

Inferenzmethoden

Teil III

Behandlung spezifischer Fragestellungen

Spezialisierte Beweistechniken



1. Verarbeitung mathematischer Theorien
2. Gleichheitsbehandlung
3. Termersetzung und -auswertung
4. Zahlen und Induktion

STEIGERUNG DER EFFIZIENZ VON BEWEISVERFAHREN

- **Reale Problemstellungen liefern komplexe Formeln**
 - Formeln enthalten große Mengen von Fakten und Zusammenhängen
 - Formeln enthalten Bezüge zu verschiedenen mathematischen Theorien
 - Zahlen, Gleichheit, Relationen, Funktionen mit “Bedeutung” (Wert)

STEIGERUNG DER EFFIZIENZ VON BEWEISVERFAHREN

- **Reale Problemstellungen liefern komplexe Formeln**
 - Formeln enthalten große Mengen von Fakten und Zusammenhängen
 - Formeln enthalten Bezüge zu verschiedenen mathematischen Theorien
 - Zahlen, Gleichheit, Relationen, Funktionen mit “Bedeutung” (Wert)
 - **Wie kann man derartige Formeln effizient genug verarbeiten?**

STEIGERUNG DER EFFIZIENZ VON BEWEISVERFAHREN

- **Reale Problemstellungen liefern komplexe Formeln**
 - Formeln enthalten große Mengen von Fakten und Zusammenhängen
 - Formeln enthalten Bezüge zu verschiedenen mathematischen Theorien
 - Zahlen, Gleichheit, Relationen, Funktionen mit “Bedeutung” (Wert)
 - **Wie kann man derartige Formeln effizient genug verarbeiten?**
- **Verdichtung allgemeiner Beweisverfahren**
 - Spezialanalyse liefert gleiche Informationen wie explizite Beweisschritte
 - Kompakte Verarbeitung von Klauselkopien durch **Faktorisierung**
 - Verdichtung wiederholter Argumenteketten durch **Zyklenanalyse**
 - **Datenbanktechniken** und **Indizierung** für große Faktenmengen

STEIGERUNG DER EFFIZIENZ VON BEWEISVERFAHREN

- **Reale Problemstellungen liefern komplexe Formeln**
 - Formeln enthalten große Mengen von Fakten und Zusammenhängen
 - Formeln enthalten Bezüge zu verschiedenen mathematischen Theorien
 - Zahlen, Gleichheit, Relationen, Funktionen mit “Bedeutung” (Wert)
 - **Wie kann man derartige Formeln effizient genug verarbeiten?**
- **Verdichtung allgemeiner Beweisverfahren**
 - Spezialanalyse liefert gleiche Informationen wie explizite Beweisschritte
 - Kompakte Verarbeitung von Klauselkopien durch **Faktorisierung**
 - Verdichtung wiederholter Argumenteketten durch **Zyklenanalyse**
 - **Datenbanktechniken** und **Indizierung** für große Faktenmengen
 - **Verfahren sind sehr technisch und liefern wenig methodische Erkenntnisse**

STEIGERUNG DER EFFIZIENZ VON BEWEISVERFAHREN

- **Reale Problemstellungen liefern komplexe Formeln**
 - Formeln enthalten große Mengen von Fakten und Zusammenhängen
 - Formeln enthalten Bezüge zu verschiedenen mathematischen Theorien
 - Zahlen, Gleichheit, Relationen, Funktionen mit “Bedeutung” (Wert)
 - **Wie kann man derartige Formeln effizient genug verarbeiten?**
- **Verdichtung allgemeiner Beweisverfahren**
 - Spezialanalyse liefert gleiche Informationen wie explizite Beweisschritte
 - Kompakte Verarbeitung von Klauselkopien durch **Faktorisierung**
 - Verdichtung wiederholter Argumenteketten durch **Zyklenanalyse**
 - **Datenbanktechniken** und **Indizierung** für große Faktenmengen
 - **Verfahren sind sehr technisch und liefern wenig methodische Erkenntnisse**
- **Theoriespezifische Spezialverfahren**
 - **Entscheidungsprozeduren** für Gleichheit, Arithmetik, etc.
 - **Rewrite-Verfahren** zur Auswertung von Termen
 - **Präfixbehandlung** für Verarbeitung nichtklassischer Logiken

STEIGERUNG DER EFFIZIENZ VON BEWEISVERFAHREN

- **Reale Problemstellungen liefern komplexe Formeln**
 - Formeln enthalten große Mengen von Fakten und Zusammenhängen
 - Formeln enthalten Bezüge zu verschiedenen mathematischen Theorien
 - Zahlen, Gleichheit, Relationen, Funktionen mit “Bedeutung” (Wert)
 - **Wie kann man derartige Formeln effizient genug verarbeiten?**
- **Verdichtung allgemeiner Beweisverfahren**
 - Spezialanalyse liefert gleiche Informationen wie explizite Beweisschritte
 - Kompakte Verarbeitung von Klauselkopien durch **Faktorisierung**
 - Verdichtung wiederholter Argumenteketten durch **Zyklenanalyse**
 - **Datenbanktechniken** und **Indizierung** für große Faktenmengen
 - **Verfahren sind sehr technisch und liefern wenig methodische Erkenntnisse**
- **Theoriespezifische Spezialverfahren**
 - **Entscheidungsprozeduren** für Gleichheit, Arithmetik, etc.
 - **Rewrite-Verfahren** zur Auswertung von Termen
 - **Präfixbehandlung** für Verarbeitung nichtklassischer Logiken
 - **Verfahren können in den Unifikationsalgorithmus integriert werden**

Inferenzmethoden

Einheit 10

Theorie- und Gleichheitsbehandlung



1. Theoriekonnectionen & Unifikationstheorie
2. Axiomatische Gleichheitsbehandlung
3. Gleichheitskonnectionen
4. Resolution und Gleichheit

THEORIEBEHANDLUNG

Separate Verarbeitung spezieller Inferenzen

Separate Verarbeitung spezieller Inferenzen

- **Theorien sind gekennzeichnet durch Axiome**
 - z.B. Reflexivität, Symmetrie, Transitivität, Substitutivität für Gleichheit
Assoziativität, Identität, Inverse für Gruppen
Peano-Axiome für natürliche Zahlen
 - Axiome sind die “Grundwahrheiten” der Theorie, aus denen alles folgt

Separate Verarbeitung spezieller Inferenzen

- **Theorien sind gekennzeichnet durch Axiome**
 - z.B. Reflexivität, Symmetrie, Transitivität, Substitutivität für Gleichheit
Assoziativität, Identität, Inverse für Gruppen
Peano-Axiome für natürliche Zahlen
 - Axiome sind die “Grundwahrheiten” der Theorie, aus denen alles folgt
- **Viele Theorien benutzen spezielle Inferenzketten**
 - Standardargumente um Axiome effizient in Schlußfolgerungen einzusetzen
 - z.B. gezieltes Einsetzen von Substitutivität beim Gleichheitsschließen

Separate Verarbeitung spezieller Inferenzen

- **Theorien sind gekennzeichnet durch Axiome**
 - z.B. Reflexivität, Symmetrie, Transitivität, Substitutivität für Gleichheit
Assoziativität, Identität, Inverse für Gruppen
Peano-Axiome für natürliche Zahlen
 - Axiome sind die “Grundwahrheiten” der Theorie, aus denen alles folgt
- **Viele Theorien benutzen spezielle Inferenzketten**
 - Standardargumente um Axiome effizient in Schlußfolgerungen einzusetzen
 - z.B. gezieltes Einsetzen von Substitutivität beim Gleichheitsschließen
- **Allgemeine Beweiser unterstützen dies nicht**
 - Theoriespezifische Inferenzen passen nicht zum allgemeinen Verfahren
 - Beweiser müssen die Axiome als zusätzliche Klauseln hinzunehmen

EINBETTUNG VON THEORIEN IN BEWEISVERFAHREN

Beweise $Pa \wedge a=b \Rightarrow Pb$ aus Gleichheitsaxiomen

EINBETTUNG VON THEORIEN IN BEWEISVERFAHREN

Beweise $Pa \wedge a=b \Rightarrow Pb$ aus Gleichheitsaxiomen

● Konventioneller Matrixbeweis mit Axiomen

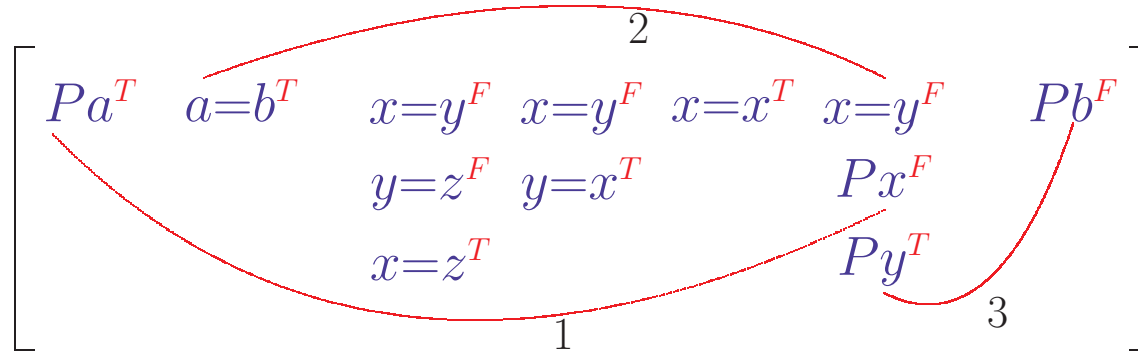
$$\left[\begin{array}{cccccc} Pa^T & a=b^T & x=y^F & x=y^F & x=x^T & x=y^F & Pb^F \\ & & y=z^F & y=x^T & & Px^F & \\ & & x=z^T & & & Py^T & \end{array} \right]$$

- Gleichheitsaxiome sind in die Matrix integriert
- Viele Schritte bei der Beweissuche nötig

EINBETTUNG VON THEORIEN IN BEWEISVERFAHREN

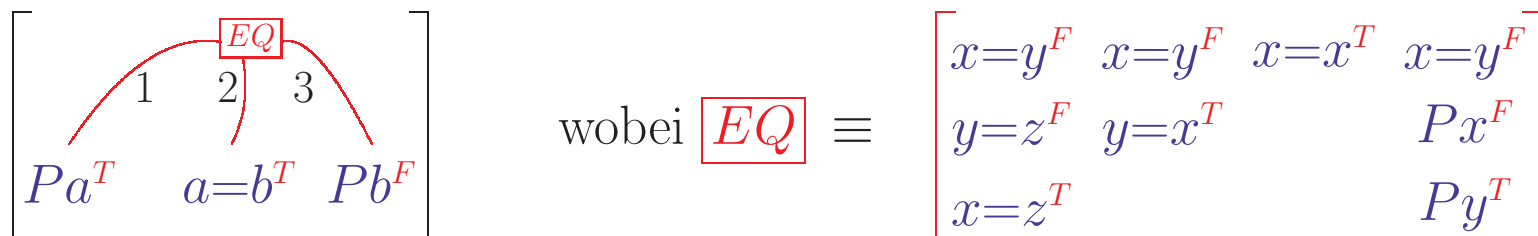
Beweise $Pa \wedge a=b \Rightarrow Pb$ aus Gleichheitsaxiomen

● Konventioneller Matrixbeweis mit Axiomen



- Gleichheitsaxiome sind in die Matrix integriert
- Viele Schritte bei der Beweissuche nötig

● Integriere Theorie durch spezielle Konnektionen



- Theoriekonnektion beinhaltet Inferenzschritte der Theorie EQ

THEORIEKONNEKTIONEN

- **Erweiterter Komplementaritätsbegriff für Theorien**
 - $P t_1^T$ und $P t_2^F$ sind **komplementär in der Theorie \mathcal{T}** , wenn $\sigma(t_1)$ und $\sigma(t_2)$ in \mathcal{T} gleich sind (σ zulässige Substitution)
 - Erlaubt Verarbeitung theoriespezifischer Inferenzketten

THEORIEKONNEKTIONEN

- **Erweiterter Komplementaritätsbegriff für Theorien**
 - $P t_1^T$ und $P t_2^F$ sind **komplementär in der Theorie \mathcal{T}** , wenn $\sigma(t_1)$ und $\sigma(t_2)$ in \mathcal{T} gleich sind (σ zulässige Substitution)
 - Erlaubt Verarbeitung theoriespezifischer Inferenzketten
- **Unifikation ist mehr als syntaktisches Gleichmachen**

$$\left[\begin{array}{c} P((x+x)-1)^T \\ P(1)^F \end{array} \right] \sigma = [1/x]$$

- Konnektion benutzt Unifikation für einfache Arithmetik

THEORIEKONNEKTIONEN

- **Erweiterter Komplementaritätsbegriff für Theorien**

- $P t_1^T$ und $P t_2^F$ sind **komplementär in der Theorie \mathcal{T}** , wenn $\sigma(t_1)$ und $\sigma(t_2)$ in \mathcal{T} gleich sind (σ zulässige Substitution)
- Erlaubt Verarbeitung theoriespezifischer Inferenzketten

- **Unifikation ist mehr als syntaktisches Gleichmachen**

$$\left[\begin{array}{c} \text{Arith} \\ P((x+x)-1)^T \end{array} \right] \xrightarrow{P(1)^F} \left[\begin{array}{c} P(1)^F \\ \sigma = [1/x] \end{array} \right]$$

- Konnektion benutzt Unifikation für einfache Arithmetik

$$\left[\begin{array}{c} \text{Group} \\ R(c, b)^T \end{array} \right] \xrightarrow{R(z \cdot (\bar{c} \cdot c), z \cdot (\bar{z} \cdot b))^F} \left[\begin{array}{c} R(z \cdot (\bar{c} \cdot c), z \cdot (\bar{z} \cdot b))^F \\ \sigma = [c/z] \end{array} \right]$$

- Konnektion benutzt Unifikation für Gruppentheorie

THEORIEKONNEKTIONEN

- **Erweiterter Komplementaritätsbegriff für Theorien**

- $P t_1^T$ und $P t_2^F$ sind **komplementär in der Theorie \mathcal{T}** , wenn $\sigma(t_1)$ und $\sigma(t_2)$ in \mathcal{T} gleich sind (σ zulässige Substitution)
- Erlaubt Verarbeitung theoriespezifischer Inferenzketten

- **Unifikation ist mehr als syntaktisches Gleichmachen**

$$\left[\begin{array}{c} \text{Arith} \\ P((x+x)-1)^T \end{array} \right] \xrightarrow{P(1)^F} \left[\begin{array}{c} P(1)^F \\ \sigma = [1/x] \end{array} \right]$$

- Konnektion benutzt Unifikation für einfache Arithmetik

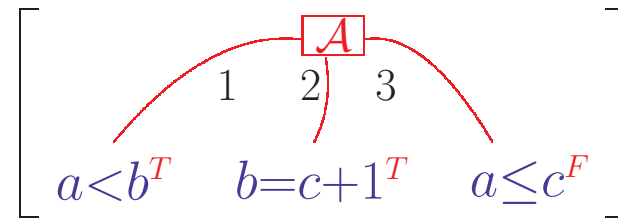
$$\left[\begin{array}{c} \text{Group} \\ R(c, b)^T \end{array} \right] \xrightarrow{R(z \cdot (\bar{c} \cdot c), z \cdot (\bar{z} \cdot b))^F} \left[\begin{array}{c} R(z \cdot (\bar{c} \cdot c), z \cdot (\bar{z} \cdot b))^F \\ \sigma = [c/z] \end{array} \right]$$

- Konnektion benutzt Unifikation für Gruppentheorie

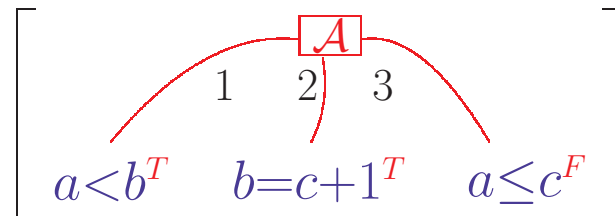
- **Allgemeiner Mechanismus noch unerforscht**

- Meist Integration von **Theorie-Unifikation** in konventionellen Beweiser

UNIFIKATIONSTHEORIE

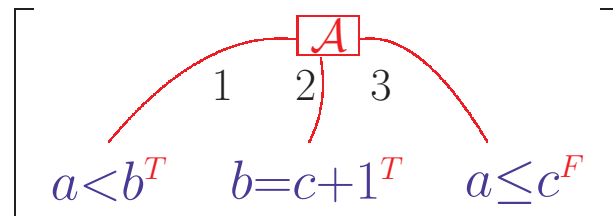


UNIFIKATIONSTHEORIE



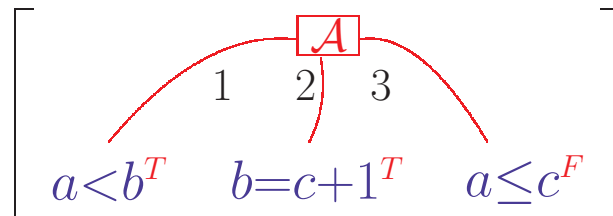
- $\{a < b^T, b = c + 1^T, a \leq c^F\}$ komplementär in Theorie \mathcal{A}
 - Konnektionen können auch unär oder ternär sein
 - Konnektionen auch zwischen verschiedenartigen Literalen möglich
z.B. bei Verwendung der Axiome $x < y \Rightarrow x + 1 \leq y$ und $x + 1 \leq y + 1 \Rightarrow x \leq y$
 - Unifikationsmechanismus muß ganze Literale wie Terme behandeln

UNIFIKATIONSTHEORIE



- $\{a < b^T, b = c + 1^T, a \leq c^F\}$ komplementär in Theorie \mathcal{A}
 - Konnektionen können auch unär oder ternär sein
 - Konnektionen auch zwischen verschiedenartigen Literalen möglich
z.B. bei Verwendung der Axiome $x < y \Rightarrow x + 1 \leq y$ und $x + 1 \leq y + 1 \Rightarrow x \leq y$
 - Unifikationsmechanismus muß ganze Literale wie Terme behandeln
- **Es gibt noch viele offene Fragen**
 - Wie genau Unifizierbarkeit modulo Theorie \mathcal{T} definieren?
 - Gibt es mgu's und, wenn ja, wieviele?
 - Gibt es Unifikationsalgorithmen (als Entscheidungsprozeduren)?
 - Was ist die Komplexität des Unifikationsverfahrens?

UNIFIKATIONSTHEORIE



- $\{a < b^T, b = c + 1^T, a \leq c^F\}$ komplementär in Theorie \mathcal{A}
 - Konnektionen können auch unär oder ternär sein
 - Konnektionen auch zwischen verschiedenartigen Literalen möglich
z.B. bei Verwendung der Axiome $x < y \Rightarrow x + 1 \leq y$ und $x + 1 \leq y + 1 \Rightarrow x \leq y$
 - Unifikationsmechanismus muß ganze Literale wie Terme behandeln
- **Es gibt noch viele offene Fragen**
 - Wie genau Unifizierbarkeit modulo Theorie \mathcal{T} definieren?
 - Gibt es mgu's und, wenn ja, wieviele?
 - Gibt es Unifikationsalgorithmen (als Entscheidungsprozeduren)?
 - Was ist die Komplexität des Unifikationsverfahrens?
- **Bisher gibt es nur wenig allgemeine Lösungen**
 - Erfolgreich nur für spezielle Theorien (Gleichheit, Gruppen, ...)

TYPEN VON UNIFIKATIONSTHEORIEN

- **unitäre:**

- Es gibt (bei Unifizierbarkeit) **genau einen** allgemeinsten Unifikator
- z.B. Standard-Unifikation

TYPEN VON UNIFIKATIONSTHEORIEN

- **unitäre:**

- Es gibt (bei Unifizierbarkeit) **genau einen** allgemeinsten Unifikator
- z.B. Standard-Unifikation

- **finitäre:**

- Es gibt bei **endlich viele** allgemeinste Unifikatoren
- z.B. AC-Unifikation, Präfix-Unifikation

TYPEN VON UNIFIKATIONSTHEORIEN

- **unitäre:**

- Es gibt (bei Unifizierbarkeit) **genau einen** allgemeinsten Unifikator
- z.B. Standard-Unifikation

- **finitäre:**

- Es gibt bei **endlich viele** allgemeinste Unifikatoren
- z.B. AC-Unifikation, Präfix-Unifikation

- **infinite:**

- Es gibt **unendlich viele** allgemeinste Unifikatoren
- z.B. Unifikation modulo Assoziativität

TYPEN VON UNIFIKATIONSTHEORIEN

- **unitäre:**

- Es gibt (bei Unifizierbarkeit) **genau einen** allgemeinsten Unifikator
- z.B. Standard-Unifikation

- **finitäre:**

- Es gibt bei **endlich viele** allgemeinste Unifikatoren
- z.B. AC-Unifikation, Präfix-Unifikation

- **infinite:**

- Es gibt **unendlich viele** allgemeinste Unifikatoren
- z.B. Unifikation modulo Assoziativität

- **leere:**

- Es gibt i.a. **keine** allgemeinsten Unifikatoren
- Wenig erwünscht

DIE THEORIE DER GLEICHHEIT

- **Wichtigste Grundbeziehung zwischen Objekten**
 - Spezialbehandlung sehr lohnenswert
 - Dargestellt als zweistelliges Prädikat zwischen Termen
 - Prädikatszeichen \doteq in Infix-Notation

DIE THEORIE DER GLEICHHEIT

- **Wichtigste Grundbeziehung zwischen Objekten**

- Spezialbehandlung sehr lohnenswert
- Dargestellt als zweistelliges Prädikat zwischen Termen
- Prädikatszeichen \doteq in Infix-Notation

- **Charakterisiert durch 5 Grundeigenschaften**

- $x \doteq x$ Reflexivität
- $x \doteq y \Rightarrow y \doteq x$ Symmetrie
- $x \doteq y \wedge y \doteq z \Rightarrow x \doteq z$ Transitivität
- $x_i \doteq y \Rightarrow f(x_1, \dots, x_i, \dots, x_n) \doteq f(x_1, \dots, y, \dots, x_n)$
Substitutivität auf Funktionen (Schema)
- $x_i \doteq y \Rightarrow [P(x_1, \dots, x_i, \dots, x_n) \Leftrightarrow P(x_1, \dots, y, \dots, x_n)]$
Substitutivität auf Prädikaten (Schema)

DIE THEORIE DER GLEICHHEIT

- **Wichtigste Grundbeziehung zwischen Objekten**

- Spezialbehandlung sehr lohnenswert
- Dargestellt als zweistelliges Prädikat zwischen Termen
- Prädikatszeichen \doteq in Infix-Notation

- **Charakterisiert durch 5 Grundeigenschaften**

- $x \doteq x$ Reflexivität
- $x \doteq y \Rightarrow y \doteq x$ Symmetrie
- $x \doteq y \wedge y \doteq z \Rightarrow x \doteq z$ Transitivität
- $x_i \doteq y \Rightarrow f(x_1, \dots, x_i, \dots, x_n) \doteq f(x_1, \dots, y, \dots, x_n)$
Substitutivität auf Funktionen (Schema)
- $x_i \doteq y \Rightarrow [P(x_1, \dots, x_i, \dots, x_n) \Leftrightarrow P(x_1, \dots, y, \dots, x_n)]$
Substitutivität auf Prädikaten (Schema)

- **Symmetrie und Transitivität sind ableitbar**

AXIOMATISCHE GLEICHHEITSBEHANDLUNG

Gleichheitsbeweise ohne Verdichtung

- **Erweitere Formel um Axiome der Gleichheit**

Gleichheitsbeweise ohne Verdichtung

- **Erweitere Formel um Axiome der Gleichheit**
- **Verwende minimale Axiomenmenge**
 - $x \doteq x$
 - $x_i \doteq y \Rightarrow f(x_1, \dots, x_i, \dots, x_n) \doteq f(x_1, \dots, y, \dots, x_n)$
 - $x_i \doteq y \Rightarrow [P(x_1, \dots, x_i, \dots, x_n) \Rightarrow P(x_1, \dots, y, \dots, x_n)]$

Gleichheitsbeweisen ohne Verdichtung

- **Erweitere Formel um Axiome der Gleichheit**
- **Verwende minimale Axiomenmenge**
 - $x \doteq x$
 - $x_i \doteq y \Rightarrow f(x_1, \dots, x_i, \dots, x_n) \doteq f(x_1, \dots, y, \dots, x_n)$
 - $x_i \doteq y \Rightarrow [P(x_1, \dots, x_i, \dots, x_n) \Rightarrow P(x_1, \dots, y, \dots, x_n)]$

Das Schema der Substitutivität muß für jedes vorkommende Funktions- und Prädikatssymbol instantiiert werden

Gleichheitsbeweise ohne Verdichtung

- **Erweitere Formel um Axiome der Gleichheit**
- **Verwende minimale Axiomenmenge**
 - $x \doteq x$
 - $x_i \doteq y \Rightarrow f(x_1, \dots, x_i, \dots, x_n) \doteq f(x_1, \dots, y, \dots, x_n)$
 - $x_i \doteq y \Rightarrow [P(x_1, \dots, x_i, \dots, x_n) \Rightarrow P(x_1, \dots, y, \dots, x_n)]$

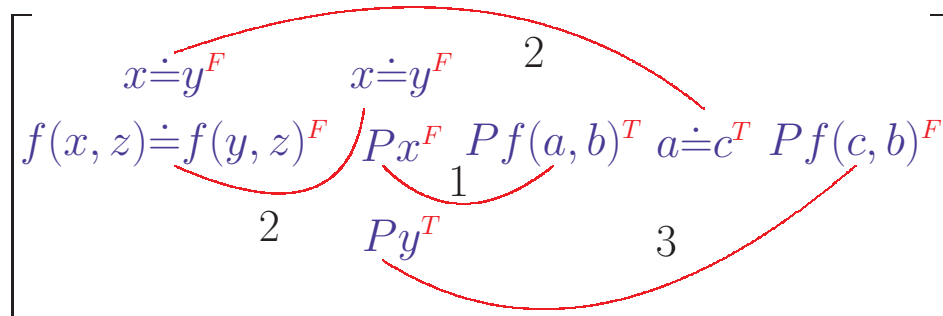
Das Schema der Substitutivität muß für jedes vorkommende Funktions- und Prädikatssymbol instantiiert werden

- **Erhebliche Vergrößerung des Suchraums**
 - Unbrauchbar für komplexe Formeln

GLEICHHEITSKONNEKTIONEN

- **Axiomatische Gleichheitsbehandlung ist aufwendig**

- Einfache Beweise wie $Pf(a, b) \wedge a \doteq c \Rightarrow Pf(c, b)$ werden umständlich

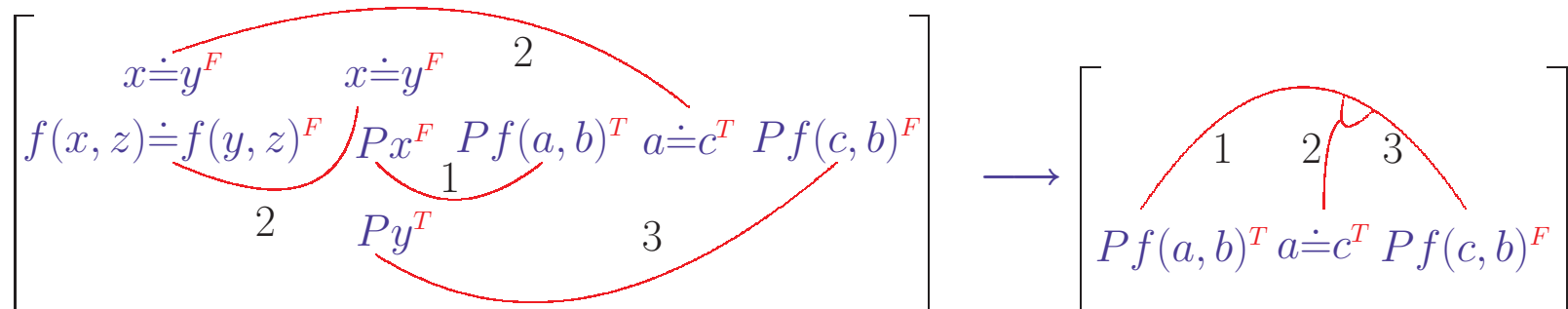


- Menschen gehen direkter mit Gleichheiten um

GLEICHHEITSKONNEKTIONEN

- **Axiomatische Gleichheitsbehandlung ist aufwendig**

- Einfache Beweise wie $Pf(a, b) \wedge a \doteq c \Rightarrow Pf(c, b)$ werden umständlich



- Menschen gehen direkter mit Gleichheiten um

- **Verdichte Beweisführung durch eq-Konnektion**

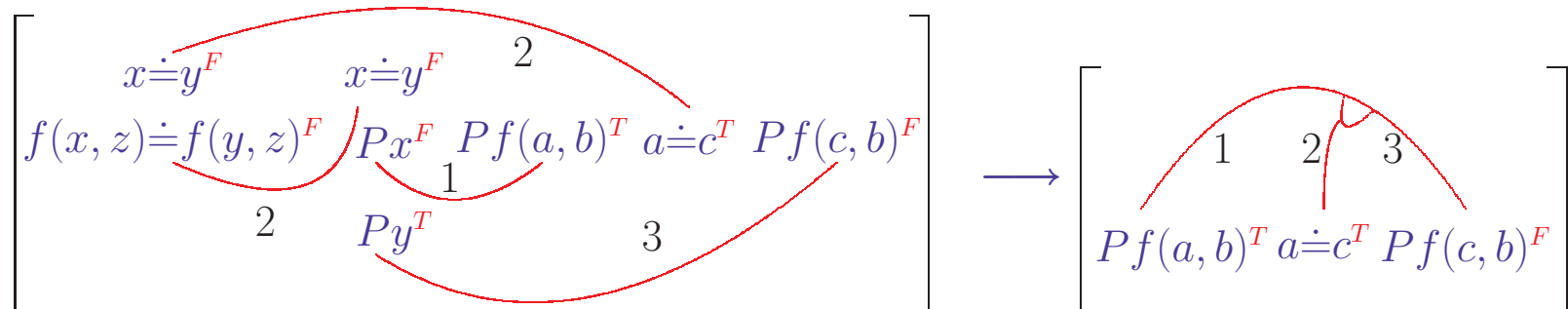
- Konnektion verbindet Literalpaar und ein oder mehrere Gleichungen

- Unifikation darf konnektierte Gleichheiten berücksichtigen

GLEICHHEITSKONNEKTIONEN

- **Axiomatische Gleichheitsbehandlung ist aufwendig**

- Einfache Beweise wie $Pf(a, b) \wedge a \doteq c \Rightarrow Pf(c, b)$ werden umständlich



- Menschen gehen direkter mit Gleichheiten um

- **Verdichte Beweisführung durch eq-Konnektion**

- Konnektion verbindet Literalpaar und ein oder mehrere Gleichungen

- Unifikation darf konnektierte Gleichheiten berücksichtigen

- **Strategische Steuerung wird aufwendiger**

- Welche Gleichheiten sind geeignet? (i.a. **unentscheidbares Problem**)

- Sehr kompliziert, wenn gleichzeitig Substitutionen zu bestimmen sind

ENTSCHEIDUNGSPROZEDUR FÜR EINFACHE GLEICHHEIT

Folgt eine Gleichheit aus anderen Gleichheiten?

ENTSCHEIDUNGSPROZEDUR FÜR EINFACHE GLEICHHEIT

Folgt eine Gleichheit aus anderen Gleichheiten?

- **Wichtig für praktische Beweisführung**

- z.B.: $f(f(a, b), b) \doteq a$ folgt aus $f(a, b) \doteq a$

- $g(a) \doteq a$ folgt aus $g(g(g(a))) \doteq a$ und $g(g(g(g(g(a)))) \doteq a$

ENTSCHEIDUNGSPROZEDUR FÜR EINFACHE GLEICHHEIT

Folgt eine Gleichheit aus anderen Gleichheiten?

- **Wichtig für praktische Beweisführung**

- z.B.: $f(f(a, b), b) \doteq a$ folgt aus $f(a, b) \doteq a$

- $g(a) \doteq a$ folgt aus $g(g(g(a))) \doteq a$ und $g(g(g(g(g(a)))) \doteq a$

- Intuitiver Beweis (gezieltes Einsetzen) einfach

ENTSCHEIDUNGSPROZEDUR FÜR EINFACHE GLEICHHEIT

Folgt eine Gleichheit aus anderen Gleichheiten?

- **Wichtig für praktische Beweisführung**

- z.B.: $f(f(a, b), b) \doteq a$ folgt aus $f(a, b) \doteq a$
 $g(a) \doteq a$ folgt aus $g(g(g(a))) \doteq a$ und $g(g(g(g(g(a)))))) \doteq a$
- Intuitiver Beweis (gezieltes Einsetzen) einfach

- **Quantorenfreie Gleichheit ist entscheidbar**

- Einfache Theorie: Gleichheiten mit uninterpretierten Symbolen
- Semantik: Reflexivität, Symmetrie, Transitivität, Substitution

ENTSCHEIDUNGSPROZEDUR FÜR EINFACHE GLEICHHEIT

Folgt eine Gleichheit aus anderen Gleichheiten?

- **Wichtig für praktische Beweisführung**
 - z.B.: $f(f(a, b), b) \doteq a$ folgt aus $f(a, b) \doteq a$
 $g(a) \doteq a$ folgt aus $g(g(g(a))) \doteq a$ und $g(g(g(g(g(a)))))) \doteq a$
 - Intuitiver Beweis (gezieltes Einsetzen) einfach
- **Quantorenfreie Gleichheit ist entscheidbar**
 - Einfache Theorie: Gleichheiten mit uninterpretierten Symbolen
 - Semantik: Reflexivität, Symmetrie, Transitivität, Substitution
- **Effiziente Verfahren verfügbar**
 - Berechnung der transitiven Hülle einer Äquivalenzrelation
 - Technisch: Kongruenzabschluß des Relationsgraphen

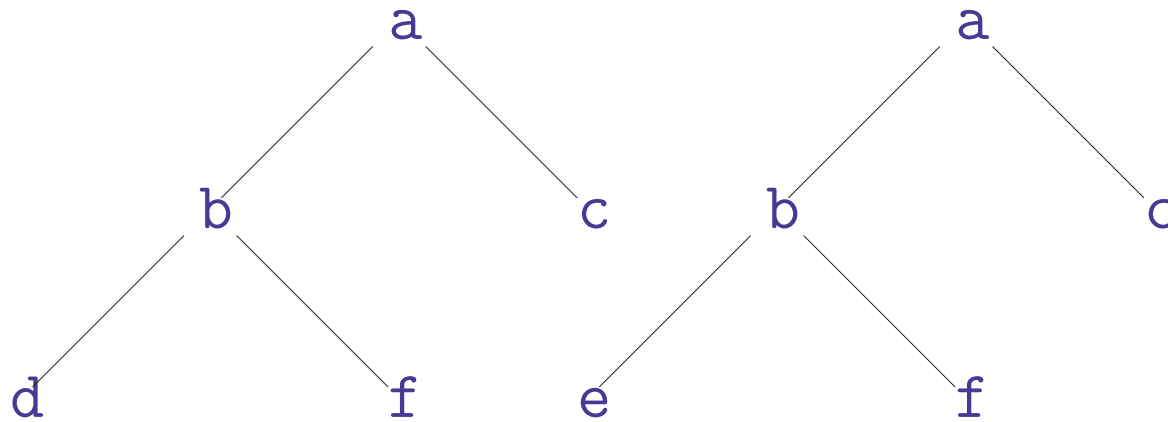
ENTSCHEIDUNGSPROZEDUR FÜR EINFACHE GLEICHHEIT

Folgt eine Gleichheit aus anderen Gleichheiten?

- **Wichtig für praktische Beweisführung**
 - z.B.: $f(f(a, b), b) \doteq a$ folgt aus $f(a, b) \doteq a$
 $g(a) \doteq a$ folgt aus $g(g(g(a))) \doteq a$ und $g(g(g(g(g(a)))))) \doteq a$
 - Intuitiver Beweis (gezieltes Einsetzen) einfach
- **Quantorenfreie Gleichheit ist entscheidbar**
 - Einfache Theorie: Gleichheiten mit uninterpretierten Symbolen
 - Semantik: Reflexivität, Symmetrie, Transitivität, Substitution
- **Effiziente Verfahren verfügbar**
 - Berechnung der transitiven Hülle einer Äquivalenzrelation
 - Technisch: Kongruenzabschluß des Relationsgraphen
- **Entscheidungsprozedur ist keine Unifikation**
 - Verfahren überprüft Gleichheiten, aber instantiiert keine Variablen

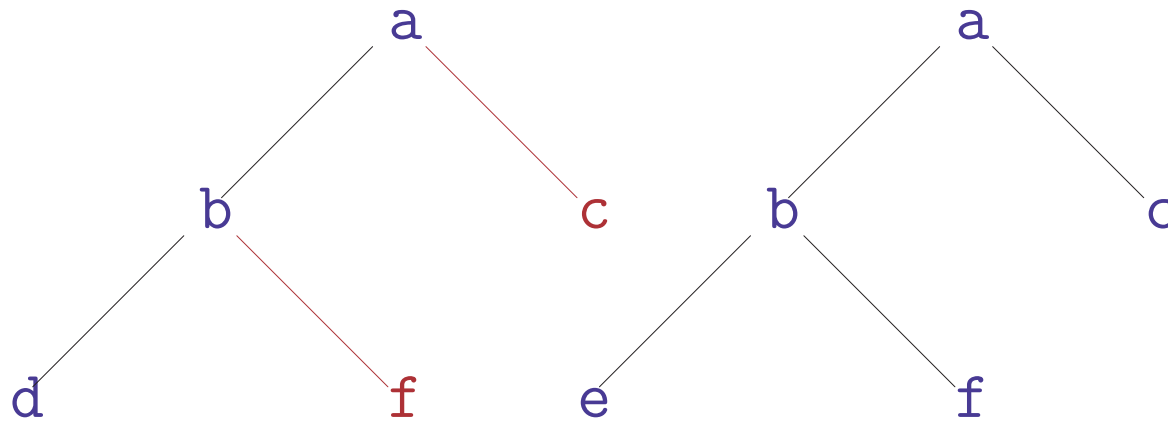
GLEICHHEITSSCHLIESSEN AM BEISPIEL

Zeige: $a(b(d,f),c) \doteq a(b(e,f),c)$ folgt aus $d \doteq e$



GLEICHHEITSSCHLIESSEN AM BEISPIEL

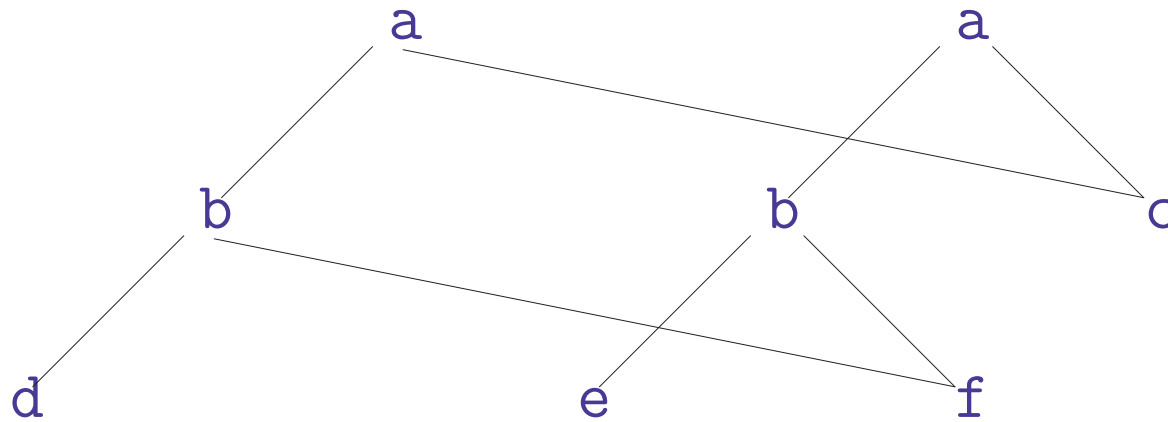
Zeige: $a(b(d,f),c) \doteq a(b(e,f),c)$ folgt aus $d \doteq e$



1. Verschmelze identische Knoten

GLEICHHEITSSCHLIESSEN AM BEISPIEL

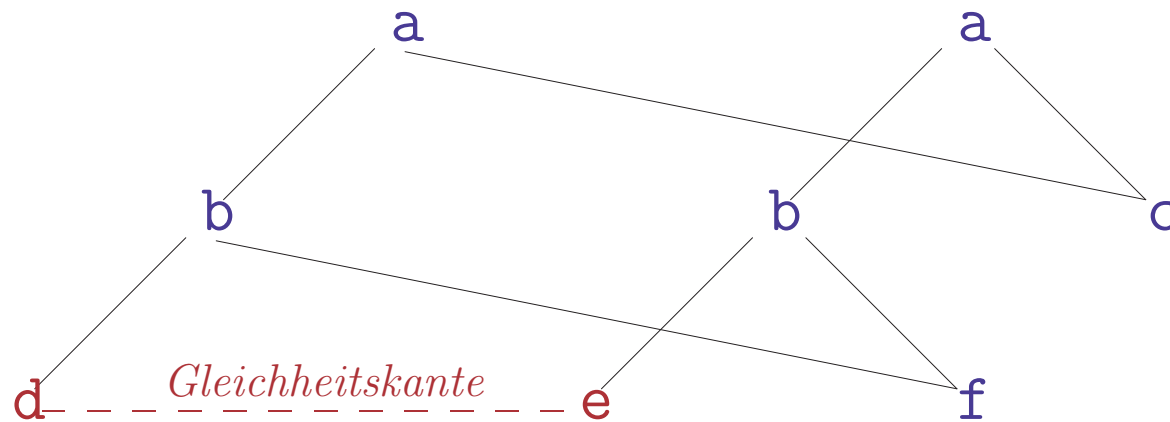
Zeige: $a(b(d,f),c) \doteq a(b(e,f),c)$ folgt aus $d \doteq e$



1. Verschmelze identische Knoten

GLEICHHEITSSCHLIESSEN AM BEISPIEL

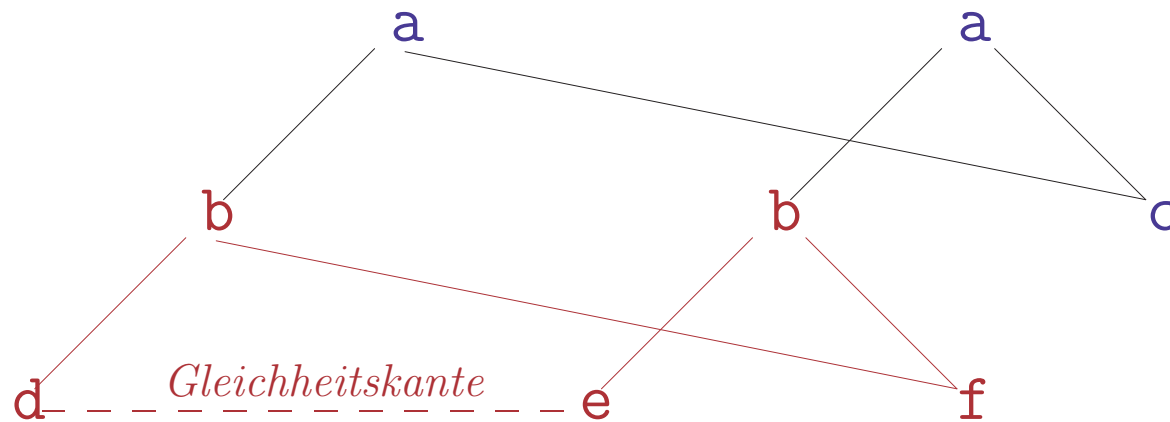
Zeige: $a(b(d,f),c) \doteq a(b(e,f),c)$ folgt aus $d \doteq e$



1. Verschmelze identische Knoten
2. Verbinde gleiche Knoten durch Gleichheitskante

GLEICHHEITSSCHLIESSEN AM BEISPIEL

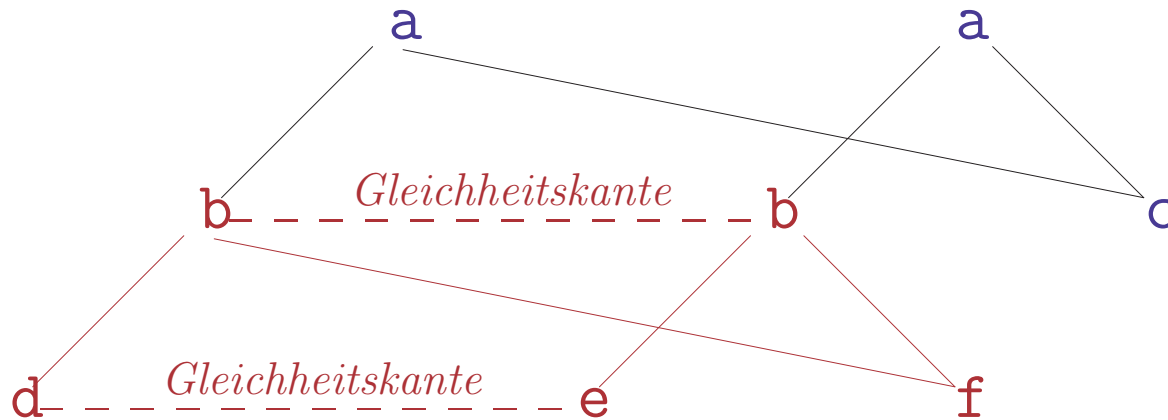
Zeige: $a(b(d,f),c) \doteq a(b(e,f),c)$ folgt aus $d \doteq e$



1. Verschmelze identische Knoten
2. Verbinde gleiche Knoten durch Gleichheitskante
3. Verbinde Wurzeln von Teilbäumen, die in allen Knoten gleich sind

GLEICHHEITSSCHLIESSEN AM BEISPIEL

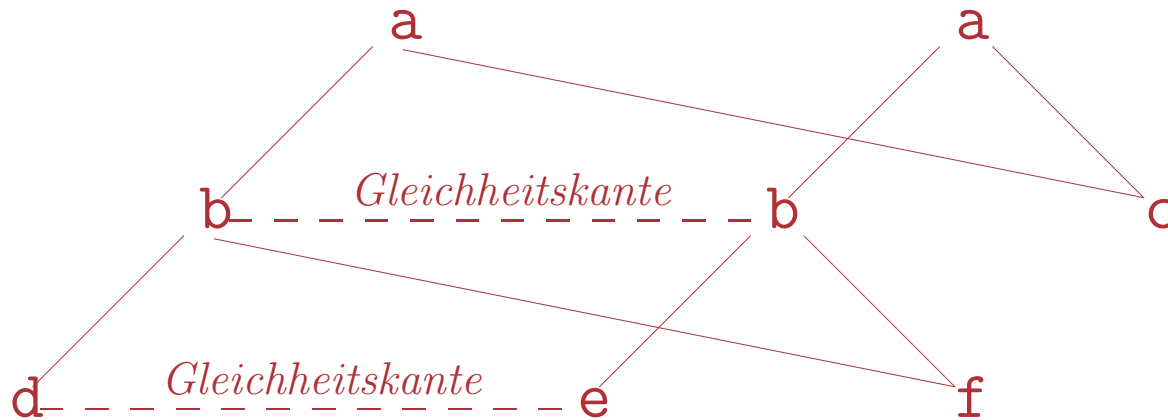
Zeige: $a(b(d,f),c) \doteq a(b(e,f),c)$ folgt aus $d \doteq e$



1. Verschmelze identische Knoten
2. Verbinde gleiche Knoten durch Gleichheitskante
3. Verbinde Wurzeln von Teilbäumen, die in allen Knoten gleich sind

GLEICHHEITSSCHLIESSEN AM BEISPIEL

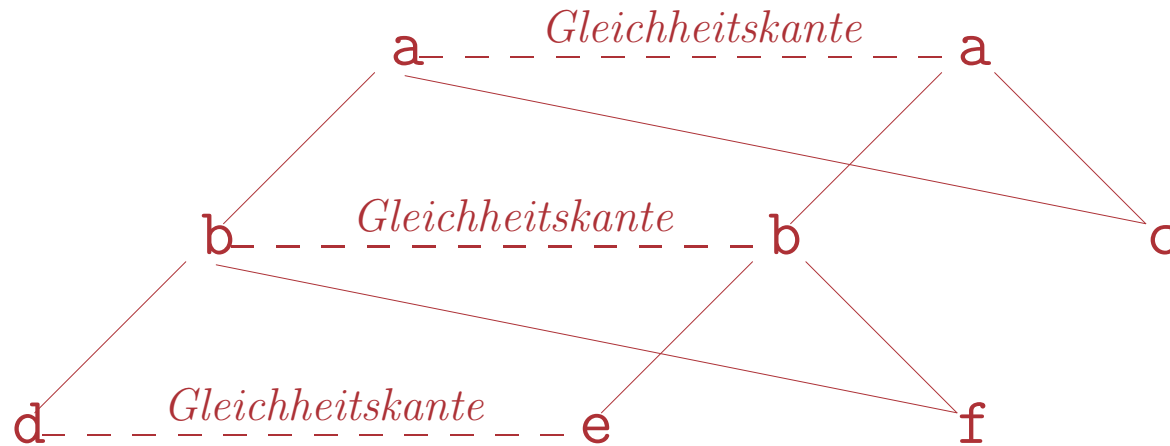
Zeige: $a(b(d,f),c) \doteq a(b(e,f),c)$ folgt aus $d \doteq e$



1. Verschmelze identische Knoten
2. Verbinde gleiche Knoten durch Gleichheitskante
3. Verbinde Wurzeln von Teilbäumen, die in allen Knoten gleich sind

GLEICHHEITSSCHLIESSEN AM BEISPIEL

Zeige: $a(b(d,f),c) \doteq a(b(e,f),c)$ folgt aus $d \doteq e$



1. Verschmelze identische Knoten
2. Verbinde gleiche Knoten durch Gleichheitskante
3. Verbinde Wurzeln von Teilbäumen, die in allen Knoten gleich sind

Gleichheit $\hat{=}$ Wurzeln der Termbäume sind verbunden

WICHTIGE GRAPHENTHEORETISCHE KONZEPTE

- **Notationen für gerichtete Graphen $G = (V, E)$**
 - $l(v)$: Markierung des Knoten v in G
 - $\delta(v)$: Anzahl der von v ausgehenden Kanten
 - $v[i]$: i -ter Nachfolgerknoten von v
 - u **Vorgänger** von v , wenn $v = u[i]$ für ein i

WICHTIGE GRAPHENTHEORETISCHE KONZEPTE

- **Notationen für gerichtete Graphen $G = (V, E)$**
 - $l(v)$: Markierung des Knoten v in G
 - $\delta(v)$: Anzahl der von v ausgehenden Kanten
 - $v[i]$: i -ter Nachfolgerknoten von v
 - u **Vorgänger** von v , wenn $v = u[i]$ für ein i
- **Begriffe für Äquivalenzrelationen R auf V**
 - u und v kongruent unter R ($u \sim_R v$):
 $l(u) = l(v)$, $\delta(u) = \delta(v)$ und für alle i $(u[i], v[i]) \in R$

WICHTIGE GRAPHENTHEORETISCHE KONZEPTE

- **Notationen für gerichtete Graphen $G = (V, E)$**
 - $l(v)$: Markierung des Knoten v in G
 - $\delta(v)$: Anzahl der von v ausgehenden Kanten
 - $v[i]$: i -ter Nachfolgerknoten von v
 - u **Vorgänger** von v , wenn $v = u[i]$ für ein i
- **Begriffe für Äquivalenzrelationen R auf V**
 - u und v **kongruent unter R** ($u \sim_R v$):
 $l(u) = l(v)$, $\delta(u) = \delta(v)$ und für alle i $(u[i], v[i]) \in R$
 - R **abgeschlossen unter Kongruenzen**: $u \sim_R v \Rightarrow (u, v) \in R$

WICHTIGE GRAPHENTHEORETISCHE KONZEPTE

- **Notationen für gerichtete Graphen $G = (V, E)$**

- $l(v)$: Markierung des Knoten v in G
- $\delta(v)$: Anzahl der von v ausgehenden Kanten
- $v[i]$: i -ter Nachfolgerknoten von v
- u **Vorgänger** von v , wenn $v = u[i]$ für ein i

- **Begriffe für Äquivalenzrelationen R auf V**

- u und v kongruent unter R ($u \sim_R v$):
 $l(u) = l(v)$, $\delta(u) = \delta(v)$ und für alle i $(u[i], v[i]) \in R$
- R abgeschlossen unter Kongruenzen: $u \sim_R v \Rightarrow (u, v) \in R$
- **Kongruenzabschluß R^*** : eindeutige minimale Erweiterung von R , die abgeschlossen unter Kongruenzen und Äquivalenzrelation ist
 $\hat{=} \text{Menge aller Äquivalenzen, die logisch aus } R \text{ folgen}$

GLEICHHEITSSCHLIESSEN ALS KONGRUENZABSCHLUSS

Folgt $s \doteq t$ aus $s_1 \doteq t_1, \dots, s_n \doteq t_n$?

GLEICHHEITSSCHLIESSEN ALS KONGRUENZABSCHLUSS

Folgt $s \doteq t$ aus $s_1 \doteq t_1, \dots, s_n \doteq t_n$?

- **Konstruiere Graph G von $s, s_1, \dots, s_n, t, t_1, \dots, t_n$**
 - G besteht aus Termbäumen von $s, s_1, \dots, s_n, t, t_1, \dots, t_n$
 - Identische Teilausdrücke werden durch denselben Teilbaum dargestellt

GLEICHHEITSSCHLIESSEN ALS KONGRUENZABSCHLUSS

Folgt $s \doteq t$ aus $s_1 \doteq t_1, \dots, s_n \doteq t_n$?

- **Konstruiere Graph G von $s, s_1, \dots, s_n, t, t_1, \dots, t_n$**
 - G besteht aus Termbäumen von $s, s_1, \dots, s_n, t, t_1, \dots, t_n$
 - Identische Teilausdrücke werden durch denselben Teilbaum dargestellt
- **Bestimme Kongruenzabschluß der $s_i \doteq t_i$ iterativ**

GLEICHHEITSSCHLIESSEN ALS KONGRUENZABSCHLUSS

Folgt $s \doteq t$ aus $s_1 \doteq t_1, \dots, s_n \doteq t_n$?

- **Konstruiere Graph G von $s, s_1, \dots, s_n, t, t_1, \dots, t_n$**
 - G besteht aus Termbäumen von $s, s_1, \dots, s_n, t, t_1, \dots, t_n$
 - Identische Teilausdrücke werden durch denselben Teilbaum dargestellt
- **Bestimme Kongruenzabschluß der $s_i \doteq t_i$ iterativ**
 - Start: Wähle R als Identitätsrelation auf den Knoten von G ($R^* = R$)

GLEICHHEITSSCHLIESSEN ALS KONGRUENZABSCHLUSS

Folgt $s \doteq t$ aus $s_1 \doteq t_1, \dots, s_n \doteq t_n$?

- **Konstruiere Graph G von $s, s_1, \dots, s_n, t, t_1, \dots, t_n$**
 - G besteht aus Termbäumen von $s, s_1, \dots, s_n, t, t_1, \dots, t_n$
 - Identische Teilausdrücke werden durch denselben Teilbaum dargestellt
- **Bestimme Kongruenzabschluß der $s_i \doteq t_i$ iterativ**
 - Start: Wähle R als Identitätsrelation auf den Knoten von G ($R^* = R$)
 - Im Schritt i bestimme Kongruenzabschluß von $R^* \cup \{(\tau(s_i), \tau(t_i))\}$
($\tau(u)$: Wurzelknoten des Termbaums von u)

GLEICHHEITSSCHLIESSEN ALS KONGRUENZABSCHLUSS

Folgt $s \doteq t$ aus $s_1 \doteq t_1, \dots, s_n \doteq t_n$?

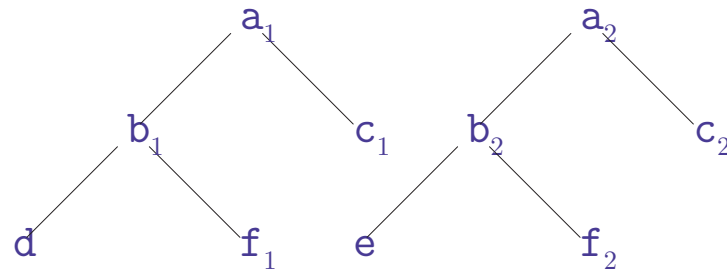
- **Konstruiere Graph G von $s, s_1, \dots, s_n, t, t_1, \dots, t_n$**
 - G besteht aus Termbäumen von $s, s_1, \dots, s_n, t, t_1, \dots, t_n$
 - Identische Teilausdrücke werden durch denselben Teilbaum dargestellt
- **Bestimme Kongruenzabschluß der $s_i \doteq t_i$ iterativ**
 - Start: Wähle R als Identitätsrelation auf den Knoten von G ($R^* = R$)
 - Im Schritt i bestimme Kongruenzabschluß von $R^* \cup \{(\tau(s_i), \tau(t_i))\}$
($\tau(u)$: Wurzelknoten des Termbaums von u)
 - Repräsentiere R^* als Menge von Äquivalenzklassen $\{ [u]_R \mid u \in V \}$
($[u]_R \equiv \{x \in V \mid (x, u) \in R\}$)

GLEICHHEITSSCHLIESSEN ALS KONGRUENZABSCHLUSS

Folgt $s \doteq t$ aus $s_1 \doteq t_1, \dots, s_n \doteq t_n$?

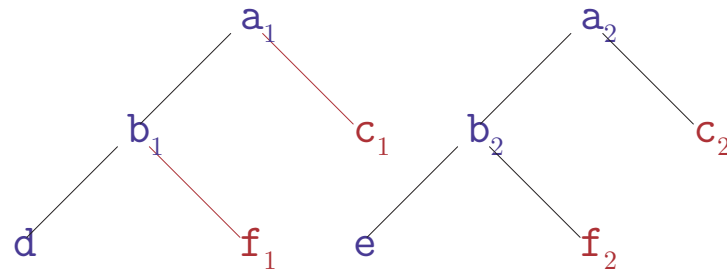
- **Konstruiere Graph G von $s, s_1, \dots, s_n, t, t_1, \dots, t_n$**
 - G besteht aus Termbäumen von $s, s_1, \dots, s_n, t, t_1, \dots, t_n$
 - Identische Teilausdrücke werden durch denselben Teilbaum dargestellt
- **Bestimme Kongruenzabschluß der $s_i \doteq t_i$ iterativ**
 - Start: Wähle R als Identitätsrelation auf den Knoten von G ($R^* = R$)
 - Im Schritt i bestimme Kongruenzabschluß von $R^* \cup \{(\tau(s_i), \tau(t_i))\}$
($\tau(u)$: Wurzelknoten des Termbaums von u)
 - Repräsentiere R^* als Menge von Äquivalenzklassen $\{ [u]_R \mid u \in V \}$
($[u]_R \equiv \{x \in V \mid (x, u) \in R\}$)
- **Teste Äquivalenz von s und t**
 - $s \doteq t$ gilt genau dann, wenn $(\tau(s), \tau(t)) \in R^*$

KONGRUENZABSCHLUSS: $d \doteq e \Rightarrow a(b(d, f), c) \doteq a(b(e, f), c)$



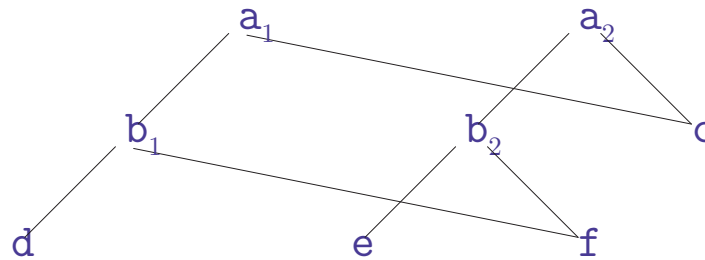
- Graph ist Termbaum von $a(b(d, f), c)$ und $a(b(e, f), c)$

KONGRUENZABSCHLUSS: $d \doteq e \Rightarrow a(b(d, f), c) \doteq a(b(e, f), c)$



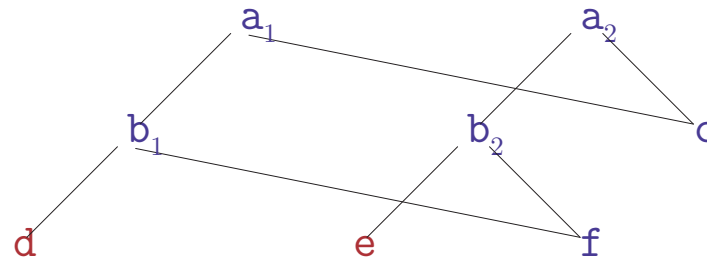
- Graph ist **Termbaum von $a(b(d, f), c)$ und $a(b(e, f), c)$**
 - Identische Teilausdrücke benutzen denselben Teilbaum

KONGRUENZABSCHLUSS: $d \doteq e \Rightarrow a(b(d, f), c) \doteq a(b(e, f), c)$



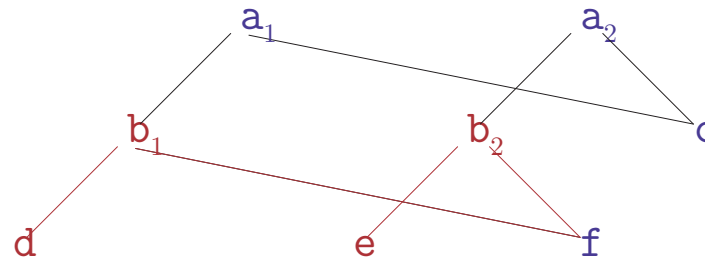
- Graph ist **Termbaum von $a(b(d, f), c)$ und $a(b(e, f), c)$**
 - Identische Teilausdrücke benutzen denselben Teilbaum
 - Initiale Relation: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d\}, \{e\}, \{f\} \}$

KONGRUENZABSCHLUSS: $d \doteq e \Rightarrow a(b(d, f), c) \doteq a(b(e, f), c)$



- Graph ist **Termbaum von $a(b(d, f), c)$ und $a(b(e, f), c)$**
 - Identische Teilausdrücke benutzen denselben Teilbaum
 - Initiale Relation: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d\}, \{e\}, \{f\} \}$
- **Hinzunahme von $d \doteq e$**

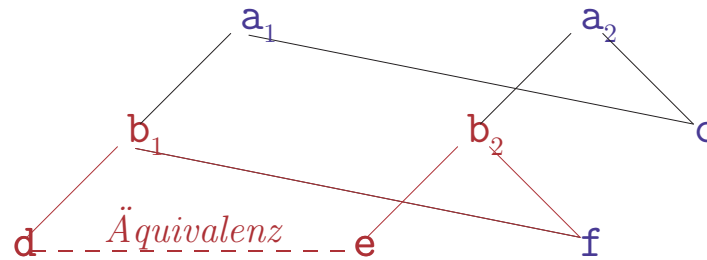
KONGRUENZABSCHLUSS: $d \doteq e \Rightarrow a(b(d, f), c) \doteq a(b(e, f), c)$



- **Graph ist Termbaum von $a(b(d, f), c)$ und $a(b(e, f), c)$**
 - Identische Teilausdrücke benutzen denselben Teilbaum
 - Initiale Relation: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d\}, \{e\}, \{f\} \}$
- **Hinzunahme von $d \doteq e$**

Bestimme Vorgänger von $[d]_R$ ($\{b_1\}$) und $[e]_R$ ($\{b_2\}$)

KONGRUENZABSCHLUSS: $d \doteq e \Rightarrow a(b(d, f), c) \doteq a(b(e, f), c)$

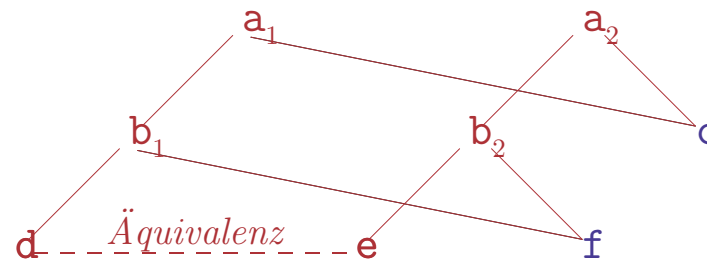


- Graph ist **Termbaum von $a(b(d, f), c)$ und $a(b(e, f), c)$**
 - Identische Teilausdrücke benutzen denselben Teilbaum
 - Initiale Relation: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d\}, \{e\}, \{f\} \}$
- **Hinzunahme von $d \doteq e$**

Bestimme Vorgänger von $[d]_R$ ($\{b_1\}$) und $[e]_R$ ($\{b_2\}$)

 - Vereinige $[d]_R$ und $[e]_R$: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d, e\}, \{f\} \}$

KONGRUENZABSCHLUSS: $d \doteq e \Rightarrow a(b(d, f), c) \doteq a(b(e, f), c)$



- Graph ist **Termbaum von $a(b(d, f), c)$ und $a(b(e, f), c)$**

- Identische Teilausdrücke benutzen denselben Teilbaum

- Initiale Relation: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d\}, \{e\}, \{f\} \}$

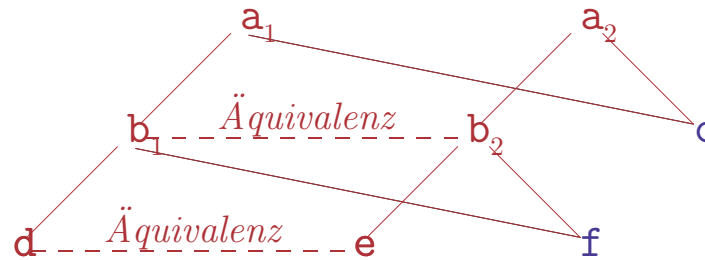
- **Hinzunahme von $d \doteq e$**

Bestimme Vorgänger von $[d]_R$ ($\{b_1\}$) und $[e]_R$ ($\{b_2\}$)

- Vereinige $[d]_R$ und $[e]_R$: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d, e\}, \{f\} \}$

Bestimme Vorgänger von $[b_1]_R$ ($\{a_1\}$) und $[b_2]_R$ ($\{a_2\}$)

KONGRUENZABSCHLUSS: $d \doteq e \Rightarrow a(b(d, f), c) \doteq a(b(e, f), c)$



- **Graph ist Termbaum von $a(b(d, f), c)$ und $a(b(e, f), c)$**

- Identische Teilausdrücke benutzen denselben Teilbaum

- Initiale Relation: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d\}, \{e\}, \{f\} \}$

- **Hinzunahme von $d \doteq e$**

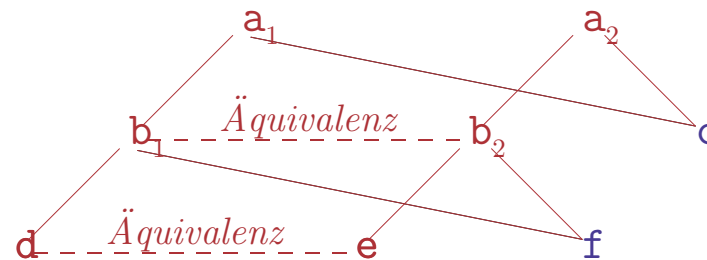
Bestimme Vorgänger von $[d]_R$ ($\{b_1\}$) und $[e]_R$ ($\{b_2\}$)

- Vereinige $[d]_R$ und $[e]_R$: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d, e\}, \{f\} \}$

Bestimme Vorgänger von $[b_1]_R$ ($\{a_1\}$) und $[b_2]_R$ ($\{a_2\}$)

- Vereinige $[b_1]_R$ und $[b_2]_R$: $R := \{ \{a_1\}, \{a_2\}, \{b_1, b_2\}, \{c\}, \{d, e\}, \{f\} \}$

KONGRUENZABSCHLUSS: $d \doteq e \Rightarrow a(b(d, f), c) \doteq a(b(e, f), c)$



- **Graph ist Termbaum von $a(b(d, f), c)$ und $a(b(e, f), c)$**

- Identische Teilausdrücke benutzen denselben Teilbaum

- Initiale Relation: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d\}, \{e\}, \{f\} \}$

- **Hinzunahme von $d \doteq e$**

Bestimme Vorgänger von $[d]_R$ ($\{b_1\}$) und $[e]_R$ ($\{b_2\}$)

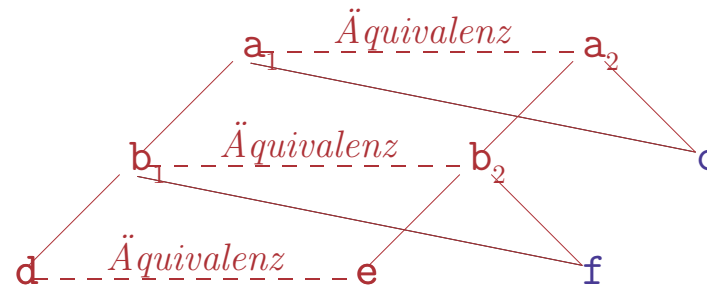
- Vereinige $[d]_R$ und $[e]_R$: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d, e\}, \{f\} \}$

Bestimme Vorgänger von $[b_1]_R$ ($\{a_1\}$) und $[b_2]_R$ ($\{a_2\}$)

- Vereinige $[b_1]_R$ und $[b_2]_R$: $R := \{ \{a_1\}, \{a_2\}, \{b_1, b_2\}, \{c\}, \{d, e\}, \{f\} \}$

Bestimme Vorgänger von $[a_1]_R$ (\emptyset) und $[a_2]_R$ (\emptyset)

KONGRUENZABSCHLUSS: $d \doteq e \Rightarrow a(b(d, f), c) \doteq a(b(e, f), c)$



- **Graph ist Termbaum von $a(b(d, f), c)$ und $a(b(e, f), c)$**

- Identische Teilausdrücke benutzen denselben Teilbaum

- Initiale Relation: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d\}, \{e\}, \{f\} \}$

- **Hinzunahme von $d \doteq e$**

Bestimme Vorgänger von $[d]_R$ ($\{b_1\}$) und $[e]_R$ ($\{b_2\}$)

- Vereinige $[d]_R$ und $[e]_R$: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d, e\}, \{f\} \}$

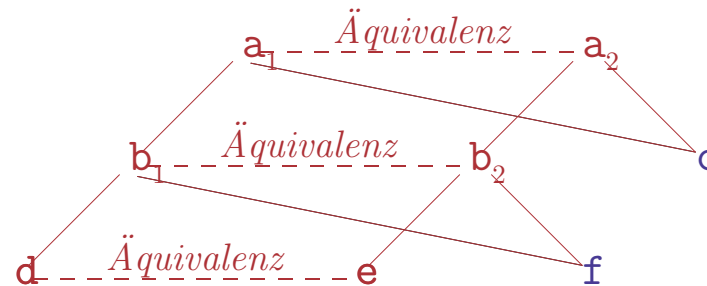
Bestimme Vorgänger von $[b_1]_R$ ($\{a_1\}$) und $[b_2]_R$ ($\{a_2\}$)

- Vereinige $[b_1]_R$ und $[b_2]_R$: $R := \{ \{a_1\}, \{a_2\}, \{b_1, b_2\}, \{c\}, \{d, e\}, \{f\} \}$

Bestimme Vorgänger von $[a_1]_R$ (\emptyset) und $[a_2]_R$ (\emptyset)

- Vereinige $[a_1]_R$ und $[a_2]_R$: $R := \{ \{a_1, a_2\}, \{b_1, b_2\}, \{c\}, \{d, e\}, \{f\} \}$

KONGRUENZABSCHLUSS: $d \doteq e \Rightarrow a(b(d, f), c) \doteq a(b(e, f), c)$



- **Graph ist Termbaum von $a(b(d, f), c)$ und $a(b(e, f), c)$**

- Identische Teilausdrücke benutzen denselben Teilbaum

- Initiale Relation: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d\}, \{e\}, \{f\} \}$

- **Hinzunahme von $d \doteq e$**

Bestimme Vorgänger von $[d]_R$ ($\{b_1\}$) und $[e]_R$ ($\{b_2\}$)

- Vereinige $[d]_R$ und $[e]_R$: $R := \{ \{a_1\}, \{a_2\}, \{b_1\}, \{b_2\}, \{c\}, \{d, e\}, \{f\} \}$

Bestimme Vorgänger von $[b_1]_R$ ($\{a_1\}$) und $[b_2]_R$ ($\{a_2\}$)

- Vereinige $[b_1]_R$ und $[b_2]_R$: $R := \{ \{a_1\}, \{a_2\}, \{b_1, b_2\}, \{c\}, \{d, e\}, \{f\} \}$

Bestimme Vorgänger von $[a_1]_R$ (\emptyset) und $[a_2]_R$ (\emptyset)

- Vereinige $[a_1]_R$ und $[a_2]_R$: $R := \{ \{a_1, a_2\}, \{b_1, b_2\}, \{c\}, \{d, e\}, \{f\} \}$

Wurzelknoten der beiden Terme sind äquivalent

BERECHNE KONGRUENZABSCHLUSS VON $R \cup \{(u, v)\}$

- **Algorithmus** $\text{MERGE}(R, u, v)$

- Eingabe: gerichteter Graph $G = (V, E)$, $u, v \in V$

- Äquivalenzrelation R (abgeschlossen unter Kongruenzen)

BERECHNE KONGRUENZABSCHLUSS VON $R \cup \{(u, v)\}$

- **Algorithmus** $\text{MERGE}(R, u, v)$

- Eingabe: gerichteter Graph $G = (V, E)$, $u, v \in V$

- Äquivalenzrelation R (abgeschlossen unter Kongruenzen)

- **Falls** $u \sim_R v$, **dann halte mit Ergebnis** R

- Es gilt $(R \cup \{(u, v)\})^* = R$

BERECHNE KONGRUENZABSCHLUSS VON $R \cup \{(u, v)\}$

- **Algorithmus MERGE(R, u, v)**
 - Eingabe: gerichteter Graph $G = (V, E)$, $u, v \in V$
Äquivalenzrelation R (abgeschlossen unter Kongruenzen)
- **Falls $u \sim_R v$, dann halte mit Ergebnis R**
 - Es gilt $(R \cup \{(u, v)\})^* = R$
- **Andernfalls modifiziere R durch Verschmelzung**
 - Setze $P_u := \{x \in V \mid \exists w \in [u]_R. x \text{ Vorgänger von } w\}$
 - Setze $P_v := \{x \in V \mid \exists w \in [v]_R. x \text{ Vorgänger von } w\}$

BERECHNE KONGRUENZABSCHLUSS VON $R \cup \{(u, v)\}$

- **Algorithmus MERGE(R, u, v)**

- Eingabe: gerichteter Graph $G = (V, E)$, $u, v \in V$

- Äquivalenzrelation R (abgeschlossen unter Kongruenzen)

- **Falls $u \sim_R v$, dann halte mit Ergebnis R**

- Es gilt $(R \cup \{(u, v)\})^* = R$

- **Andernfalls modifiziere R durch Verschmelzung**

- Setze $P_u := \{x \in V \mid \exists w \in [u]_R. x \text{ Vorgänger von } w\}$

- Setze $P_v := \{x \in V \mid \exists w \in [v]_R. x \text{ Vorgänger von } w\}$

- Vereinige Äquivalenzklassen $[u]_R$ und $[v]_R$ in R

BERECHNE KONGRUENZABSCHLUSS VON $R \cup \{(u, v)\}$

- **Algorithmus MERGE(R, u, v)**

- Eingabe: gerichteter Graph $G = (V, E)$, $u, v \in V$

- Äquivalenzrelation R (abgeschlossen unter Kongruenzen)

- **Falls $u \sim_R v$, dann halte mit Ergebnis R**

- Es gilt $(R \cup \{(u, v)\})^* = R$

- **Andernfalls modifiziere R durch Verschmelzung**

- Setze $P_u := \{x \in V \mid \exists w \in [u]_R. x \text{ Vorgänger von } w\}$

- Setze $P_v := \{x \in V \mid \exists w \in [v]_R. x \text{ Vorgänger von } w\}$

- Vereinige Äquivalenzklassen $[u]_R$ und $[v]_R$ in R

- Wiederhole für $x \in P_u$ und $y \in P_v$

- Falls $x \sim_R y$ und $[x]_R \neq [y]_R$ dann setze $R := \text{MERGE}(R, x, y)$

BERECHNE KONGRUENZABSCHLUSS VON $R \cup \{(u, v)\}$

- **Algorithmus MERGE(R, u, v)**

- Eingabe: gerichteter Graph $G = (V, E)$, $u, v \in V$
Äquivalenzrelation R (abgeschlossen unter Kongruenzen)

- **Falls $u \sim_R v$, dann halte mit Ergebnis R**

- Es gilt $(R \cup \{(u, v)\})^* = R$

- **Andernfalls modifiziere R durch Verschmelzung**

- Setze $P_u := \{x \in V \mid \exists w \in [u]_R. x \text{ Vorgänger von } w\}$
- Setze $P_v := \{x \in V \mid \exists w \in [v]_R. x \text{ Vorgänger von } w\}$
- Vereinige Äquivalenzklassen $[u]_R$ und $[v]_R$ in R
- Wiederhole für $x \in P_u$ und $y \in P_v$

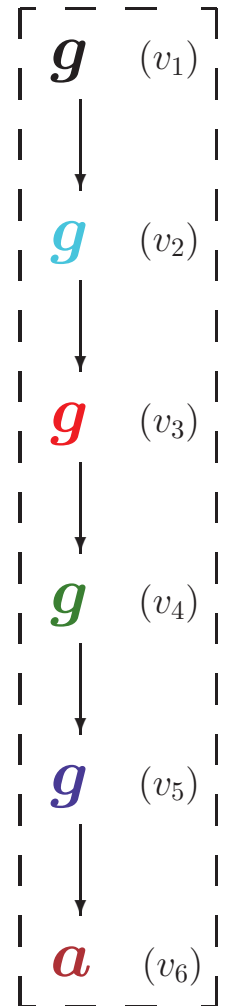
Falls $x \sim_R y$ und $[x]_R \neq [y]_R$ dann setze $R := \text{MERGE}(R, x, y)$

Halte mit der modifizierten Relation R als Ergebnis

KONGRUENZABSCHLUSS: $g(g(g(a))) \doteq a$, $g(g(g(g(g(a)))) \doteq a$

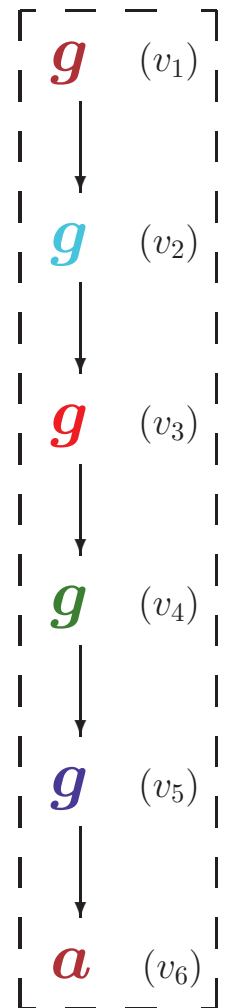
- Graph ist **Termbaum** von $g(g(g(g(g(a))))$

– Initiale Relation: $R := \{ \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_6\} \}$



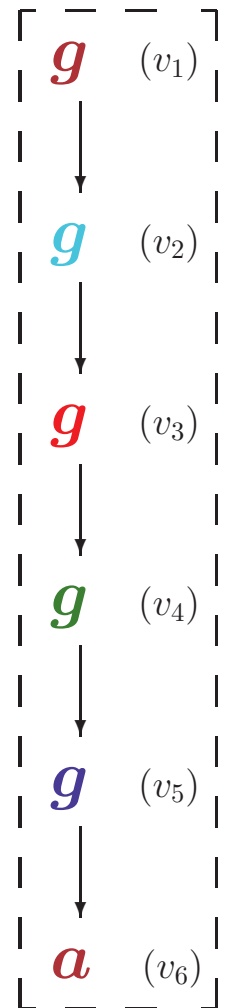
KONGRUENZABSCHLUSS: $g(g(g(a))) \doteq a$, $g(g(g(g(g(a)))) \doteq a$

- Graph ist **Termbaum** von $g(g(g(g(g(a))))$
 - Initiale Relation: $R := \{ \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_6\} \}$
- **Hinzunahme** von $g(g(g(g(g(a)))) \doteq a$
 - $R := \{ \{v_1, v_6\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\} \}$ ist abgeschlossen



KONGRUENZABSCHLUSS: $g(g(g(a))) \doteq a$, $g(g(g(g(g(a)))) \doteq a$

- Graph ist **Termbaum** von $g(g(g(g(g(a))))$
 - Initiale Relation: $R := \{ \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_6\} \}$
- **Hinzunahme** von $g(g(g(g(g(a)))) \doteq a$
 - $R := \{ \{v_1, v_6\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\} \}$ ist abgeschlossen
- **Hinzunahme** von $g(g(g(a))) \doteq a$

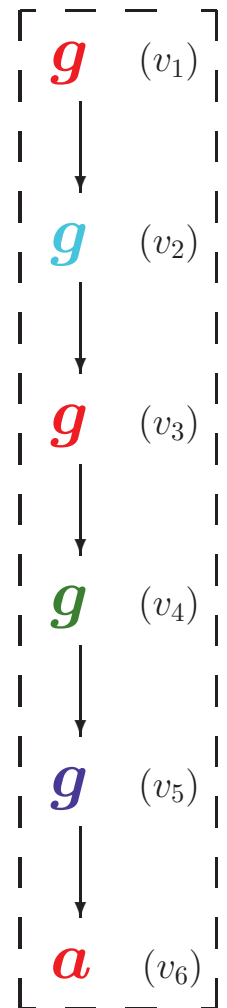


KONGRUENZABSCHLUSS: $g(g(g(a))) \doteq a$, $g(g(g(g(g(a)))) \doteq a$

- Graph ist **Termbaum** von $g(g(g(g(g(a))))$
 - Initiale Relation: $R := \{ \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_6\} \}$
- **Hinzunahme** von $g(g(g(g(g(a)))) \doteq a$
 - $R := \{ \{v_1, v_6\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\} \}$ ist abgeschlossen
- **Hinzunahme** von $g(g(g(a))) \doteq a$

MERGE(R, v_3, v_6):

- $P_{v_3} := \{v_2\}$, $P_{v_6} := \{v_5\}$, $R := \{ \{v_1, v_6, v_3\}, \{v_2\}, \{v_4\}, \{v_5\} \}$
- Wegen $(v_3, v_6) \in R$ gilt $v_2 \sim_R v_5$ aber $[v_2]_R \neq [v_5]_R$



KONGRUENZABSCHLUSS: $g(g(g(a))) \doteq a, \quad g(g(g(g(g(a)))) \doteq a$

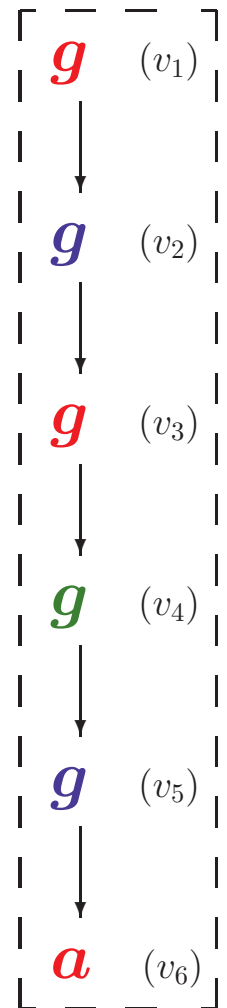
- Graph ist **Termbaum** von $g(g(g(g(g(a))))$
 - Initiale Relation: $R := \{ \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_6\} \}$
- **Hinzunahme** von $g(g(g(g(g(a)))) \doteq a$
 - $R := \{ \{v_1, v_6\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\} \}$ ist abgeschlossen
- **Hinzunahme** von $g(g(g(a))) \doteq a$

MERGE(R, v_3, v_6):

- $P_{v_3} := \{v_2\}, P_{v_6} := \{v_5\}, R := \{ \{v_1, v_6, v_3\}, \{v_2\}, \{v_4\}, \{v_5\} \}$
- Wegen $(v_3, v_6) \in R$ gilt $v_2 \sim_R v_5$ aber $[v_2]_R \neq [v_5]_R$

MERGE(R, v_2, v_5):

- $P_{v_2} := \{v_1\}, P_{v_5} := \{v_4\}, R := \{ \{v_1, v_6, v_3\}, \{v_4\}, \{v_2, v_5\} \}$
- Wegen $(v_2, v_5) \in R$ gilt $v_1 \sim_R v_4$ aber $[v_1]_R \neq [v_4]_R$



KONGRUENZABSCHLUSS: $g(g(g(a))) \doteq a, \quad g(g(g(g(g(a)))) \doteq a$

- **Graph ist Termbaum von $g(g(g(g(g(a))))$**
 - Initiale Relation: $R := \{ \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_6\} \}$
- **Hinzunahme von $g(g(g(g(g(a)))) \doteq a$**
 - $R := \{ \{v_1, v_6\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\} \}$ ist abgeschlossen
- **Hinzunahme von $g(g(g(a))) \doteq a$**

MERGE(R, v_3, v_6):

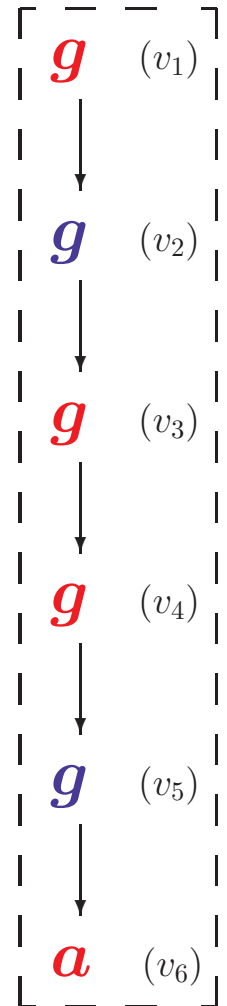
- $P_{v_3} := \{v_2\}, P_{v_6} := \{v_5\}, R := \{ \{v_1, v_6, v_3\}, \{v_2\}, \{v_4\}, \{v_5\} \}$
- Wegen $(v_3, v_6) \in R$ gilt $v_2 \sim_R v_5$ aber $[v_2]_R \neq [v_5]_R$

MERGE(R, v_2, v_5):

- $P_{v_2} := \{v_1\}, P_{v_5} := \{v_4\}, R := \{ \{v_1, v_6, v_3\}, \{v_4\}, \{v_2, v_5\} \}$
- Wegen $(v_2, v_5) \in R$ gilt $v_1 \sim_R v_4$ aber $[v_1]_R \neq [v_4]_R$

MERGE(R, v_1, v_4):

- $P_{v_1} := \{v_2, v_5\}, P_{v_4} := \{v_3\}, R := \{ \{v_1, v_6, v_3, v_4\}, \{v_2, v_5\} \}$
- Wegen $(v_6, v_4) \in R$ gilt $v_5 \sim_R v_3$ aber $[v_5]_R \neq [v_3]_R$



KONGRUENZABSCHLUSS: $g(g(g(a))) \doteq a$, $g(g(g(g(g(a)))) \doteq a$

- Graph ist **Termbaum** von $g(g(g(g(g(a))))$

- Initiale Relation: $R := \{ \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_6\} \}$

- **Hinzunahme** von $g(g(g(g(g(a)))) \doteq a$

- $R := \{ \{v_1, v_6\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\} \}$ ist abgeschlossen

- **Hinzunahme** von $g(g(g(a))) \doteq a$

MERGE(R, v_3, v_6):

- $P_{v_3} := \{v_2\}$, $P_{v_6} := \{v_5\}$, $R := \{ \{v_1, v_6, v_3\}, \{v_2\}, \{v_4\}, \{v_5\} \}$

- Wegen $(v_3, v_6) \in R$ gilt $v_2 \sim_R v_5$ aber $[v_2]_R \neq [v_5]_R$

MERGE(R, v_2, v_5):

- $P_{v_2} := \{v_1\}$, $P_{v_5} := \{v_4\}$, $R := \{ \{v_1, v_6, v_3\}, \{v_4\}, \{v_2, v_5\} \}$

- Wegen $(v_2, v_5) \in R$ gilt $v_1 \sim_R v_4$ aber $[v_1]_R \neq [v_4]_R$

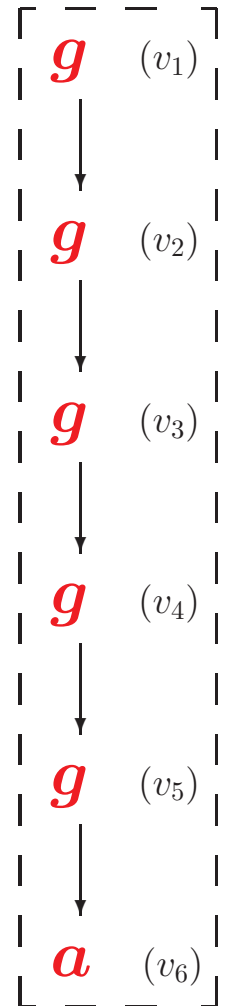
MERGE(R, v_1, v_4):

- $P_{v_1} := \{v_2, v_5\}$, $P_{v_4} := \{v_3\}$, $R := \{ \{v_1, v_6, v_3, v_4\}, \{v_2, v_5\} \}$

- Wegen $(v_6, v_4) \in R$ gilt $v_5 \sim_R v_3$ aber $[v_5]_R \neq [v_3]_R$

MERGE(R, v_5, v_3):

- $P_{v_5} := \{v_1, v_4\}$, $P_{v_3} := \{v_2, v_5, v_3\}$, $R := \{ \{v_1, v_6, v_3, v_4, v_2, v_5\} \}$



KONGRUENZABSCHLUSS: $g(g(g(a))) \doteq a$, $g(g(g(g(g(a)))) \doteq a$

- Graph ist **Termbaum** von $g(g(g(g(g(a))))$

- Initiale Relation: $R := \{ \{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}, \{v_6\} \}$

- **Hinzunahme** von $g(g(g(g(g(a)))) \doteq a$

- $R := \{ \{v_1, v_6\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\} \}$ ist abgeschlossen

- **Hinzunahme** von $g(g(g(a))) \doteq a$

MERGE(R, v_3, v_6):

- $P_{v_3} := \{v_2\}$, $P_{v_6} := \{v_5\}$, $R := \{ \{v_1, v_6, v_3\}, \{v_2\}, \{v_4\}, \{v_5\} \}$

- Wegen $(v_3, v_6) \in R$ gilt $v_2 \sim_R v_5$ aber $[v_2]_R \neq [v_5]_R$

MERGE(R, v_2, v_5):

- $P_{v_2} := \{v_1\}$, $P_{v_5} := \{v_4\}$, $R := \{ \{v_1, v_6, v_3\}, \{v_4\}, \{v_2, v_5\} \}$

- Wegen $(v_2, v_5) \in R$ gilt $v_1 \sim_R v_4$ aber $[v_1]_R \neq [v_4]_R$

MERGE(R, v_1, v_4):

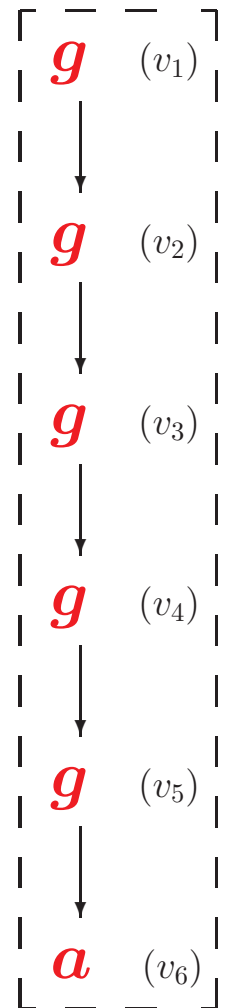
- $P_{v_1} := \{v_2, v_5\}$, $P_{v_4} := \{v_3\}$, $R := \{ \{v_1, v_6, v_3, v_4\}, \{v_2, v_5\} \}$

- Wegen $(v_6, v_4) \in R$ gilt $v_5 \sim_R v_3$ aber $[v_5]_R \neq [v_3]_R$

MERGE(R, v_5, v_3):

- $P_{v_5} := \{v_1, v_4\}$, $P_{v_3} := \{v_2, v_5, v_3\}$, $R := \{ \{v_1, v_6, v_3, v_4, v_2, v_5\} \}$

Alle Knoten sind äquivalent: $R=R^*$



● Resolutionsähnliche Kalkülregel **Paramodulation**

- Wähle zwei **Elternklauseln** $\{L\}UC_1$ und $\{r \doteq s\}UC_2$
und **kennzeichne** einen Teilterm t von L
- Bestimme **allgemeinsten Unifikator** σ von r und t
- Generiere **Paramodulant** $\sigma(\{L'\}UC_1UC_2)$, wobei $L' = "L[s/t]"$

● Resolutionsähnliche Kalkülregel **Paramodulation**

- Wähle zwei **Elternklauseln** $\{L\}UC_1$ und $\{r \doteq s\}UC_2$
und **kennzeichne** einen Teilterm t von L
- Bestimme **allgemeinsten Unifikator** σ von r und t
- Generiere **Paramodulant** $\sigma(\{L'\}UC_1UC_2)$, wobei $L' = "L[s/t]"$

Zusammen mit Resolution vollständig für Prädikatenlogik mit Gleichheit

● Resolutionsähnliche Kalkülregel **Paramodulation**

- Wähle zwei **Elternklauseln** $\{L\}UC_1$ und $\{r \doteq s\}UC_2$
und **kennzeichne** einen Teilterm t von L
- Bestimme **allgemeinsten Unifikator** σ von r und t
- Generiere **Paramodulant** $\sigma(\{L'\}UC_1UC_2)$, wobei $L' = "L[s/t]"$

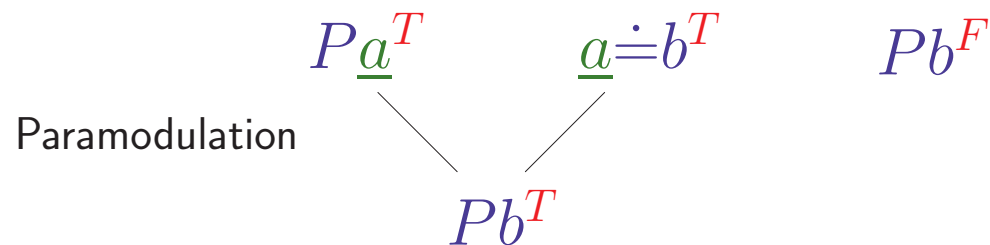
Zusammen mit Resolution vollständig für Prädikatenlogik mit Gleichheit

$$P\underline{a}^T \quad \underline{a} \doteq b^T \quad Pb^F$$

● Resolutionsähnliche Kalkülregel **Paramodulation**

- Wähle zwei **Elternklauseln** $\{L\}UC_1$ und $\{r \doteq s\}UC_2$ und **kennzeichne** einen Teilterm t von L
- Bestimme **allgemeinsten Unifikator** σ von r und t
- Generiere **Paramodulant** $\sigma(\{L'\}UC_1UC_2)$, wobei $L' = "L[s/t]"$

Zusammen mit Resolution vollständig für Prädikatenlogik mit Gleichheit

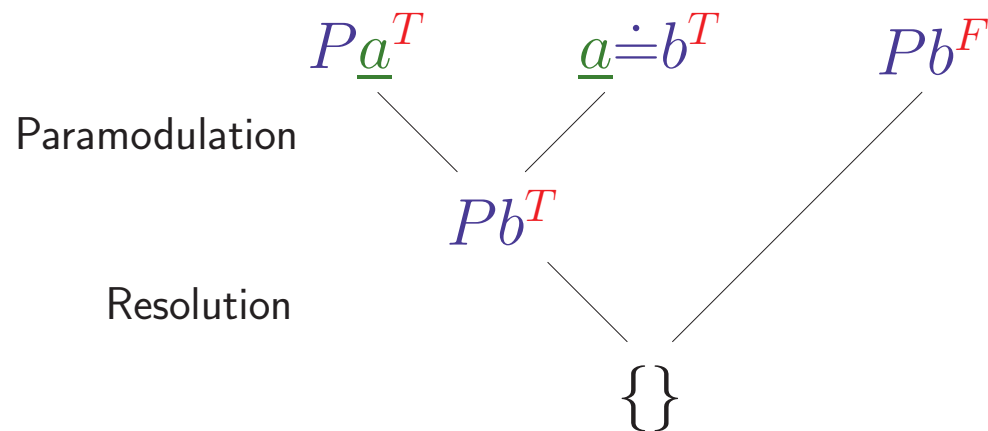


GLEICHHEITSBEHANDLUNG IN RESOLUTIONSBEWEISERN

● Resolutionsähnliche Kalkülregel **Paramodulation**

- Wähle zwei **Elternklauseln** $\{L\}UC_1$ und $\{r \doteq s\}UC_2$ und **kennzeichne** einen Teilterm t von L
- Bestimme **allgemeinsten Unifikator** σ von r und t
- Generiere **Paramodulant** $\sigma(\{L'\}UC_1UC_2)$, wobei $L' = "L[s/t]"$

Zusammen mit Resolution vollständig für Prädikatenlogik mit Gleichheit

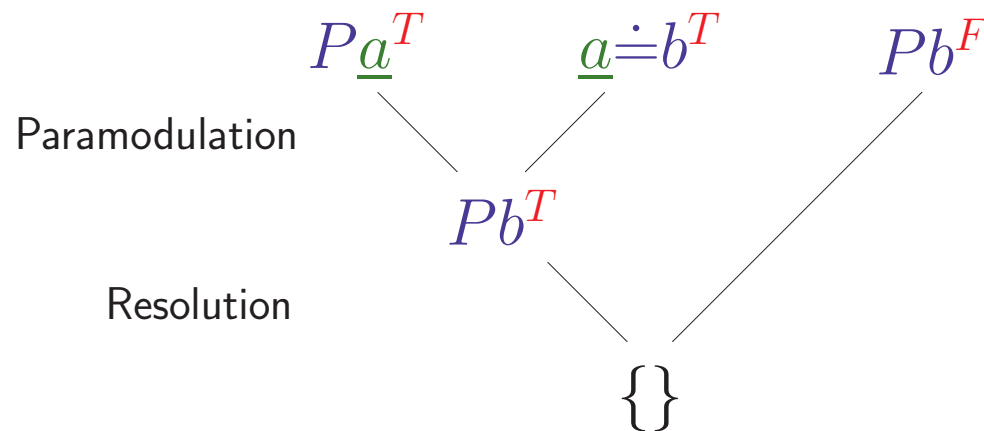


GLEICHHEITSBEHANDLUNG IN RESOLUTIONSBEWEISERN

● Resolutionsähnliche Kalkülregel **Paramodulation**

- Wähle zwei **Elternklauseln** $\{L\}UC_1$ und $\{r \doteq s\}UC_2$ und kennzeichne einen Teilterm t von L
- Bestimme allgemeinsten Unifikator σ von r und t
- Generiere **Paramodulant** $\sigma(\{L'\}UC_1UC_2)$, wobei $L' = "L[s/t]"$

Zusammen mit Resolution vollständig für Prädikatenlogik mit Gleichheit



● Alternative Sicht als **bedingte Termersetzungsregel**

- C_2 ist Bedingung für die Ersetzung von r durch s in einem Literal L

PROBLEME DER PARAMODULATION MIT UNIFIKATION

- Welche Richtung einer Gleichheit wird gewählt?

$$\begin{array}{ccc} Qb \vee P\underline{x}^T & & \underline{a} \doteq b \vee Rb^T \\ & \searrow & \swarrow \\ & Qb \vee Pb \vee Rb^T & \\ & \sigma = [a/x] & \end{array}$$

PROBLEME DER PARAMODULATION MIT UNIFIKATION

- Welche Richtung einer Gleichheit wird gewählt?

$$Qb \vee P\underline{x}^T \quad \underline{a} \doteq b \vee Rb^T$$

$$Qb \vee Pb \vee Rb^T$$

$$\sigma = [a/x]$$

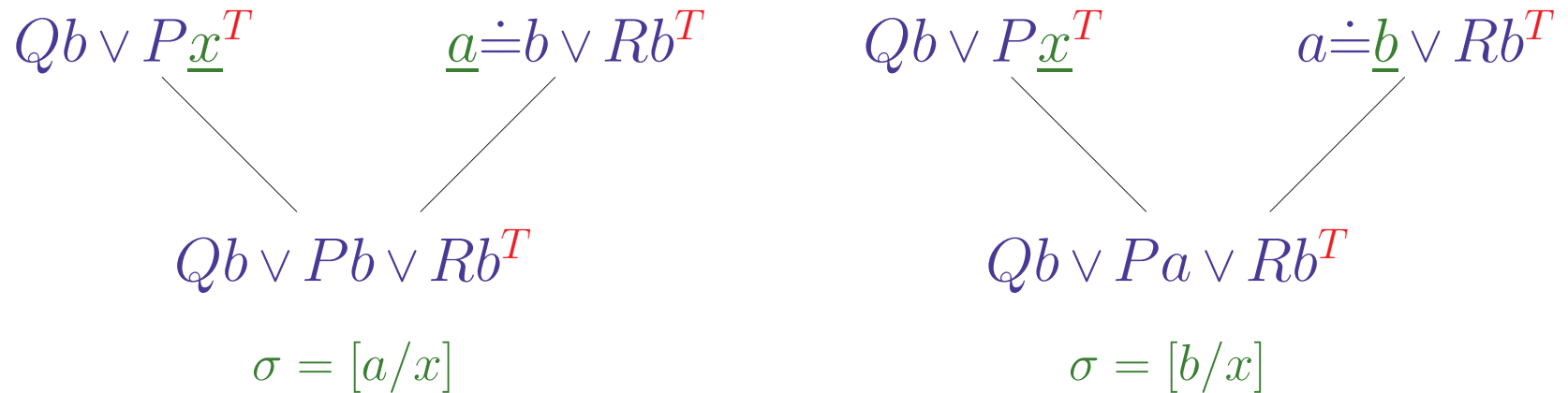
$$Qb \vee P\underline{x}^T \quad a \doteq \underline{b} \vee Rb^T$$

$$Qb \vee Pa \vee Rb^T$$

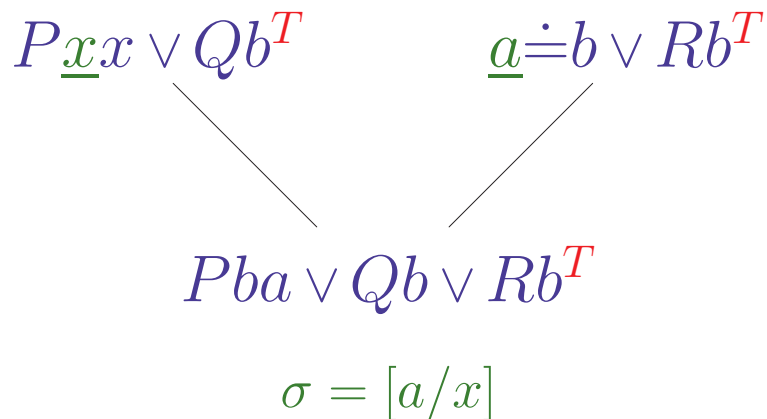
$$\sigma = [b/x]$$

PROBLEME DER PARAMODULATION MIT UNIFIKATION

- Welche Richtung einer Gleichheit wird gewählt?

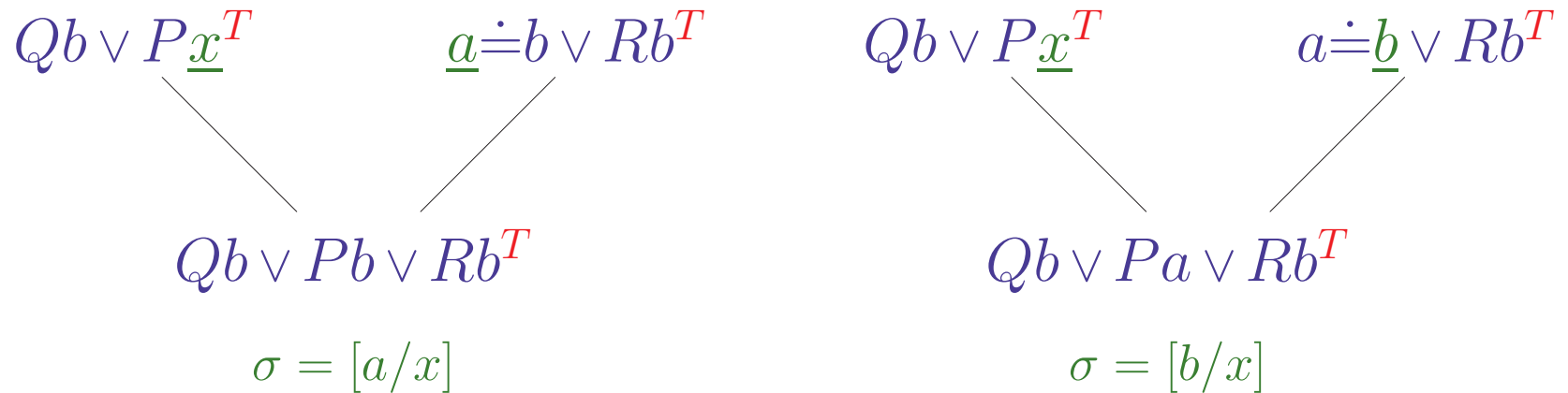


- Welches Variablenvorkommen wird ersetzt?

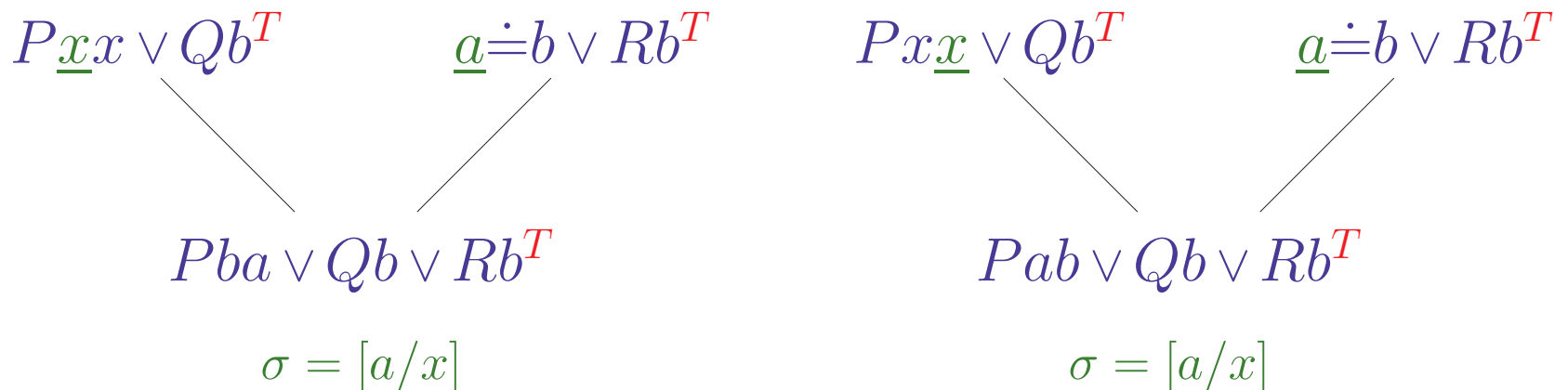


PROBLEME DER PARAMODULATION MIT UNIFIKATION

- Welche Richtung einer Gleichheit wird gewählt?



- Welches Variablenvorkommen wird ersetzt?



- **Koppelung von Resolution + Paramodulation**
 - Vollständig und konsistent für Prädikatenlogik mit Gleichheit
 - Benötigt Reflexivitätsaxiom

RESOLUTION UND GLEICHHEIT

- **Koppelung von Resolution + Paramodulation**
 - Vollständig und konsistent für Prädikatenlogik mit Gleichheit
 - Benötigt Reflexivitätsaxiom
- **Lokale Sicht bedeutet großen Suchraum**
 - Viele Paramodulanten möglich
 - Effiziente Suchstrategien erforderlich

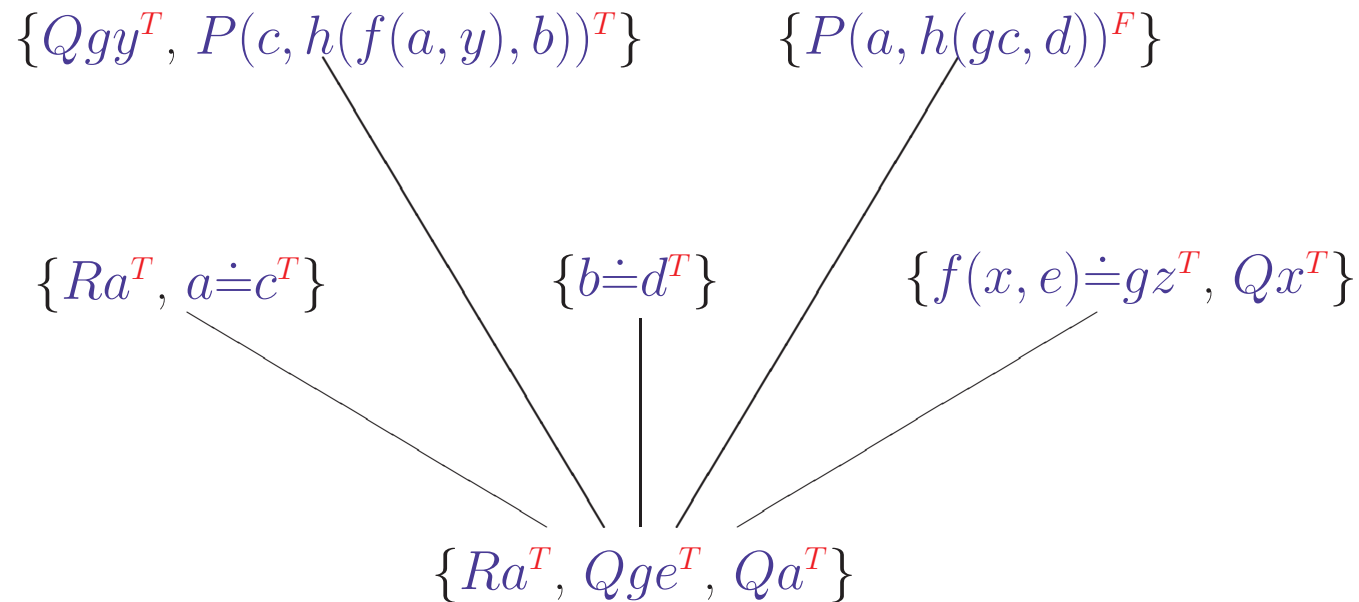
- **Koppelung von Resolution + Paramodulation**
 - Vollständig und konsistent für Prädikatenlogik mit Gleichheit
 - Benötigt Reflexivitätsaxiom
- **Lokale Sicht bedeutet großen Suchraum**
 - Viele Paramodulanten möglich
 - Effiziente Suchstrategien erforderlich
- **Strategien für Resolution mit Gleichheit**
 - **Demodulation**: gerichtete Anwendung von Gleichheiten ↦ Rewriting
 - **E-Resolution**: komplexes Äquivalent zu eq-Konnektionen und eq-Literalen
 - **RUE-Resolution**: Erzeugung von Paramodulationsgleichungen bei Bedarf

E-RESOLUTION

Resolution nach Einsetzung assoziierter Gleichungen

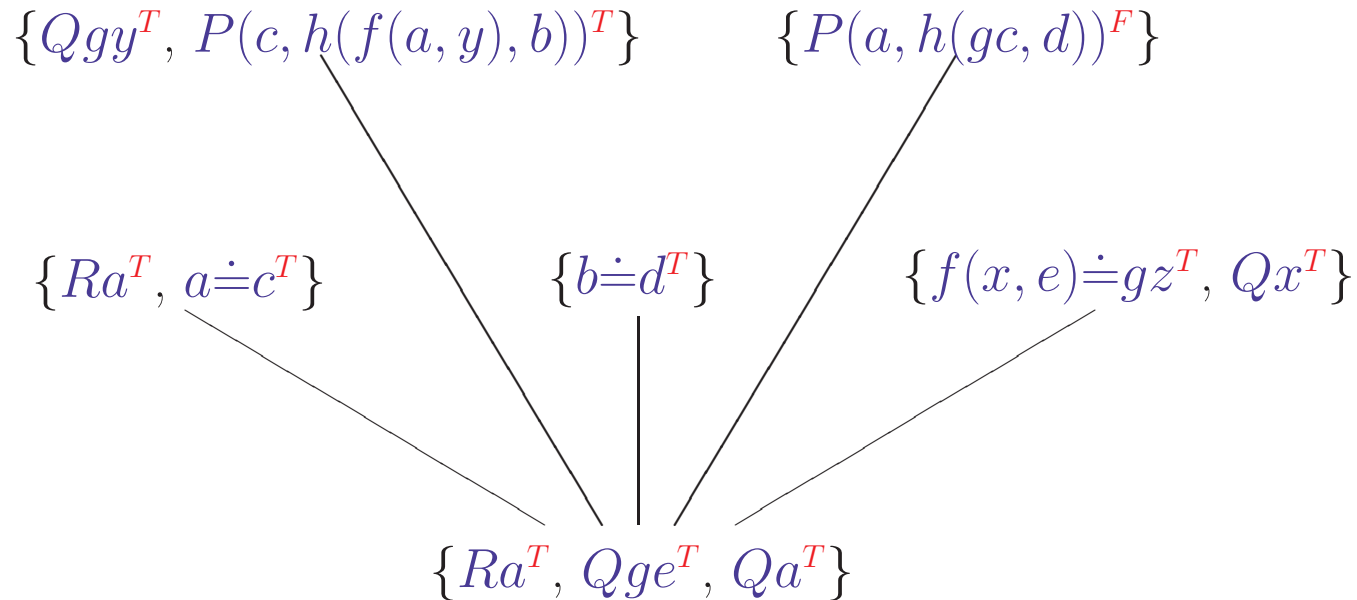
E-RESOLUTION

Resolution nach Einsetzung assoziierter Gleichungen



E-RESOLUTION

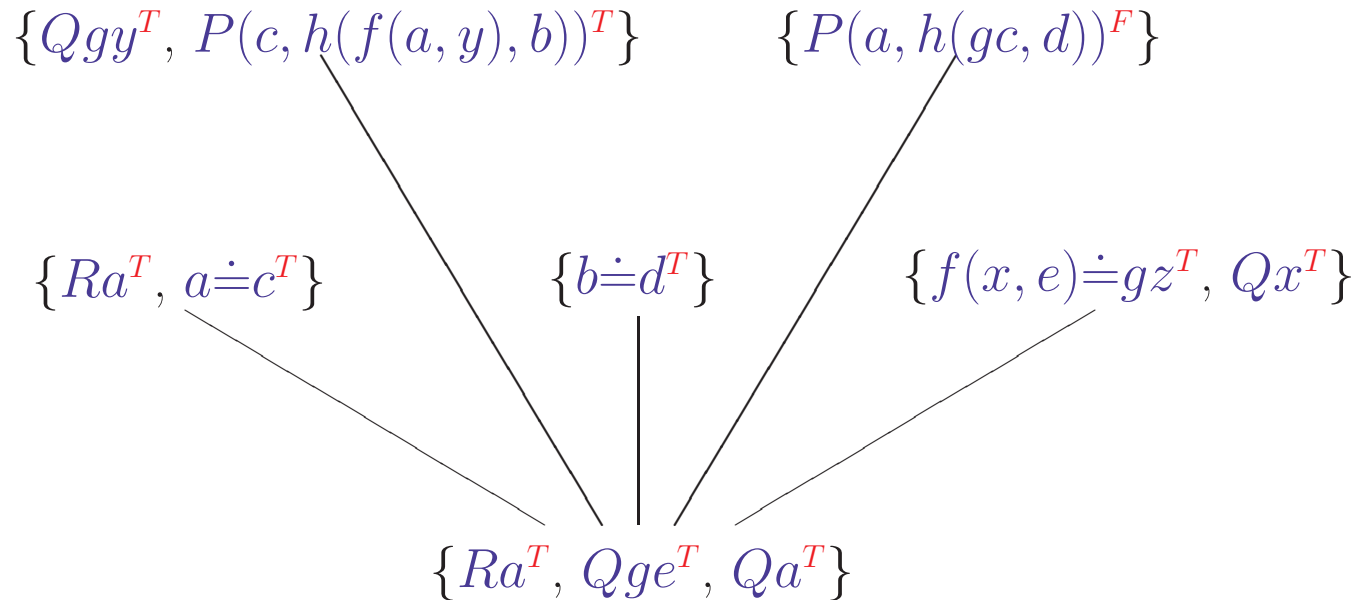
Resolution nach Einsetzung assoziierter Gleichungen



- Bedingungs-literale der Gleichungen bleiben in Resolvente

E-RESOLUTION

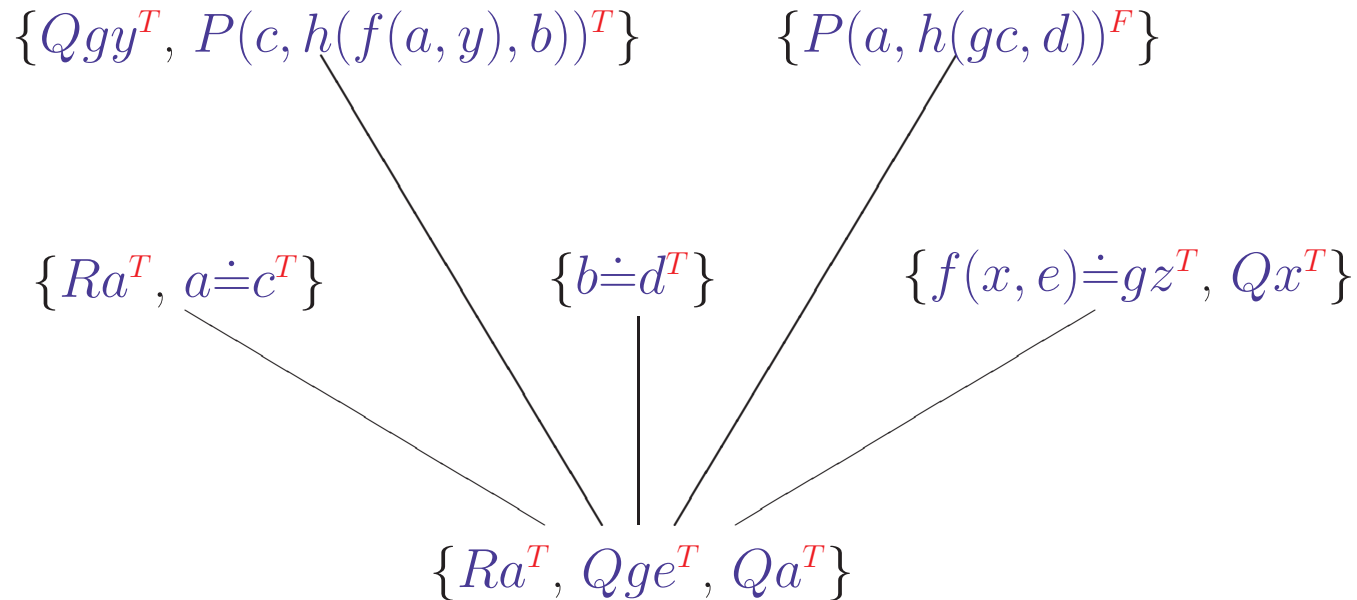
Resolution nach Einsetzung assoziierter Gleichungen



- Bedingungs-literale der Gleichungen bleiben in Resolvente
- Vollständig und korrekt

E-RESOLUTION

Resolution nach Einsetzung assoziierter Gleichungen



- **Bedingungs-literale der Gleichungen bleiben in Resolvente**
- **Vollständig und korrekt**
- **Durchführung aufwendig**
 - Suche nach geeigneten Gleichungen unentscheidbar

RUE-RESOLUTION

Unterteilung von E-Resolution in Teilschritte

RUE-RESOLUTION

Unterteilung von E-Resolution in Teilschritte

- **Resolution anwendbar auch ohne Unifizierbarkeit der Literale**
 - Verbleibende Gleichheitsbedingungen erscheinen in Resolvente
 - Gleichheitsbedingungen werden später verarbeitet

RUE-RESOLUTION

Unterteilung von E-Resolution in Teilschritte

- Resolution anwendbar auch ohne Unifizierbarkeit der Literale
 - Verbleibende Gleichheitsbedingungen erscheinen in Resolvente
 - Gleichheitsbedingungen werden später verarbeitet

$$\begin{array}{ccc} \{Qgy^T, P(c, h(f(a, y), b))^T\} & & \{P(a, h(gc, d))^F\} \\ & \searrow & \swarrow \\ & & \{Qgy^T, c \doteq a^F, fay \doteq gc^F, b \doteq d^F\} \end{array}$$