{err}
7
x==-1
$\bot$ 7
ℓ_1
4
y==-1
$x \geq 0$ 7 ,11
10
$\bot$ 10
ℓ_1
…
$y \geq 0$ 10
x++
y++
ℓ_{err}
11
$\bot$ 11
ℓ_1
8
y==-1
$x \geq 0$ 11
14
x==-1
ℓ_{err}
15
ℓ_1
9
x==-1
x++
y++
ℓ_1
12
ℓ_1
13

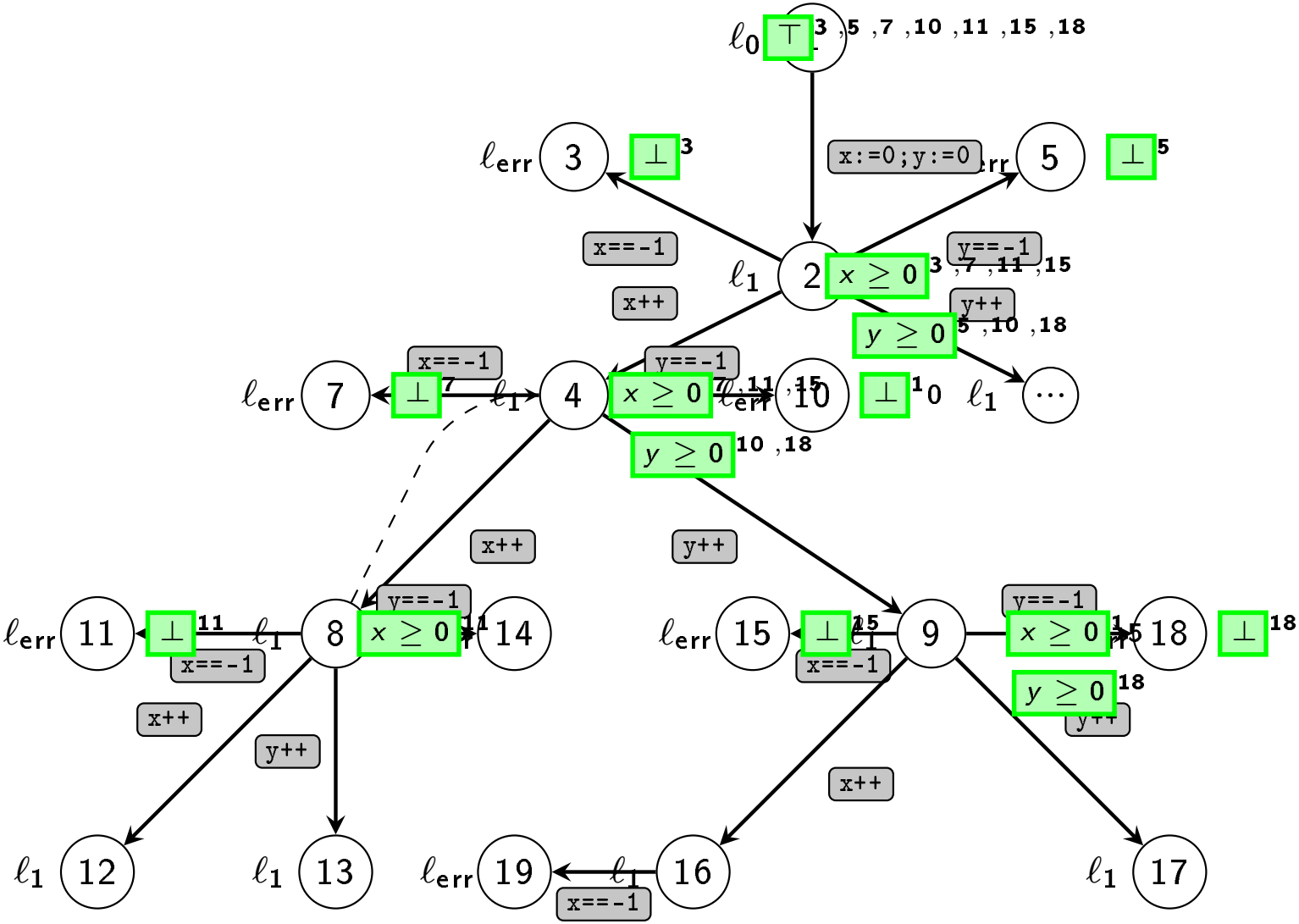

ℓ_0
$\top$
3 ,5 ,7 ,10 ,11
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
x==-1
ℓ_1
2
$x \geq 0$
3 ,7 ,11
y==-1
x++
y++
$y \geq 0$
5 ,10
x==-1
y==-1
ℓ_{err}
7
$\bot$
7
ℓ_1
4
$x \geq 0$
7 ,11
10
$\bot$
10
ℓ_1
$\cdots$
$y \geq 0$
10
x++
y++
y==-1
ℓ_{err}
11
$\bot$
11
ℓ_1
8
$x \geq 0$
11
14
ℓ_{err}
15
ℓ_1
9
x==-1
x==-1
x++
y++
x++
ℓ_1
12
ℓ_1
13
ℓ_1
16

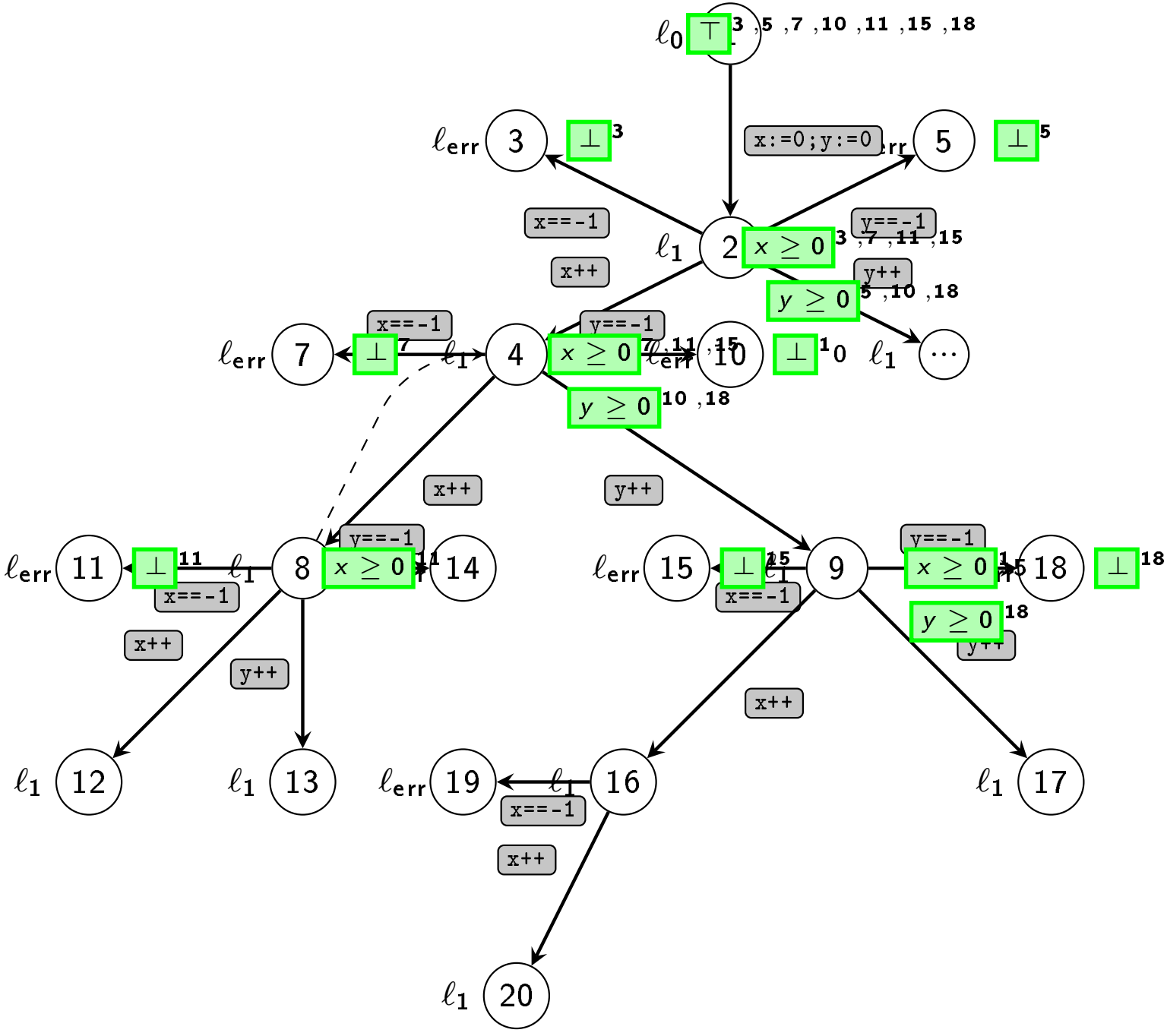

ℓ_0
$\top$ 3 ,5 ,7 ,10 ,11
ℓ_{err}
3
$\bot$ 3
x:=0;y:=0
5
$\bot$ 5
x==-1
ℓ_1
2
$x \geq 0$ 3 ,7 ,11
y==-1
x++
y++
$y \geq 0$ 5 ,10
ℓ_{err}
7
x==-1
$\bot$ 7
ℓ_1
4
y==-1
$x \geq 0$ 7 ,11
10
$\bot$ 10
ℓ_1
...
$y \geq 0$ 10
x++
y++
ℓ_{err}
11
$\bot$ 11
ℓ_1
8
y==-1
$x \geq 0$ 11
14
x==-1
ℓ_{err}
15
ℓ_1
9
x==-1
x++
y++
y++
x++
ℓ_1
12
ℓ_1
13
ℓ_1
16
ℓ_1
17

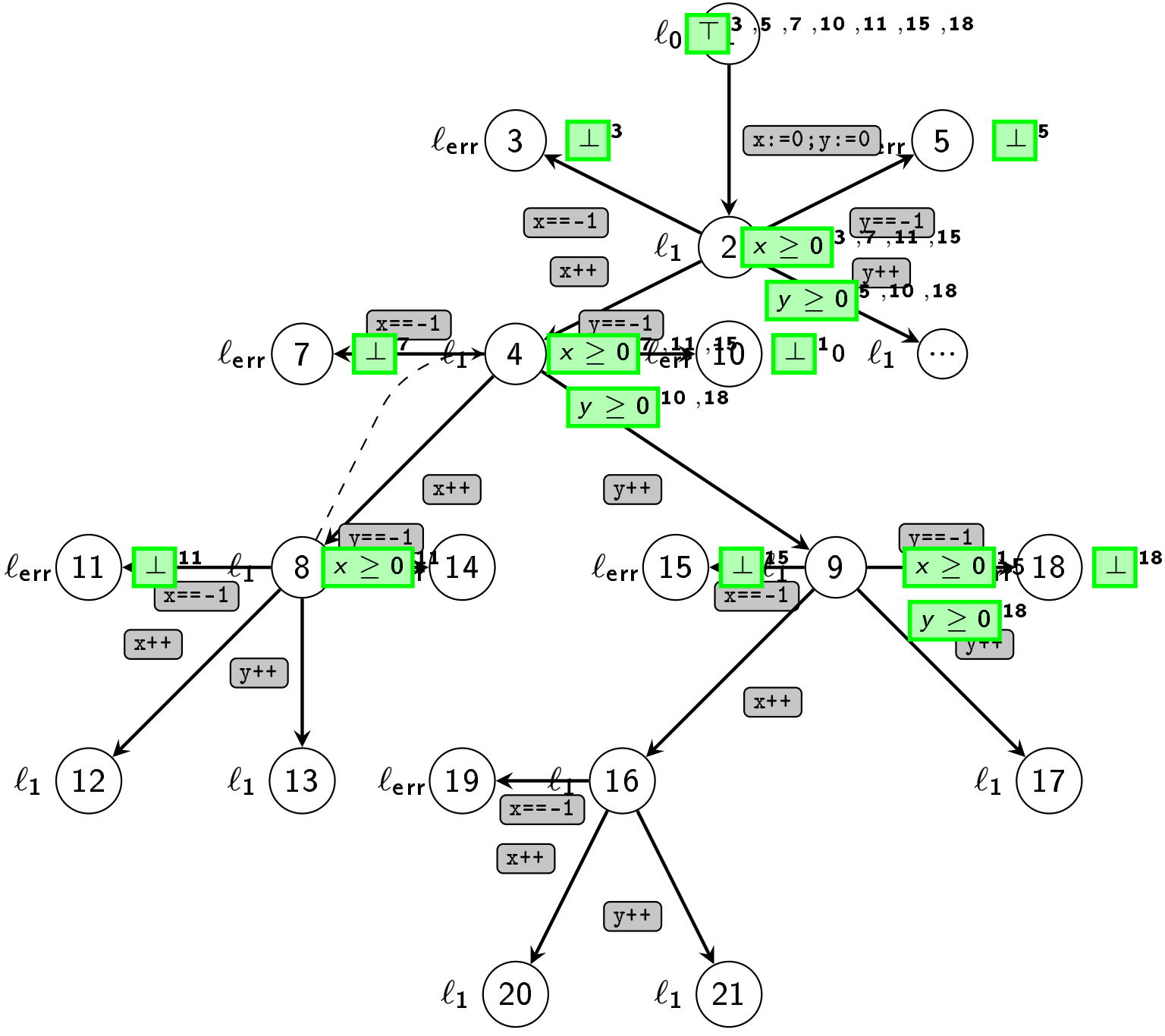

ℓ_0
$\top$ 3 ,5 ,7 ,10 ,11
ℓ_{err}
3
$\bot$ 3
x:=0;y:=0
5
$\bot$ 5
x==-1
y==-1
ℓ_1
2
$x \geq 0$ 3 ,7 ,11
x++
y++
$y \geq 0$ 5 ,10
x==-1
y==-1
ℓ_{err}
7
$\bot$ 7
ℓ_1
4
$x \geq 0$ 7 ,11
10
$\bot$ 10
ℓ_1
...
$y \geq 0$ 10
x++
y++
y==-1
ℓ_{err}
11
$\bot$ 11
ℓ_1
8
$x \geq 0$ 11
14
x==-1
x++
y++
ℓ_{err}
15
ℓ_1
9
y==-1
18
x==-1
y++
x++
ℓ_1
12
ℓ_1
13
ℓ_1
16
ℓ_1
17

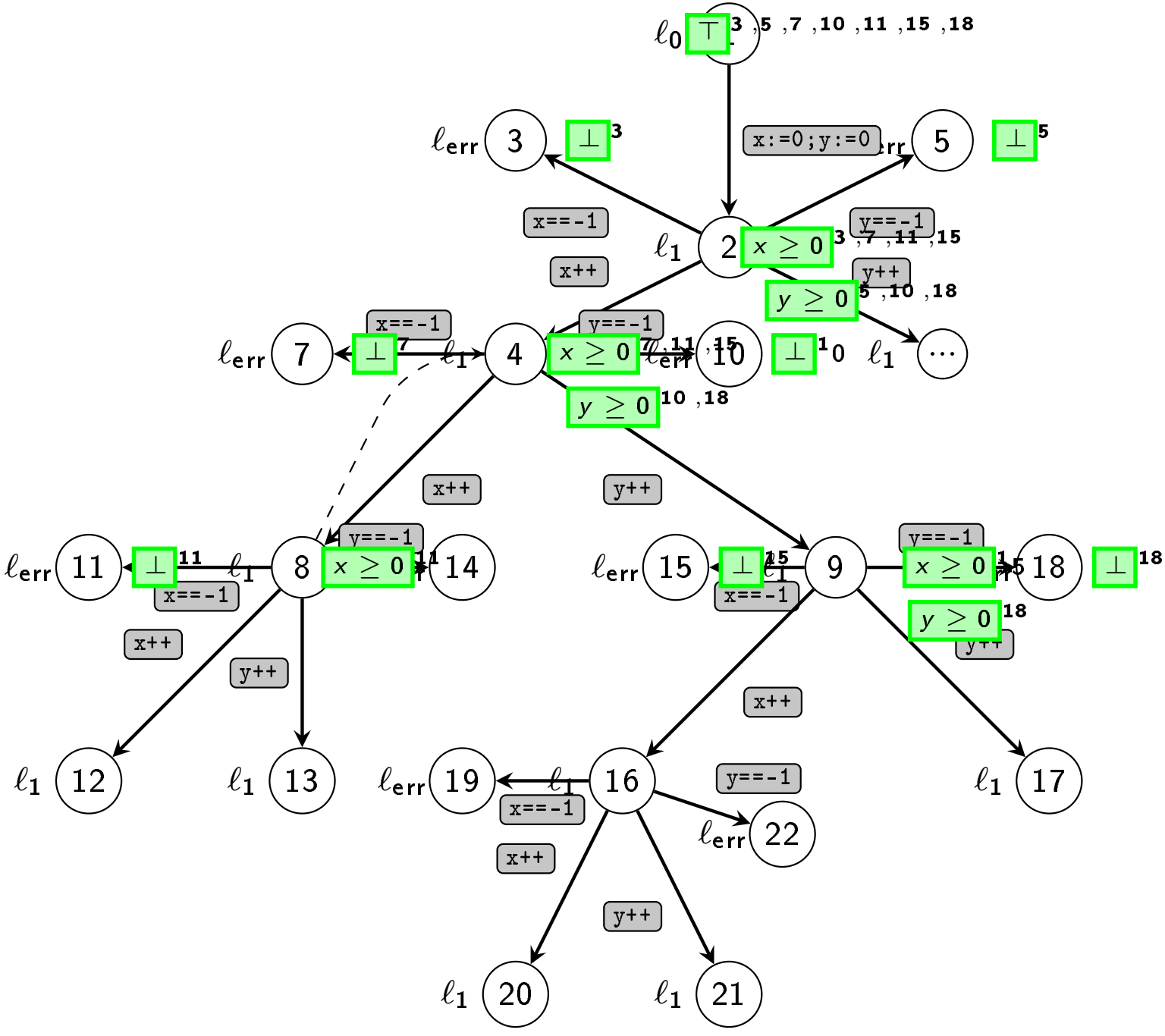

ℓ_0
$\top$
3 ,5 ,7 ,10 ,11 ,15
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
x==-1
ℓ_1
2
$x \geq 0$
3 ,7 ,11 ,15
y==-1
x++
y++
$y \geq 0$
5 ,10
ℓ_{err}
7
x==-1
$\bot$
7
ℓ_1
4
$x \geq 0$
7 ,11 ,15
y==-1
ℓ_{err}
10
$\bot$
10
ℓ_1
$\cdots$
$y \geq 0$
10
x++
y++
ℓ_{err}
11
$\bot$
11
ℓ_1
8
$x \geq 0$
11
y==-1
14
x==-1
ℓ_{err}
15
$\bot$
15
ℓ_1
9
$x \geq 0$
15
y==-1
18
x==-1
x++
y++
y++
x++
ℓ_1
12
ℓ_1
13
ℓ_1
16
ℓ_1
17

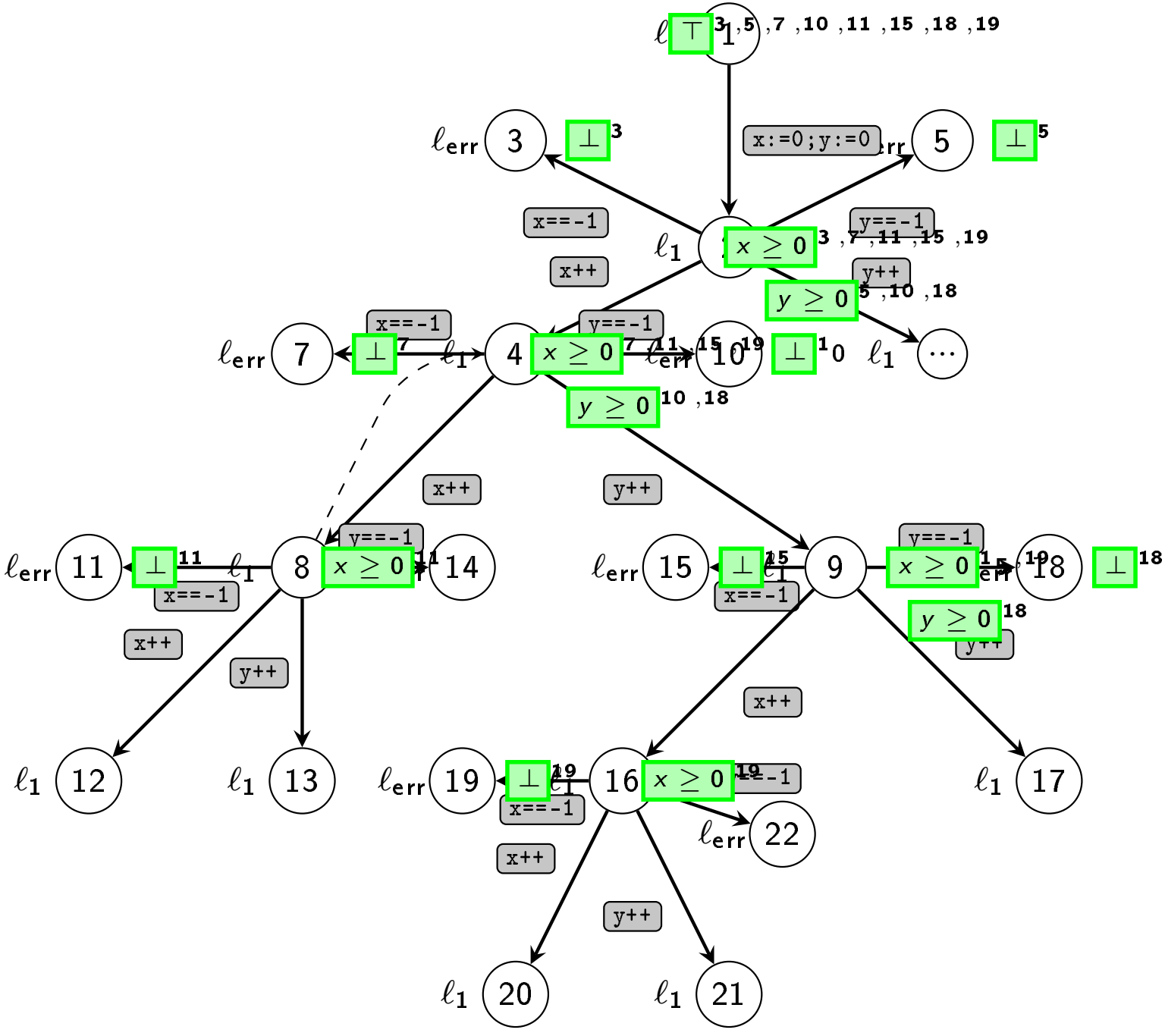

ℓ_0
$\top$
3 ,5 ,7 ,10 ,11 ,15 ,18
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
x==-1
y==-1
ℓ_1
2
$x \geq 0$
3 ,7 ,11 ,15
x++
y++
$y \geq 0$
5 ,10 ,18
x==-1
y==-1
ℓ_{err}
7
$\bot$
7
ℓ_1
4
$x \geq 0$
7 ,11 ,15
ℓ_{err}
10
$\bot$
10
ℓ_1
$\cdots$
$y \geq 0$
10 ,18
x++
y++
y==-1
ℓ_{err}
11
$\bot$
11
ℓ_1
8
$x \geq 0$
11
14
x==-1
x++
y++
ℓ_1
12
ℓ_1
13
y==-1
ℓ_{err}
15
$\bot$
15
ℓ_1
9
$x \geq 0$
15
18
$\bot$
18
x==-1
$y \geq 0$
18
x++
ℓ_1
16
ℓ_1
17

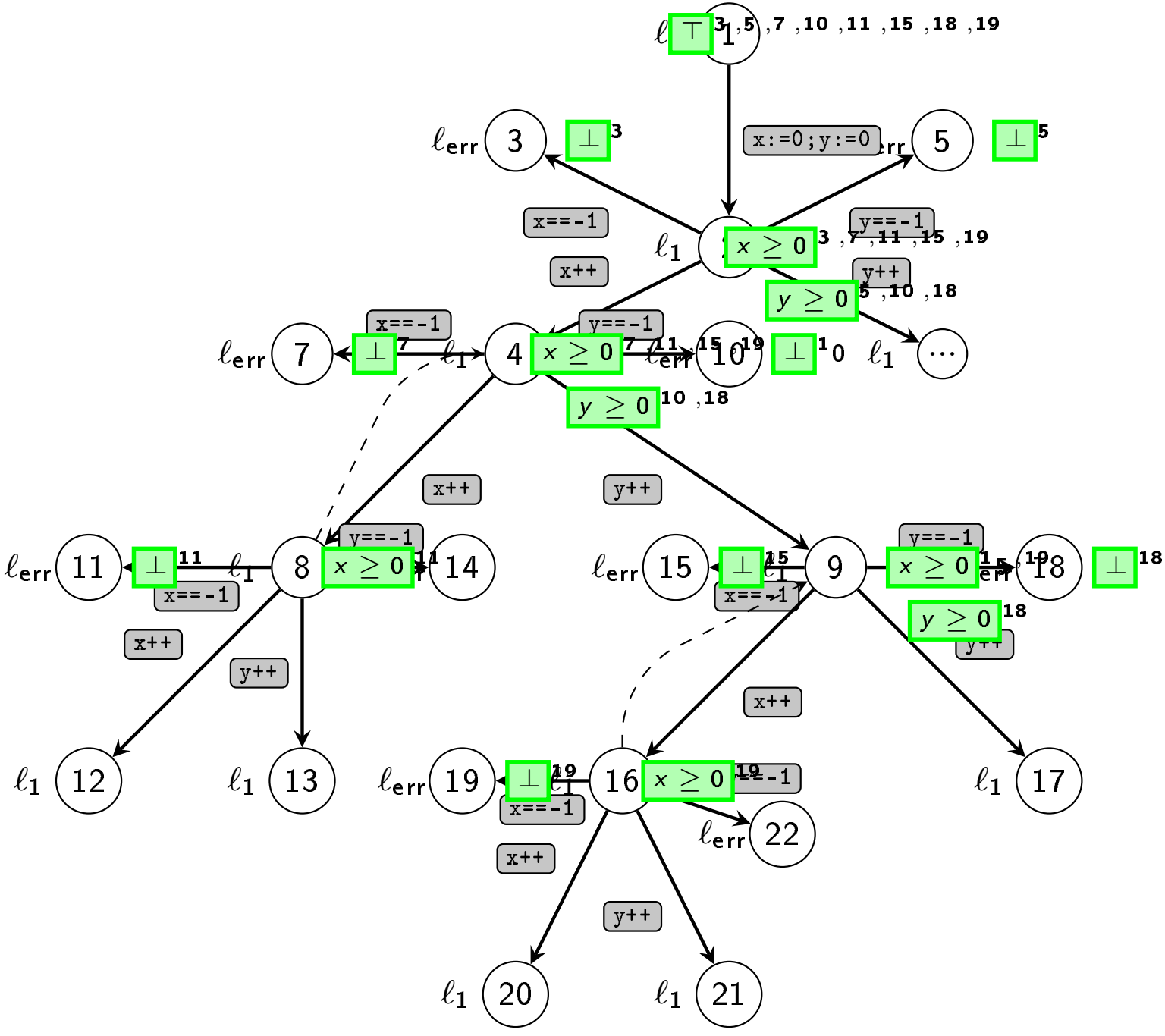

ℓ_0
$\top$
3 ,5 ,7 ,10 ,11 ,15 ,18
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
x==-1
y==-1
ℓ_1
2
$x \geq 0$
3 ,7 ,11 ,15
x++
y++
$y \geq 0$
5 ,10 ,18
x==-1
y==-1
ℓ_{err}
7
$\bot$
7
ℓ_1
4
$x \geq 0$
7 ,11 ,15
ℓ_{err}
10
$\bot$
10
ℓ_1
$\cdots$
$y \geq 0$
10 ,18
x++
y++
y==-1
ℓ_{err}
11
$\bot$
11
ℓ_1
8
$x \geq 0$
11
14
x==-1
x++
y++
ℓ_{err}
15
$\bot$
15
ℓ_1
9
y==-1
$x \geq 0$
15
18
$\bot$
18
x==-1
$y \geq 0$
18
x++
ℓ_1
12
ℓ_1
13
ℓ_{err}
19
ℓ_1
16
x==-1
x++
ℓ_1
17

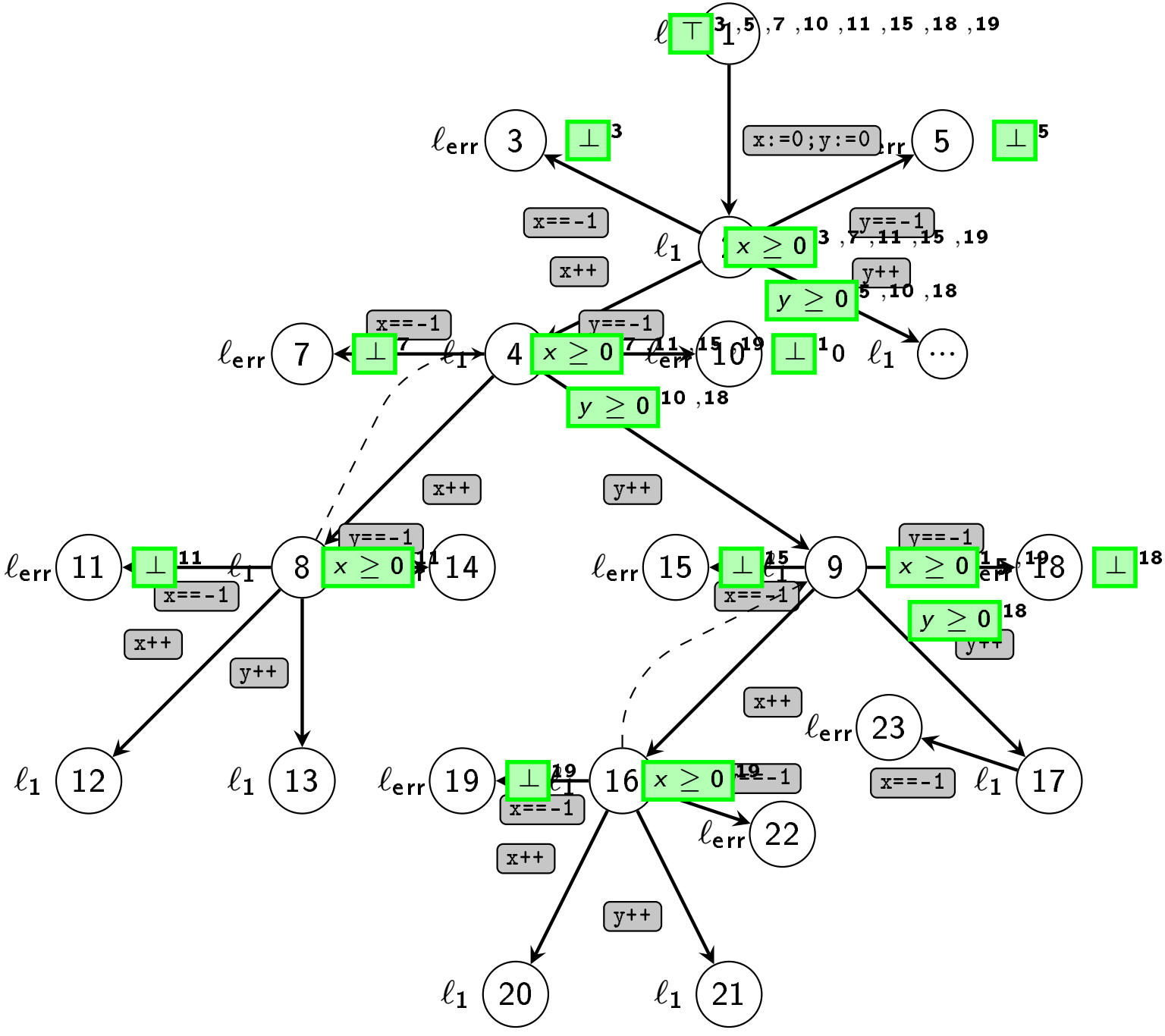

ℓ_0
$\top$
3 ,5 ,7 ,10 ,11 ,15 ,18
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
x==-1
y==-1
ℓ_1
2
$x \geq 0$
3 ,7 ,11 ,15
x++
y++
$y \geq 0$
5 ,10 ,18
ℓ_{err}
7
x==-1
$\bot$
7
ℓ_1
4
y==-1
$x \geq 0$
7 ,11 ,15
ℓ_{err}
10
$\bot$
10
ℓ_1
...
$y \geq 0$
10 ,18
x++
y++
ℓ_{err}
11
$\bot$
11
ℓ_1
8
y==-1
$x \geq 0$
11
14
x==-1
x++
y++
ℓ_{err}
15
$\bot$
15
ℓ_1
9
y==-1
$x \geq 0$
15
18
$\bot$
18
x==-1
$y \geq 0$
18
x++
ℓ_1
12
ℓ_1
13
ℓ_{err}
19
ℓ_1
16
ℓ_1
17
x==-1
x++
ℓ_1
20

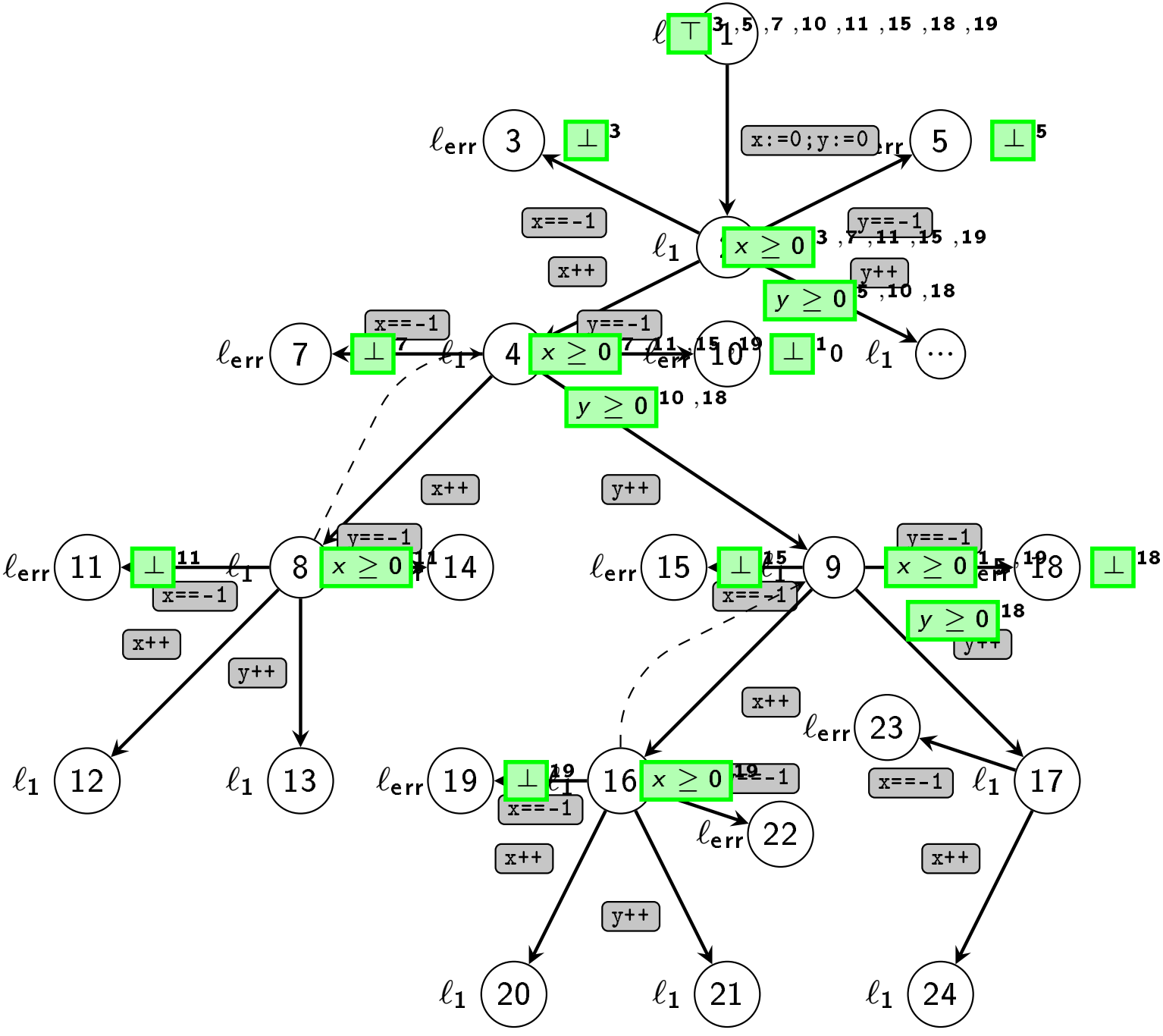

ℓ_0
$\top$
3 ,5 ,7 ,10 ,11 ,15 ,18
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
x==-1
y==-1
ℓ_1
2
$x \geq 0$
3 ,7 ,11 ,15
x++
y++
$y \geq 0$
5 ,10 ,18
ℓ_{err}
7
x==-1
$\bot$
7
ℓ_1
4
y==-1
$x \geq 0$
7 ,11 ,15
ℓ_{err}
10
$\bot$
10
ℓ_1
$\cdots$
$y \geq 0$
10 ,18
x++
y++
ℓ_{err}
11
$\bot$
11
ℓ_1
8
y==-1
$x \geq 0$
11
14
x==-1
x++
y++
ℓ_1
12
ℓ_1
13
ℓ_{err}
15
$\bot$
15
ℓ_1
9
y==-1
$x \geq 0$
15
18
$\bot$
18
x==-1
$y \geq 0$
18
x++
ℓ_{err}
19
ℓ_1
16
ℓ_1
17
x==-1
x++
y++
ℓ_1
20
ℓ_1
21

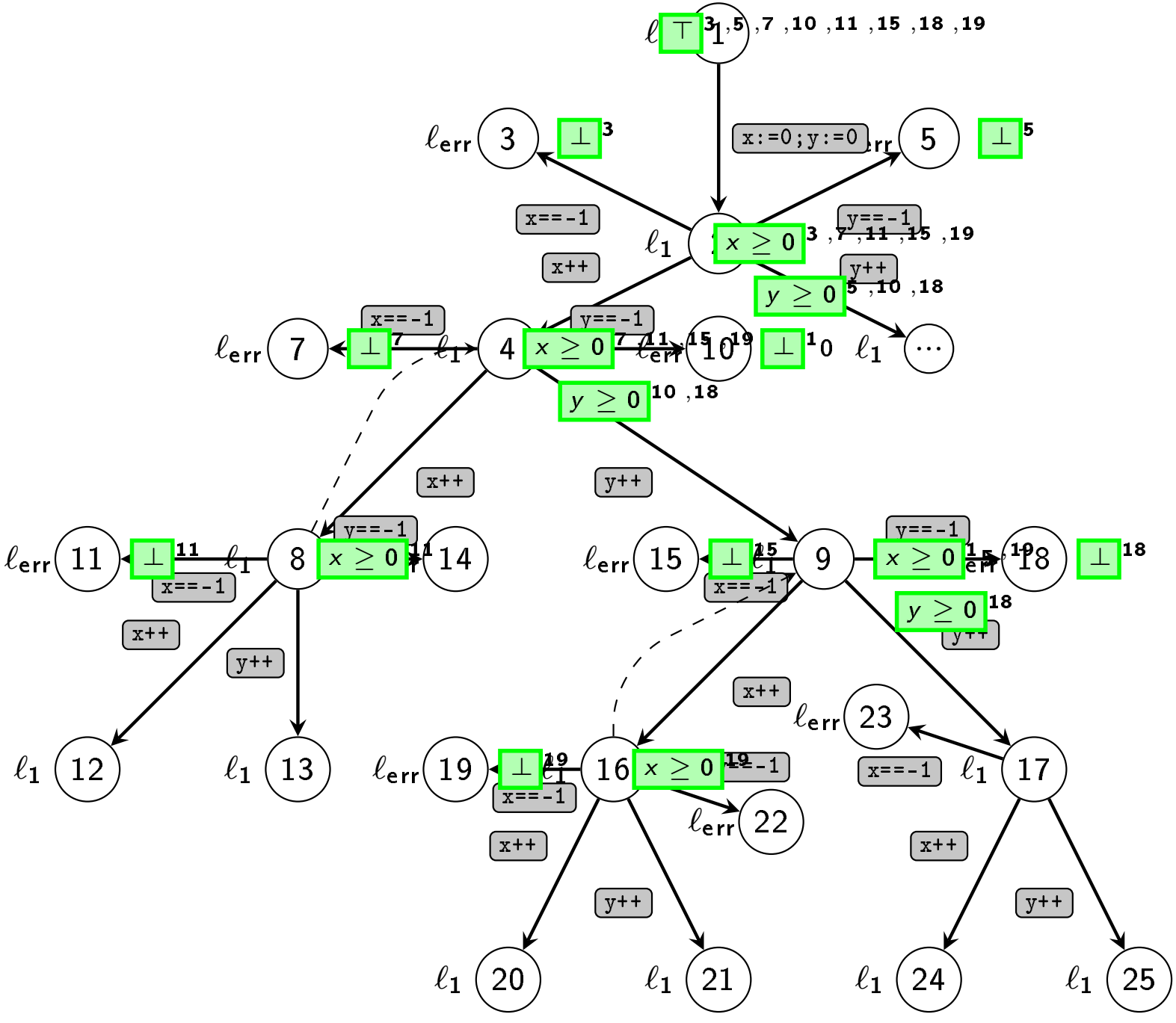

ℓ_0
$\top$
3 ,5 ,7 ,10 ,11 ,15 ,18
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
x==-1
y==-1
ℓ_1
2
$x \geq 0$
3 ,7 ,11 ,15
x++
y++
$y \geq 0$
5 ,10 ,18
ℓ_{err}
7
x==-1
$\bot$
7
ℓ_1
4
y==-1
$x \geq 0$
7 ,11 ,15
ℓ_{err}
10
$\bot$
10
ℓ_1
$\cdots$
$y \geq 0$
10 ,18
x++
y++
ℓ_{err}
11
$\bot$
11
ℓ_1
8
y==-1
$x \geq 0$
11
14
x==-1
x++
y++
ℓ_1
12
ℓ_1
13
ℓ_{err}
15
$\bot$
15
ℓ_1
9
y==-1
$x \geq 0$
15
18
$\bot$
18
x==-1
$y \geq 0$
18
x++
ℓ_{err}
19
ℓ_1
16
y==-1
x==-1
ℓ_{err}
22
x++
y++
ℓ_1
17
ℓ_1
20
ℓ_1
21

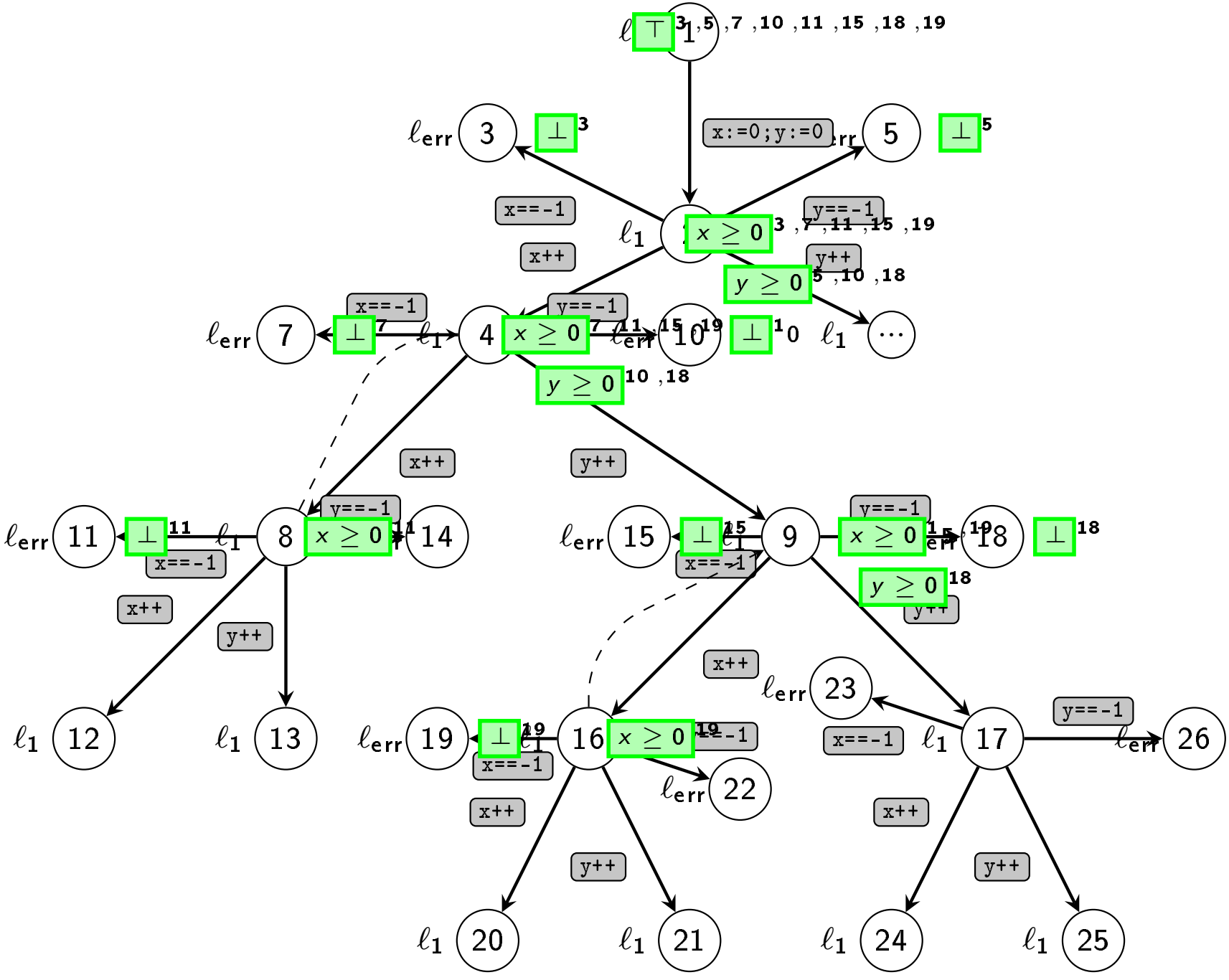

ℓ_{err}
3
$\bot$
x:=0;y:=0
5
$\bot$
ℓ_1
1
$\top$
x==-1
x++
y==-1
$x \geq 0$
y++
$y \geq 0$
ℓ_{err}
7
x==-1
$\bot$
ℓ_1
4
y==-1
$x \geq 0$
ℓ_{err}
10
$\bot$
ℓ_1
$\cdots$
$y \geq 0$
x++
y++
ℓ_{err}
11
$\bot$
ℓ_1
8
y==-1
$x \geq 0$
14
x==-1
x++
y++
ℓ_{err}
15
$\bot$
ℓ_1
9
y==-1
$x \geq 0$
18
$\bot$
x==-1
$y \geq 0$
y++
x++
ℓ_1
12
ℓ_1
13
ℓ_{err}
19
$\bot$
ℓ_1
16
$x \geq 0$
ℓ_1
17
x==-1
ℓ_{err}
22
x++
y++
ℓ_1
20
ℓ_1
21

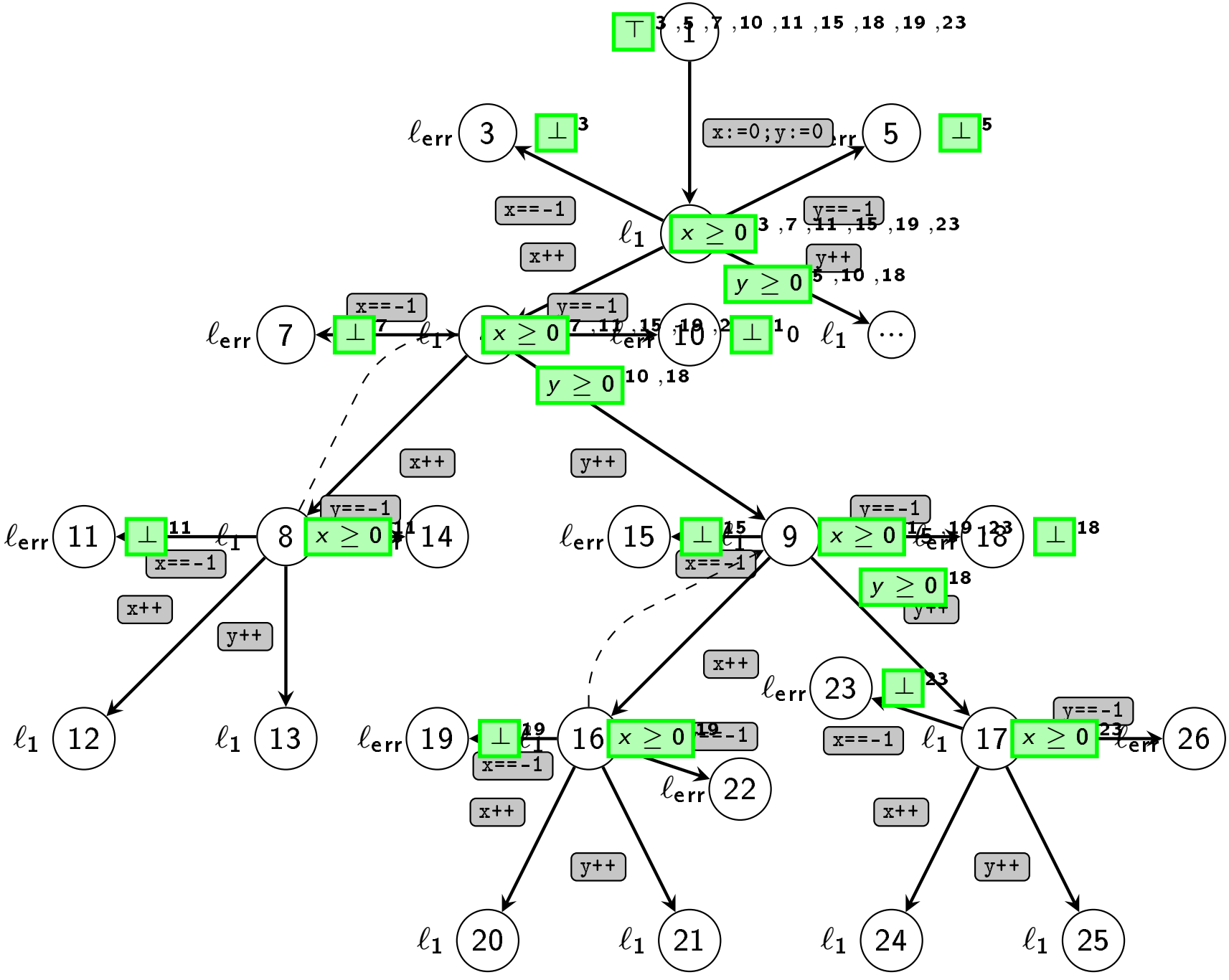

1
⊤
3,5,7,10,11,15,18,19
3
⊥
x:=0;y:=0
5
⊥
x==-1
x++
y==-1
y++
x ≥ 0
y ≥ 0
4
x ≥ 0
7
⊥
10
⊥
y ≥ 0
10,18
x==-1
y==-1
x++
y++
8
x ≥ 0
11
⊥
14
9
x ≥ 0
15
⊥
18
⊥
y ≥ 0
12
13
16
x ≥ 0
19
⊥
22
17
20
21

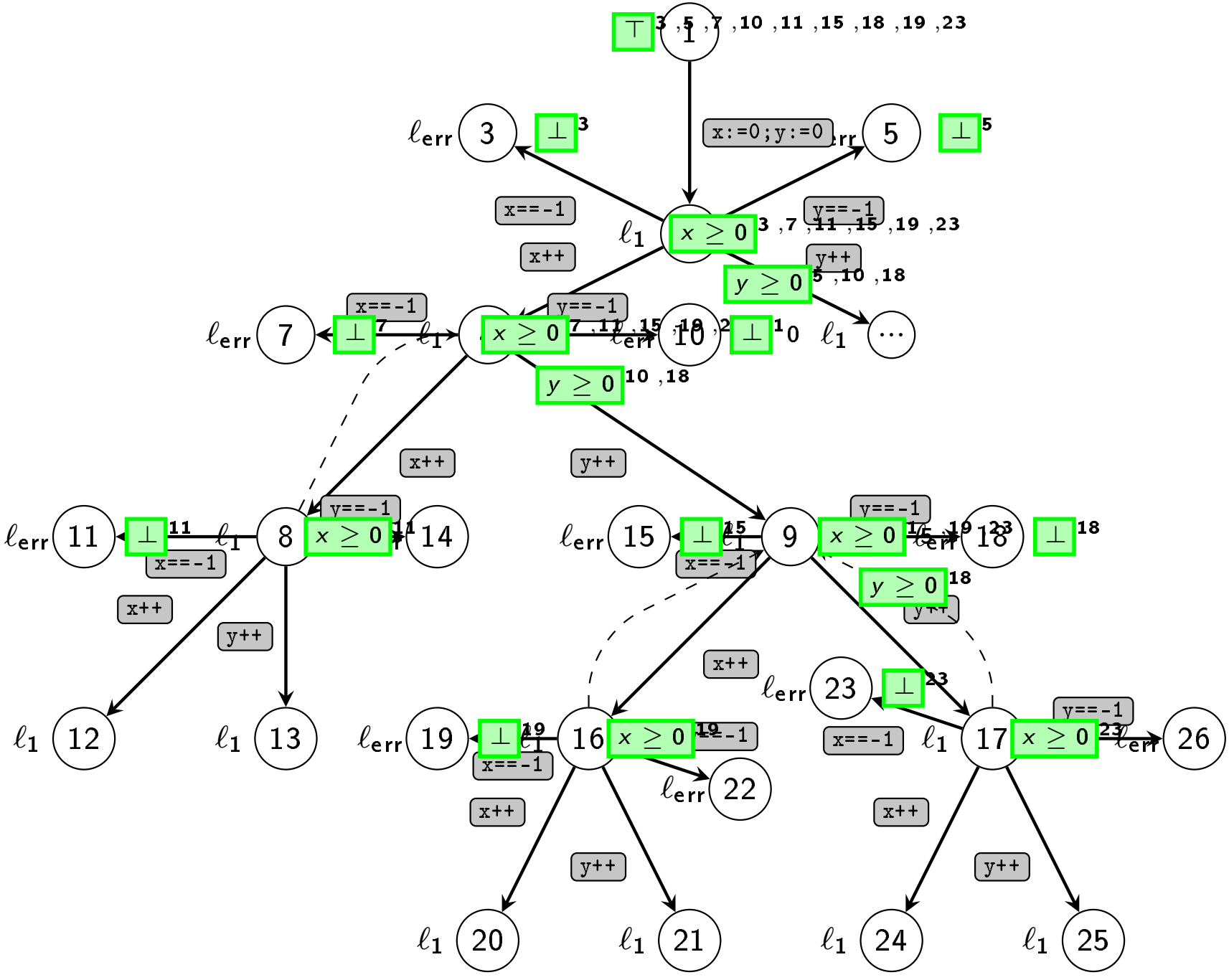

1
⊤
3,5,7,10,11,15,18,19
3
⊥
x:=0;y:=0
5
⊥
x==-1
x++
y==-1
y++
x ≥ 0
y ≥ 0
4
7
10
11
14
8
15
9
18
12
13
19
16
22
23
17
20
21
…


ℓ_{err}
$\top$
1
3,5,7,10,11,15,18,19
3
$\perp$
3
x:=0;y:=0
5
$\perp$
5
x==-1
x++
ℓ_1
$x \geq 0$
3,7,11,15,19
y==-1
y++
$y \geq 0$
5,10,18
7
$\perp$
7
4
$x \geq 0$
7,11,15,19
10
$\perp$
10
ℓ_1
...
$y \geq 0$
10,18
x++
y++
11
$\perp$
11
8
$x \geq 0$
11
14
x==-1
x++
y++
15
$\perp$
15
9
$x \geq 0$
15
18
$\perp$
18
x==-1
$y \geq 0$
18
x++
y++
12
13
19
$\perp$
19
16
$x \geq 0$
19
22
x==-1
x++
y++
23
x==-1
17
x++
20
21
24


1
$\top$
3,5,7,10,11,15,18,19
ℓ_{err}
3
$\bot$
3
x:=0;y:=0
5
$\bot$
5
ℓ_1
x==-1
x++
$x \geq 0$
3,7,11,15,19
y==-1
y++
$y \geq 0$
5,10,18
ℓ_{err}
7
$\bot$
7
ℓ_1
4
$x \geq 0$
7,11,15,19
ℓ_{err}
10
$\bot$
10
ℓ_1
$\cdots$
x==-1
y==-1
$y \geq 0$
10,18
x++
y++
ℓ_{err}
11
$\bot$
11
ℓ_1
8
$x \geq 0$
11
14
y==-1
x==-1
x++
y++
ℓ_{err}
15
$\bot$
15
ℓ_1
9
$x \geq 0$
15
18
$\bot$
18
y==-1
x==-1
$y \geq 0$
18
x++
ℓ_1
12
ℓ_1
13
ℓ_{err}
19
$\bot$
19
ℓ_1
16
$x \geq 0$
19
ℓ_{err}
22
x==-1
x++
y++
ℓ_1
20
ℓ_1
21
x++
ℓ_{err}
23
x==-1
ℓ_1
17
x++
y++
ℓ_1
24
ℓ_1
25


1
⊤
3,5,7,10,11,15,18,19
3
⊥
x:=0;y:=0
5
⊥
x==-1
x++
y==-1
x ≥ 0
3,7,11,15,19
y++
y ≥ 0
5,10,18
4
x ≥ 0
7,11,15,19
10
⊥
10
y ≥ 0
10,18
7
⊥
x==-1
y==-1
x++
y++
8
x ≥ 0
11
⊥
11
x==-1
y==-1
x++
y++
9
x ≥ 0
15
⊥
15
x==-1
y==-1
y ≥ 0
18
18
⊥
18
12
13
19
⊥
19
x==-1
16
x ≥ 0
19
x==-1
22
x++
y++
23
x==-1
17
y==-1
26
x++
y++
20
21
24
25


⊤
1
3 ,5 ,7 ,10 ,11 ,15 ,18 ,19 ,23
ℓ_{err}
3
⊥
3
x:=0;y:=0
5
⊥
5
x==-1
y==-1
ℓ_1
$x \geq 0$
3 ,7 ,11 ,15 ,19 ,23
x++
y++
$y \geq 0$
5 ,10 ,18
7
⊥
ℓ_1
$x \geq 0$
10
⊥
ℓ_1
…
$y \geq 0$
10 ,18
x++
y++
11
⊥
8
$x \geq 0$
14
15
⊥
9
$x \geq 0$
18
⊥
18
$y \geq 0$
18
12
13
19
⊥
16
$x \geq 0$
22
23
⊥
17
$x \geq 0$
26
20
21
24
25


⊤
1
3 ,5 ,7 ,10 ,11 ,15 ,18 ,19 ,23
ℓ_{err}
3
⊥
3
x:=0;y:=0
5
⊥
5
x==-1
y==-1
ℓ_1
$x \geq 0$
3 ,7 ,11 ,15 ,19 ,23
x++
y++
$y \geq 0$
5 ,10 ,18
7
⊥
ℓ_1
10
⊥
ℓ_1
…
$y \geq 0$
10 ,18
x++
y++
11
⊥
8
$x \geq 0$
14
15
⊥
9
$x \geq 0$
18
⊥
$y \geq 0$
18
12
13
19
⊥
16
$x \geq 0$
22
23
⊥
17
$x \geq 0$
26
20
21
24
25

IMPACT-Variation 3 - Vergleich

Mögliche Analogieen zwischen IMPACT-3 und Trace Abstraction:

- pro Refutation erkannte Sprachen
- Anzahl der Refutations

IMPACT-Variation 3 - Vergleich

Mögliche Analogieen zwischen IMPACT-3 und Trace Abstraction:

- pro Refutation erkannte Sprachen:
  kein offensichtlicherZusammenhang, keine Inklusion
- Anzahl der Refutations:
  nicht identisch, Unterschied möglicherweise konstanter Faktor

Fazit

- Ähnliche Verfahren, jedoch
  - Undwindings für IMPACT natürlicher
  - Automaten für Trace Abstraction natürlicher

  Unterschied in den Datenstrukturen macht Vergleich schwierig
- Hilfsmittel zur Analyse von Verfahren: (z.B. IMPACT-Varianten)
  - Unterapproximation der Sprache der Infeasible Traces
  - Unwindings mit Abbruchkriterium