

Eingabe: Anfrage und Programm,
zum Beispiel
`mult(N,M,s(0))` und:

```
add(0,X,X) .  
add(s(X),Y,s(Z)) :- add(X,Y,Z) .  
  
mult(0,_,0) .  
mult(s(X),Y,Z) :- add(U,Y,Z),mult(X,Y,U) .
```

Ausgabe: Baum, erzeugt durch folgende Schritte:

1. Erzeuge Wurzelknoten mit Anfrage, merke als noch zu bearbeiten.
2. Solange noch zu bearbeitende Knoten vorhanden:
 - wähle linken solchen Knoten
 - ermittle alle Regeln, deren Kopf mit dem linken Literal im Knoten unifizierbar ist
 - erzeuge für jede solche Regel einen (noch weiter zu bearbeitenden) Nachfolgerknoten durch Resolution
 - vermerke jeweils verwendeten Unifikator
 - sortiere Nachfolgerknoten von links nach rechts entsprechend der Reihenfolge verwendeter Regeln von oben nach unten

`?- mult(N,M,s(0)) .`



`{N/s(X),M/Y,Z/s(0)}`

`?- add(U,Y,s(0)),mult(X,Y,U) .`

noch zu bearbeiten

2. Solange noch zu bearbeitende Knoten vorhanden:

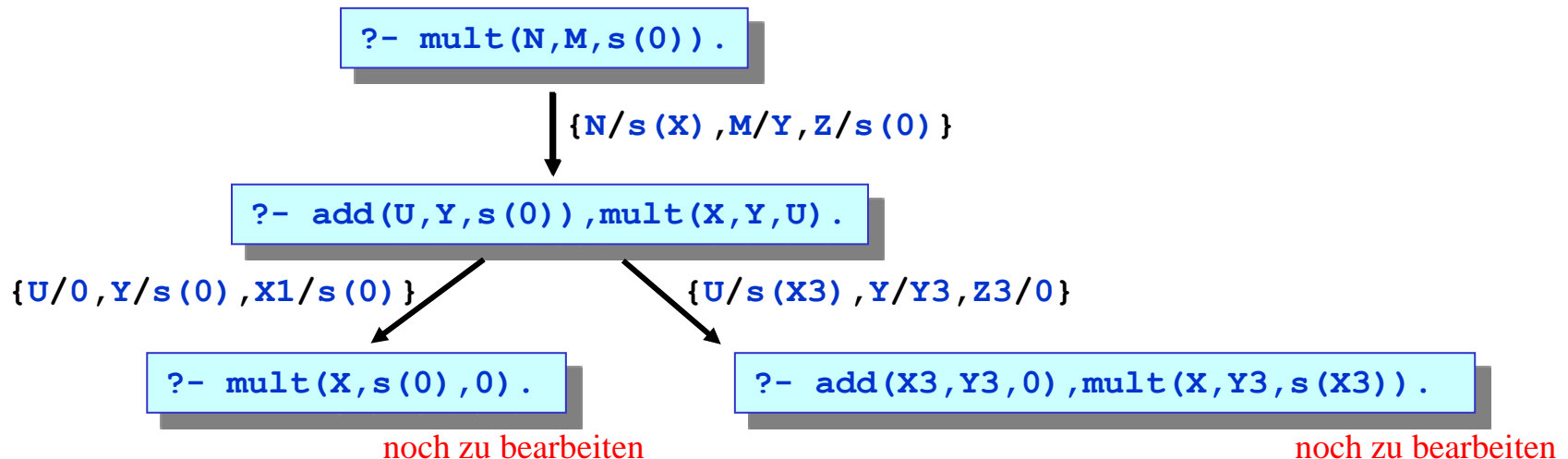
- wähle linkensten solchen Knoten
- ermittle alle Regeln, deren Kopf mit dem linkensten Literal im Knoten unifizierbar ist
- erzeuge für jede solche Regel einen (noch weiter zu bearbeitenden) Nachfolgerknoten durch Resolution
- vermerke jeweils verwendeten Unifikator
- sortiere Nachfolgerknoten von links nach rechts entsprechend der Reihenfolge verwendeter Regeln von oben nach unten

```
add(0,X,X).
```

```
add(s(X),Y,s(Z)) :- add(X,Y,Z).
```


```
mult(0,_,0).
```

```
mult(s(X),Y,Z) :- add(U,Y,Z),mult(X,Y,U).
```



Wiederholung Ableitungsbäume

2. Solange noch zu bearbeitende Knoten vorhanden:

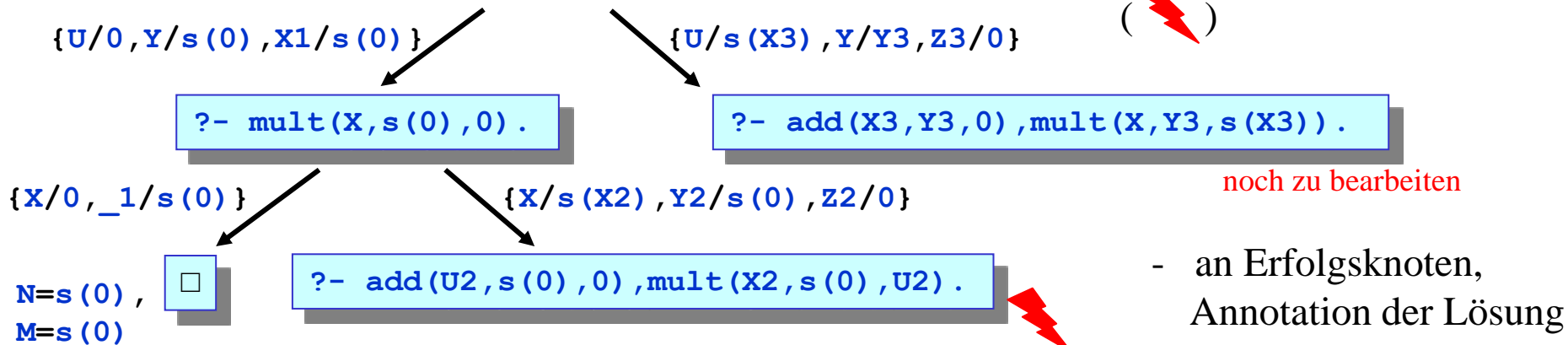
- wähle linkensten solchen Knoten
- ermittle alle Regeln, deren Kopf mit dem linkensten Literal im Knoten unifizierbar ist
- erzeuge für jede solche Regel einen (noch weiter zu bearbeitenden) Nachfolgerknoten durch Resolution
- vermerke jeweils verwendeten Unifikator
- sortiere Nachfolgerknoten von links nach rechts entsprechend der Reihenfolge verwendeter Regeln von oben nach unten
- markiere Knoten als nicht weiter zu bearbeiten, wenn leer () oder linkstes Literal mit keinem Regelkopf unifizierbar

```
add(0,X,X).
```

```
add(s(X),Y,s(Z)) :- add(X,Y,Z).
```

```
mult(0,_,0).
```

```
mult(s(X),Y,Z) :- add(U,Y,Z),mult(X,Y,U).
```



2. Solange noch zu bearbeitende Knoten vorhanden:

- wähle linkensten solchen Knoten
- ermittle alle Regeln, deren Kopf mit dem linkensten Literal im Knoten unifizierbar ist
- erzeuge für jede solche Regel einen (noch weiter zu bearbeitenden) Nachfolgerknoten durch Resolution
- vermerke jeweils verwendeten Unifikator
- sortiere Nachfolgerknoten von links nach rechts entsprechend der Reihenfolge verwendeter Regeln von oben nach unten
- markiere Knoten als nicht weiter zu bearbeiten, wenn leer oder linkstes Literal mit keinem Regelkopf unifizierbar
- an Erfolgsknoten, Annotation der Lösung (Komposition der Unifikatoren, angewandt auf relevante Variablen)

```
add(0,X,X).
```

```
add(s(X),Y,s(Z)) :- add(X,Y,Z).
```

```
mult(0,_,0).
```

```
mult(s(X),Y,Z) :- add(U,Y,Z),mult(X,Y,U).
```

```
?- mult(X,s(0),0).
```

```
?- add(X3,Y3,0),mult(X,Y3,s(X3)).
```

$\{X/0, _1/s(0)\}$

$\{X/s(X2), Y2/s(0), Z2/0\}$

$\{X3/0, Y3/0, X4/0\}$

$N=s(0),$
 $M=s(0)$



```
?- add(U2,s(0),0),mult(X2,s(0),U2).
```



```
?- mult(X,0,s(0)).
```

noch zu bearbeiten

Wiederholung Ableitungsbäume

bisher insgesamt:

`?- mult(N,M,s(0)).`

$\{N/s(X), M/Y, Z/s(0)\}$

`?- add(U,Y,s(0)),mult(X,Y,U).`

$\{U/0, Y/s(0), X1/s(0)\}$

$\{U/s(X3), Y/Y3, Z3/0\}$

`?- mult(X,s(0),0).`

`?- add(X3,Y3,0),mult(X,Y3,s(X3)).`

$\{X/0, _1/s(0)\}$

$\{X/s(X2), Y2/s(0), Z2/0\}$

$\{X3/0, Y3/0, X4/0\}$



`?- add(U2,s(0),0),mult(X2,s(0),U2).`

`?- mult(X,0,s(0)).`

$N=s(0),$
 $M=s(0)$

noch zu bearbeiten

weiter:

- wähle linkensten noch zu bearbeitenden Knoten
- ermittle alle Regeln, deren Kopf mit dem linkensten Literal im Knoten unifizierbar ist
- erzeuge für jede solche Regel einen Nachfolgerknoten
- ...

`add(0,X,X).`

`add(s(X),Y,s(Z)) :- add(X,Y,Z).`

`mult(0,_,0).`

`mult(s(X),Y,Z) :- add(U,Y,Z),mult(X,Y,U).`

Wiederholung Ableitungsbäume

```
add(0,X,X) .  
add(s(X),Y,s(Z)) :- add(X,Y,Z) .  
  
mult(0,_,0) .  
mult(s(X),Y,Z) :- add(U,Y,Z),mult(X,Y,U) .
```

↙
?- add(X3,Y3,0),mult(X,Y3,s(X3)) .

↓ {X3/0,Y3/0,X4/0}

?- mult(X,0,s(0)) .

↓ {X/s(X5),Y5/0,Z5/s(0)}

?- add(U5,0,s(0)),mult(X5,0,U5) .

↓ {U5/s(X6),Y6/0,Z6/0}

?- add(X6,0,0),mult(X5,0,s(X6)) .

↓ {X6/0,X7/0}

⋮