Rechnerübung zu Theorembeweiser und ihre Anwendungen

Prof. Dr.-Ing. Gregor Snelting
Dipl.-Inf. Univ. Daniel Wasserrab

Lehrstuhl Programmierparadigmen
IPD Snelting
Universität Karlsruhe (TH)

Teil II

Quantoren in Isabelle/HOL

Quantoren in Isabelle/HOL

Die üblichen zwei Quantoren der Logik:

Existenzquantor: $\exists$ (geschrieben \<exists>), Syntax: $\exists x.\ P$

Allquantor: $\forall$ (geschrieben \<forall>), Syntax: $\forall x.\ P$

Gültigkeitsbereich der gebundenen Variablen: bis zum nächsten ; bzw. $\Longrightarrow$

Beispiele:

$\forall x.\ P\ x \Longrightarrow Q\ x$: x in Konklusion nicht gebunden durch Allquantor

$P\ y \Longrightarrow \exists y.\ P\ y$: y in Prämisse nicht gebunden durch Existenzquantor

$[\![\forall x.\ P\ x;\ \exists x.\ Q\ x]\!] \Longrightarrow R$:

Zwei verschiedene x in den Annahmen

gleichbedeutend mit $[\![\forall y.\ P\ y;\ \exists z.\ Q\ z]\!] \Longrightarrow R$

(*gebundene Namen sind Schall und Rauch*)

$\forall x.\ P\ x \longrightarrow Q\ x$: *gleiches* x für P und Q

Quantoren in Isabelle/HOL

Die üblichen zwei Quantoren der Logik:

Existenzquantor: $\exists$ (geschrieben `\<exists>`), Syntax: $\exists x.\ P$

Allquantor: $\forall$ (geschrieben `\<forall>`), Syntax: $\forall x.\ P$

Gültigkeitsbereich der gebundenen Variablen: bis zum nächsten ; bzw. $\Longrightarrow$

Beispiele:

$\forall x.\ P\ x \Longrightarrow Q\ x$: x in Konklusion nicht gebunden durch Allquantor

$P\ y \Longrightarrow \exists y.\ P\ y$: y in Prämisse nicht gebunden durch Existenzquantor

$[\![\forall x.\ P\ x;\ \exists x.\ Q\ x]\!] \Longrightarrow R$:

Zwei verschiedene x in den Annahmen

gleichbedeutend mit $[\![\forall y.\ P\ y;\ \exists z.\ Q\ z]\!] \Longrightarrow R$

(*gebundene Namen sind Schall und Rauch*)

$\forall x.\ P\ x \longrightarrow Q\ x$: *gleiches* x für P und Q

Quantoren in Isabelle/HOL

Die üblichen zwei Quantoren der Logik:

Existenzquantor: $\exists$ (geschrieben `\<exists>`), Syntax: $\exists x.\ P$

Allquantor: $\forall$ (geschrieben `\<forall>`), Syntax: $\forall x.\ P$

Gültigkeitsbereich der gebundenen Variablen: bis zum nächsten ; bzw. $\Longrightarrow$
Beispiele:

$\forall x.\ P\ x \Longrightarrow Q\ x$: x in Konklusion nicht gebunden durch Allquantor

$P\ y \Longrightarrow \exists y.\ P\ y$: y in Prämisse nicht gebunden durch Existenzquantor

$[\![\forall x.\ P\ x;\ \exists x.\ Q\ x]\!] \Longrightarrow R$:
Zwei verschiedene x in den Annahmen
gleichbedeutend mit $[\![\forall y.\ P\ y;\ \exists z.\ Q\ z]\!] \Longrightarrow R$
(*gebundene Namen sind Schall und Rauch*)

$\forall x.\ P\ x \longrightarrow Q\ x$: *gleiches* x für P und Q

Quantoren in Isabelle/HOL

Die üblichen zwei Quantoren der Logik:

Existenzquantor: $\exists$ (geschrieben `\<exists>`), Syntax: $\exists x.\ P$

Allquantor: $\forall$ (geschrieben `\<forall>`), Syntax: $\forall x.\ P$

Gültigkeitsbereich der gebundenen Variablen: bis zum nächsten ; bzw. $\Longrightarrow$
Beispiele:

$\forall x.\ P\ x \Longrightarrow Q\ x$: x in Konklusion nicht gebunden durch Allquantor

$P\ y \Longrightarrow \exists y.\ P\ y$: y in Prämisse nicht gebunden durch Existenzquantor

$[\![\forall x.\ P\ x;\ \exists x.\ Q\ x]\!] \Longrightarrow R$:
Zwei verschiedene x in den Annahmen
gleichbedeutend mit $[\![\forall y.\ P\ y;\ \exists z.\ Q\ z]\!] \Longrightarrow R$
(*gebundene Namen sind Schall und Rauch*)

$\forall x.\ P\ x \longrightarrow Q\ x$: *gleiches* x für P und Q

Quantoren in Isabelle/HOL

Die üblichen zwei Quantoren der Logik:

Existenzquantor: $\exists$ (geschrieben \<exists>), Syntax: $\exists x.\ P$

Allquantor: $\forall$ (geschrieben \<forall>), Syntax: $\forall x.\ P$

Gültigkeitsbereich der gebundenen Variablen: bis zum nächsten ; bzw. $\Longrightarrow$
Beispiele:

$\forall x.\ P\ x \Longrightarrow Q\ x$: x in Konklusion nicht gebunden durch Allquantor

$P\ y \Longrightarrow \exists y.\ P\ y$: y in Prämisse nicht gebunden durch Existenzquantor

$[\![\forall x.\ P\ x;\ \exists x.\ Q\ x]\!] \Longrightarrow R$:
Zwei verschiedene x in den Annahmen
gleichbedeutend mit $[\![\forall y.\ P\ y;\ \exists z.\ Q\ z]\!] \Longrightarrow R$
(*gebundene Namen sind Schall und Rauch*)

$\forall x.\ P\ x \longrightarrow Q\ x$: *gleiches* x für P und Q

Quantoren in Isabelle/HOL

Die üblichen zwei Quantoren der Logik:

Existenzquantor: $\exists$ (geschrieben \<exists>), Syntax: $\exists x.\ P$

Allquantor: $\forall$ (geschrieben \<forall>), Syntax: $\forall x.\ P$

Gültigkeitsbereich der gebundenen Variablen: bis zum nächsten ; bzw. $\Longrightarrow$
Beispiele:

$\forall x.\ P\ x \Longrightarrow Q\ x$: x in Konklusion nicht gebunden durch Allquantor

$P\ y \Longrightarrow \exists y.\ P\ y$: y in Prämisse nicht gebunden durch Existenzquantor

$[\![\forall x.\ P\ x;\ \exists x.\ Q\ x]\!] \Longrightarrow R$:
Zwei verschiedene x in den Annahmen
gleichbedeutend mit $[\![\forall y.\ P\ y;\ \exists z.\ Q\ z]\!] \Longrightarrow R$
(*gebundene Namen sind Schall und Rauch*)

$\forall x.\ P\ x \longrightarrow Q\ x$: *gleiches* x für P und Q

Wie sagt man es Isabelle...?

Argumentation mit Quantoren erfordert Aussagen über *beliebige* Werte
Nur: wie weiss Isabelle, das ein Wert *beliebig* ist?

Wie sagt man es Isabelle...?

Argumentation mit Quantoren erfordert Aussagen über *beliebige* Werte
Nur: wie weiss Isabelle, das ein Wert *beliebig* ist?

Lösung: Meta-Logik!
x in einer Aussage beliebig: brauchen Quantor dafür
Syntax in Isabelle: $\bigwedge x.\ [\![\ldots]\!] \Longrightarrow \ldots$
$\bigwedge$ heisst Meta-Allquantor, Variablen dahinter Parameter
Gültigkeitsbereich der Parameter: ganzes Subgoal
Beispiel: $\bigwedge x\ y.\ [\![\forall y.\ P\ y \longrightarrow Q\ z\ y;\ Q\ x\ y]\!] \Longrightarrow \exists x.\ Q\ x\ y$
entspricht $\bigwedge x\ y.\ [\![\forall y_1.\ P\ y_1 \longrightarrow Q\ z\ y_1;\ Q\ x\ y]\!] \Longrightarrow \exists x_1.\ Q\ x_1\ y$

Wie sagt man es Isabelle...?

Argumentation mit Quantoren erfordert Aussagen über *beliebige* Werte
Nur: wie weiss Isabelle, das ein Wert *beliebig* ist?

Lösung: Meta-Logik!
x in einer Aussage beliebig: brauchen Quantor dafür
Syntax in Isabelle: $\bigwedge x.\ [\![\ldots]\!] \Longrightarrow \ldots$
$\bigwedge$ heisst Meta-Allquantor, Variablen dahinter Parameter
Gültigkeitsbereich der Parameter: ganzes Subgoal
Beispiel: $\bigwedge x\ y.\ [\![\forall y.\ P\ y \longrightarrow Q\ z\ y;\ Q\ x\ y]\!] \Longrightarrow \exists x.\ Q\ x\ y$
entspricht $\bigwedge x\ y.\ [\![\forall y_1.\ P\ y_1 \longrightarrow Q\ z\ y_1;\ Q\ x\ y]\!] \Longrightarrow \exists x_1.\ Q\ x_1\ y$

Auch $\Longrightarrow$ ist Teil der Meta-Logik, entspricht Meta-Implikation
Trennt Annahmen und Konklusion

Wie sagt man es Isabelle...?

Argumentation mit Quantoren erfordert Aussagen über *beliebige* Werte
Nur: wie weiss Isabelle, das ein Wert *beliebig* ist?

Lösung: Meta-Logik!
x in einer Aussage beliebig: brauchen Quantor dafür
Syntax in Isabelle: $\bigwedge x.\ [\![\ldots]\!] \Longrightarrow \ldots$
$\bigwedge$ heisst Meta-Allquantor, Variablen dahinter Parameter
Gültigkeitsbereich der Parameter: ganzes Subgoal
Beispiel: $\bigwedge x\ y.\ [\![\forall y.\ P\ y \longrightarrow Q\ z\ y;\ Q\ x\ y]\!] \Longrightarrow \exists x.\ Q\ x\ y$
entspricht $\bigwedge x\ y.\ [\![\forall y_1.\ P\ y_1 \longrightarrow Q\ z\ y_1;\ Q\ x\ y]\!] \Longrightarrow \exists x_1.\ Q\ x_1\ y$

Auch $\Longrightarrow$ ist Teil der Meta-Logik, entspricht Meta-Implikation
Trennt Annahmen und Konklusion

$\forall$ und $\longrightarrow$ entsprechen nicht $\bigwedge$ und $\Longrightarrow$, die ersten beiden nur in HOL!

Regeln

Jeder Quantor Introduktions- und Eliminationsregel:

- `allI: (⋀x. P x) ⟹ ∀x. P x`
  Eine Aussage gilt für beliebige x (Meta-Ebene),
  also gilt sie auch für alle (HOL-Ebene)
- `allE: ⟦∀x. P x; P ?x ⟹ R⟧ ⟹ R`
  Eine Aussage gilt für alle x, also folgt die Konklusion auch,
  wenn diese Aussage für irgendeine (selbst wählbare) Variable x gilt
  Vorsicht: x nach Anwendung der Regel beliebige Variable ($?x$)!
  Möglichst gleich spezifizieren duch `erule_tac`

Regeln

Jeder Quantor Introduktions- und Eliminationsregel:

- `allI: (⋀x. P x) ⟹ ∀x. P x`
  Eine Aussage gilt für beliebige x (Meta-Ebene),
  also gilt sie auch für alle (HOL-Ebene)
- `allE: ⟦∀x. P x; P ?x ⟹ R⟧ ⟹ R`
  Eine Aussage gilt für alle x, also folgt die Konklusion auch,
  wenn diese Aussage für irgendeine (selbst wählbare) Variable x gilt
  Vorsicht: x nach Anwendung der Regel beliebige Variable ($?x$)!
  Möglichst gleich spezifizieren duch `erule_tac`

Regeln

Jeder Quantor Introduktions- und Eliminationsregel:

- `allI: (⋀x. P x) ⟹ ∀x. P x`
  Eine Aussage gilt für beliebige x (Meta-Ebene),
  also gilt sie auch für alle (HOL-Ebene)
- `allE: ⟦∀x. P x; P ?x ⟹ R⟧ ⟹ R`
  Eine Aussage gilt für alle x, also folgt die Konklusion auch,
  wenn diese Aussage für irgendeine (selbst wählbare) Variable x gilt
  Vorsicht: x nach Anwendung der Regel beliebige Variable ($?x$)!
  Möglichst gleich spezifizieren duch `erule_tac`

Regeln

- *exI:* $P\ ?x \Longrightarrow \exists x.\ P\ x$
  Eine Aussage gilt für eine Variable x, also gibt es ein x, wofür sie gilt
  Vorsicht: x nach Anwendung der Regel beliebige Variable ($?x$)!
  Möglichst gleich spezifizieren duch *rule_tac*
- *exE:* $[\![\exists x.\ P\ x;\ \bigwedge x.\ P\ x \Longrightarrow Q]\!] \Longrightarrow Q$
  Eine Aussage gilt für ein x, also folgt die Konklusion auch,
  wenn diese Aussage für eine beliebige (vorgegebene!) Variable gilt.

Regeln

- *exI:* $P\ ?x \Longrightarrow \exists x.\ P\ x$
  Eine Aussage gilt für eine Variable x, also gibt es ein x, wofür sie gilt
  Vorsicht: x nach Anwendung der Regel beliebige Variable ($?x$)!
  Möglichst gleich spezifizieren duch *rule_tac*
- *exE:* $[\![\exists x.\ P\ x;\ \bigwedge x.\ P\ x \Longrightarrow Q]\!] \Longrightarrow Q$
  Eine Aussage gilt für ein x, also folgt die Konklusion auch,
  wenn diese Aussage für eine beliebige (vorgegebene!) Variable gilt.

Variablen festlegen bei Regelanwendung

Den Regeln *allE* und *exI* gemeinsam:
nach Anwendung der entsprechenden Methodik (also *erule* bzw. *rule*)
unspezifizierte Variablen ($?x$) in Subgoal
meist nicht gewollt, da schlecht Aussagen darüber möglich

Besser: entsprechende Variable gleich festlegen

Variablen festlegen bei Regelanwendung

Den Regeln *allE* und *exI* gemeinsam:
nach Anwendung der entsprechenden Methodik (also *erule* bzw. *rule*)
unspezifizierte Variablen (?x) in Subgoal
meist nicht gewollt, da schlecht Aussagen darüber möglich

Besser: entsprechende Variable gleich festlegen
Methodik: *rule_tac v1 = t1* **and** ... **and** *vk = tk* **in** *R*,
?v1, ..., *?vk* freie Variable in der anzuwendenden Regel *R*
(nicht im aktuellen Subgoal!)
analog: *erule_tac*

Variablen festlegen bei Regelanwendung

Den Regeln *allE* und *exI* gemeinsam:
nach Anwendung der entsprechenden Methodik (also *erule* bzw. *rule*)
unspezifizierte Variablen (?x) in Subgoal
meist nicht gewollt, da schlecht Aussagen darüber möglich

Besser: entsprechende Variable gleich festlegen
Methodik: *rule_tac v1 = t1* **and** ... **and** *vk = tk* **in** *R*,
?v1, ..., *?vk* freie Variable in der anzuwendenden Regel *R*
(nicht im aktuellen Subgoal!)
analog: *erule_tac*

also möglichst immer **apply**(*rule_tac x=gewollte Variable* **in** *exI*)
bzw. **apply**(*erule_tac x=gewollte Variable* **in** *allE*)