Alle Gleichheiten sind gleich, aber manche sind gleicher

From Analysis to Homotopy Theory
A conference in honor of Ulrich Bunke's 60th birthday
Mai 2024, Alfried Krupp Wissenschaftskolleg, Greifswald

Clara Löh
Fakultät für Mathematik. Universität Regensburg

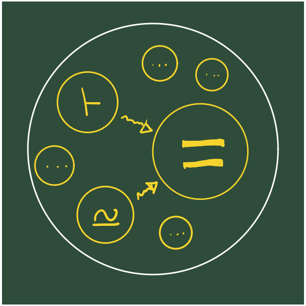
Universität Regensburg

$$x = x$$

Ende

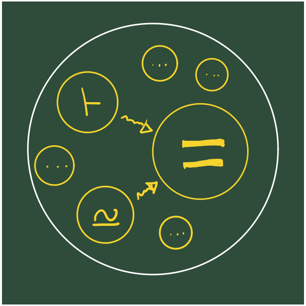
Überblick

Mathematik



Überblick

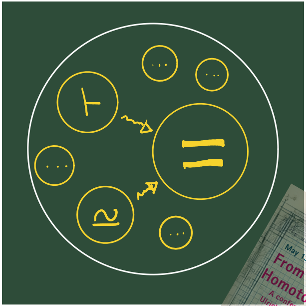
Mathematik



Homotopietheorie
Typtheorie

Überblick

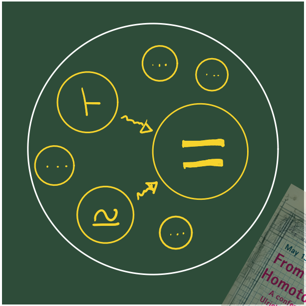
Mathematik



Homotopietheorie
Typtheorie

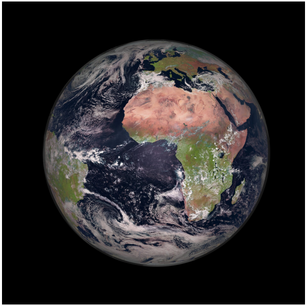
Überblick

Mathematik



Homotopietheorie
Typtheorie

Anwendung



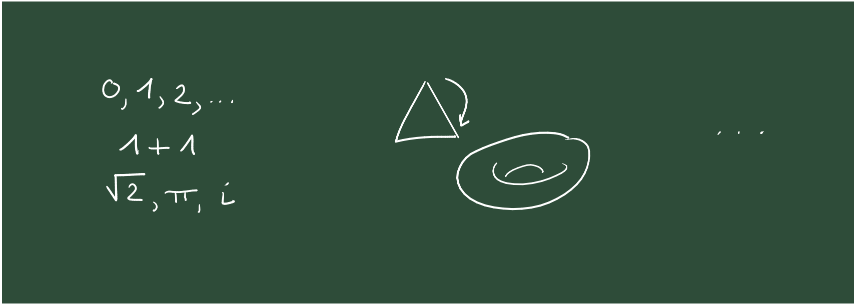
©ESA: EUMETSAT/ESA

Computerverifikation

Was ist Mathematik?

Was ist Mathematik?

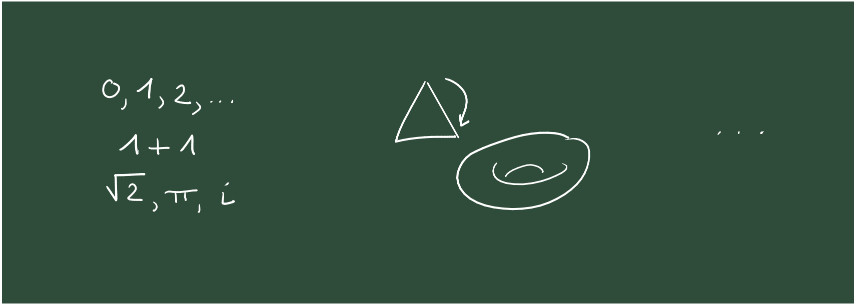
3. Mathematische Ideen:



Was ist Mathematik?

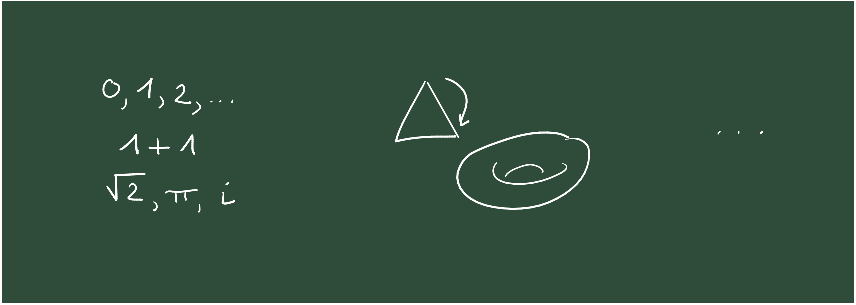
1. Syntax: Beweiskalkül, d.h.:
   - Formale Sprache und
   - Regeln, wie Elemente dieser Sprache manipuliert werden dürfen.

3. Mathematische Ideen:



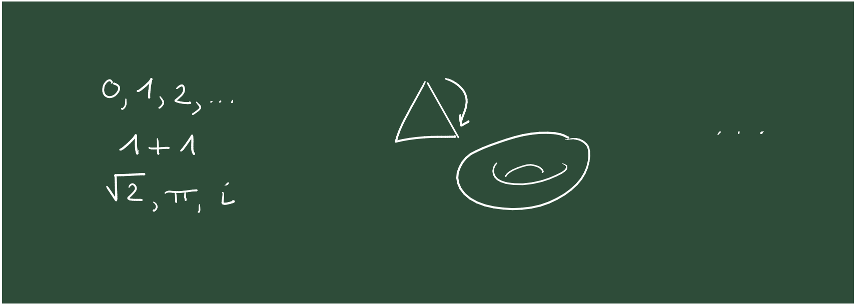
Was ist Mathematik?

1. Syntax: Beweiskalkül, d.h.:
   - Formale Sprache und
   - Regeln, wie Elemente dieser Sprache manipuliert werden dürfen.
2. Semantik: Interpretation dieser formalen Ebene
3. Mathematische Ideen:



Was ist Mathematik?

1. Syntax: Beweiskalkül, d.h.:
   - Formale Sprache und
   - Regeln, wie Elemente dieser Sprache manipuliert werden dürfen.
2. Semantik: Interpretation dieser formalen Ebene
3. Mathematische Ideen:



Da 1./2. sehr abstrakt sind: Vergleich mit ähnlichen Situationen

Was ist Musik?

Was ist Musik?

3. Musikalische Ideen: . . .

Was ist Musik?

1. Syntax: Notensysteme

F. Schubert, Op. 142.2 (transp)

3. Musikalische Ideen: ...

Was ist Musik?

1. Syntax: Notensysteme

F. Schubert, Op. 142.2 (transp)

2. Semantik: Spielen der Noten auf Instrumenten

3. Musikalische Ideen: . . .

Was ist Musik?

1. Syntax: Notensysteme

F. Schubert, Op. 142.2 (transp)

2. Semantik: Spielen der Noten auf Instrumenten

Verschiedene Instrumente bieten verschiedene Möglichkeiten
und diese sind teilweise nicht in jedem Notensystem vorgesehen!

3. Musikalische Ideen: . . .

Was ist Programmierung?

Was ist Programmierung?

3. Programmierideen: Anwendungssoftware

©Nintendo

Was ist Programmierung?

1. Syntax: Programmiersprachen

3. Programmierideen: Anwendungssoftware

Was ist Programmierung?

1. Syntax: Programmiersprachen
2. Semantik:
   - Übersetzung in Maschinensprache
   - und Ausführung von Programmen auf Computern
3. Programmierideen: Anwendungssoftware

Nochmal: Was ist Mathematik?

3. Mathematische Ideen: natürliche Zahlen, Geometrie, Gleichheit, ...

Nochmal: Was ist Mathematik?

1. Syntax:

   klassisch
   Prädikatenlogik
   Mengenlehre

3. Mathematische Ideen: natürliche Zahlen, Geometrie, Gleichheit, ...

Nochmal: Was ist Mathematik?

1. Syntax:

   klassisch
   Prädikatenlogik
   Mengenlehre

2. Semantik:

   klassisch
   Modelle der Mengenlehre

3. Mathematische Ideen: natürliche Zahlen, Geometrie, Gleichheit, ...

Nochmal: Was ist Mathematik?

1. Syntax:

   klassisch
   Prädikatenlogik
   Mengenlehre

   typtheoretisch
   Terme, Typen, Typregeln

2. Semantik:

   klassisch
   Modelle der Mengenlehre

   typtheoretisch
   Modelle der Typtheorie
   z.B. homotopietheoretisch

3. Mathematische Ideen: natürliche Zahlen, Geometrie, Gleichheit, ...

Gleichheit

$0 = 0$

- Reflexivität:
  Für jedes x gilt $x = x$.

Gleichheit

$0 = 0$

$0 = 11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23$

$0 = 0 \cdot z \quad$ für alle ganzen Zahlen z

- Reflexivität:
  Für jedes x gilt $x = x$.

Gleichheit

$0 = 0$

$0 = 11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23$

$0 = 0 \cdot z$ für alle ganzen Zahlen z

$0 = (11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23) \cdot z$ für alle ganzen Zahlen z

- Reflexivität:
  Für jedes x gilt $x = x$.
- Substitutionsprinzip:
  Ist $x = y$, so kann man jedes Auftreten von x durch y ersetzen.

Gleichheit

$$0 = 0$$

$$0 = 11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23$$

$$0 = 0 \cdot z \quad \text{für alle ganzen Zahlen } z$$

$$0 = (11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23) \cdot z \quad \text{für alle ganzen Zahlen } z$$

$$0 = \int_{-2024}^{2024} x^{60} \cdot \sin x \, dx$$

$$0 = \text{Anzahl aller natürlichen Zahlen } n \geq 3,$$
$$\text{sodass es teilerfremde ganze Zahlen } x, y, z \text{ mit } x^n + y^n = z^n \text{ gibt}$$

- Reflexivität:
  Für jedes x gilt $x = x$.
- Substitutionsprinzip:
  Ist $x = y$, so kann man jedes Auftreten von x durch y ersetzen.

Gleichheit

$$0 = 0$$

$$0 = 11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23$$

$$0 = 0 \cdot z \quad \text{für alle ganzen Zahlen } z$$

$$0 = (11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23) \cdot z \quad \text{für alle ganzen Zahlen } z$$

$$0 = \int_{-2024}^{2024} x^{60} \cdot \sin x \, dx$$

$$0 = \text{Anzahl aller natürlichen Zahlen } n \geq 3,$$
$$\text{sodass es teilerfremde ganze Zahlen } x, y, z \text{ mit } x^n + y^n = z^n \text{ gibt}$$

- Reflexivität:
  Für jedes x gilt $x = x$.
- Substitutionsprinzip:
  Ist $x = y$, so kann man jedes Auftreten von x durch y ersetzen.

Achtung! Hängt vom Kontext ab! Z.B.: „Welche Termart?“

Gleichheit: Interessanter: „Warum?“ statt „Ob?“

Warum gilt für jede ganze Zahl z, dass $0 = (11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23) \cdot z$?

Gleichheit: Interessanter: „Warum?“ statt „Ob?“

Warum gilt für jede ganze Zahl z, dass $0 = (11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23) \cdot z$?

Beweis. Sei z eine ganze Zahl. Dann gilt

$$\begin{aligned}(11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23) \cdot z &= (2024 - 2024) \cdot z \\ &= 0 \cdot z \\ &= 0.\end{aligned}$$

□

Gleichheit: Interessanter: „Warum?“ statt „Ob?“

Warum gilt für jede ganze Zahl z, dass $0 = (11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23) \cdot z$?

Beweis. Sei z eine ganze Zahl. Dann gilt

$$\begin{aligned}(11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23) \cdot z &= (2024 - 2024) \cdot z \\ &= 0 \cdot z \\ &= 0. \end{aligned}$$

$\square$

Alternativ:

Beweis. Sei z eine ganze Zahl. Dann gilt:

$$\begin{aligned}(11 \cdot 184 - 2 \cdot 2 \cdot 22 \cdot 23) \cdot z &= 11 \cdot 184 \cdot z - 2 \cdot 2 \cdot 22 \cdot 23 \cdot z \\ &= 2024 \cdot z - 2024 \cdot z \\ &= 0. \end{aligned}$$

$\square$

Gleichheit: Der Schlüssel zu anderen Interpretationen

Klassische Interpretation:

$$x = y \quad \text{ist} \quad \begin{cases} \textbf{wahr} & \text{oder} \\ \textbf{falsch} & \end{cases}$$

Gleichheit: Der Schlüssel zu anderen Interpretationen

Klassische Interpretation:

$$x = y \quad \text{ist} \quad \begin{cases} \textbf{wahr} & \text{oder} \\ \textbf{falsch} \end{cases}$$

ist die Sammlung nicht-leer?

Idee für eine andere Interpretation:

$x = y$ ist die **Sammlung aller Beweise**, dass x und y gleich sind

Gleichheit: Der Schlüssel zu anderen Interpretationen

Klassische Interpretation:

$$x = y \quad \text{ist} \quad \begin{cases} \textbf{wahr} & \text{oder} \\ \textbf{falsch} & \end{cases}$$

ist die Sammlung nicht-leer?

Idee für eine andere Interpretation:

$x = y$ ist die **Sammlung aller Beweise**, dass x und y gleich sind

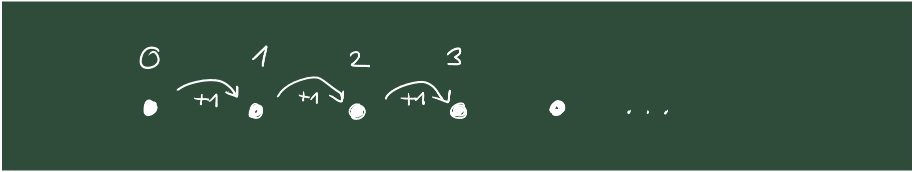
Umsetzung: Benötigen zunächst eine Formalisierung von Gleichheit.

Formalisierung von Gleichheit

Vorübung: Formalisierung der natürlichen Zahlen

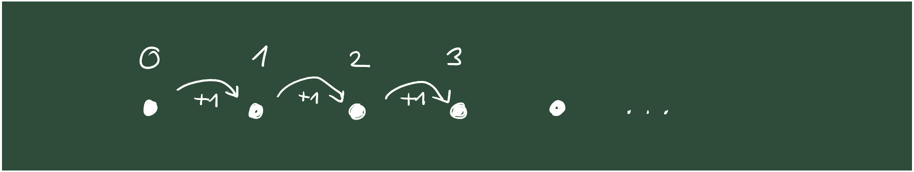
Formalisierung von Gleichheit

Vorübung: Formalisierung der natürlichen Zahlen



Formalisierung von Gleichheit

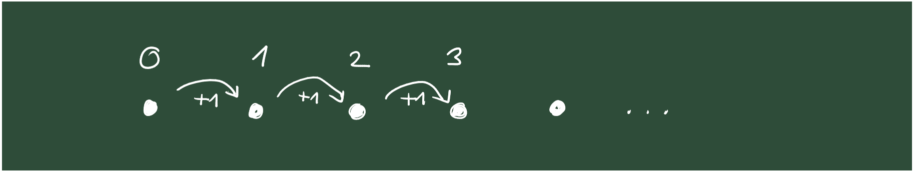
Vorübung: Formalisierung der natürlichen Zahlen



- Typdeklarationen:
  - Nat ist ein Typ.

Formalisierung von Gleichheit

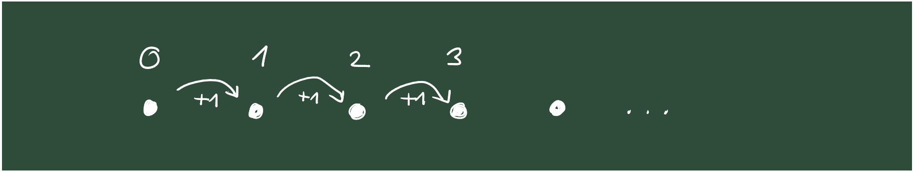
Vorübung: Formalisierung der natürlichen Zahlen



- Typdeklarationen:
  - Nat ist ein Typ.
- Intro:
  - 0 ist vom Typ Nat.
  - Ist x vom Typ Nat, so ist $S(x)$ vom Typ Nat.

Formalisierung von Gleichheit

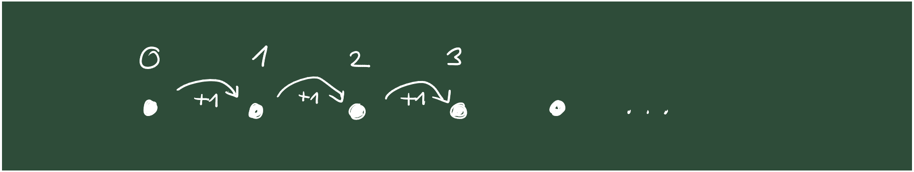
Vorübung: Formalisierung der natürlichen Zahlen



- Typdeklarationen:
  - Nat ist ein Typ.
- Intro:
  - 0 ist vom Typ Nat.
  - Ist x vom Typ Nat, so ist $S(x)$ vom Typ Nat.
- Rekursionsprinzip: Um eine Funktion f auf Nat zu definieren, genügt es
  - den Wert von f für 0 zu definieren und
  - für x vom Typ Nat anzugeben,
    wie man den Wert für $S(x)$ aus dem Wert für x erhält.

Formalisierung von Gleichheit

Vorübung: Formalisierung der natürlichen Zahlen



- Typdeklarationen:
  - Nat ist ein Typ.
- Intro:
  - 0 ist vom Typ Nat.
  - Ist x vom Typ Nat, so ist $S(x)$ vom Typ Nat.
- Rekursionsprinzip: Um eine Funktion f auf Nat zu definieren, genügt es
  - den Wert von f für 0 zu definieren und
  - für x vom Typ Nat anzugeben,
    wie man den Wert für $S(x)$ aus dem Wert für x erhält.

Klassische Interpretation: Nat ist die Menge der natürlichen Zahlen,
0 ist die Null, S ist die Abbildung „$+1$“.

Formalisierung von Gleichheit

Erinnerung: Idee:

$x = y$ ist die **Sammlung aller Beweise**,
dass x und y gleich sind

Formalisierung von Gleichheit

Erinnerung: Idee:

$x = y$ ist die **Sammlung aller Beweise**,
dass x und y gleich sind

- Typdeklarationen:
  - Ist X ein Typ und sind x und y vom Typ X,
    so ist $x =_X y$ ein Typ.

Formalisierung von Gleichheit

Erinnerung: Idee:

$x = y$ ist die **Sammlung aller Beweise**,
dass x und y gleich sind

- Typdeklarationen:
  - Ist X ein Typ und sind x und y vom Typ X,
    so ist $x =_X y$ ein Typ.
- Intro: Ist X ein Typ und x vom Typ X, so ist $\mathrm{refl}_X x$ vom Typ $x =_X x$.

Formalisierung von Gleichheit

Erinnerung: Idee:

$x = y$ ist die **Sammlung aller Beweise**,
dass x und y gleich sind

- Typdeklarationen:
  - Ist X ein Typ und sind x und y vom Typ X,
    so ist $x =_X y$ ein Typ.
- Intro: Ist X ein Typ und x vom Typ X, so ist $\mathrm{refl}_X x$ vom Typ $x =_X x$.
- Rekursionsprinzip: Um eine Funktion f auf allen Gleichheitstypen zu definieren, genügt es,
  - für jeden Typ X und jedes x vom Typ X,
    den Wert auf $\mathrm{refl}_X x$ zu definieren.

Formalisierung von Gleichheit

Erinnerung: Idee:

$x = y$ ist die **Sammlung aller Beweise**,
dass x und y gleich sind

- Typdeklarationen:
  - Ist X ein Typ und sind x und y vom Typ X,
    so ist $x =_X y$ ein Typ.
- Intro: Ist X ein Typ und x vom Typ X, so ist $\mathrm{refl}_X x$ vom Typ $x =_X x$.
- Rekursionsprinzip: Um eine Funktion f auf allen Gleichheitstypen zu definieren, genügt es,
  - für jeden Typ X und jedes x vom Typ X,
    den Wert auf $\mathrm{refl}_X x$ zu definieren.

Aus diesem Rekursionsprinzip folgt das Substitutionsprinzip für Gleichheit.

Klassische Interpretation von Gleichheit

- Typen interpretieren wir als: Mengen
  „ist vom Typ“ interpretieren wir als: „ist Element von“
  Außerdem: Mengen sind im mengentheoretischen Sinn gleich, wenn sie dieselben Elemente enthalten.

Klassische Interpretation von Gleichheit

- Typen interpretieren wir als: Mengen
  „ist vom Typ“ interpretieren wir als: „ist Element von“
  Außerdem: Mengen sind im mengentheoretischen Sinn gleich, wenn sie dieselben Elemente enthalten.
- Sei X eine Menge und seien x und y Elemente von X.
  Dann interpretieren wir $x =_X y$ als die Menge

$$\begin{cases} \{*\} & \text{falls } x \text{ und } y \text{ im mengentheoretischen Sinn gleich sind} \\ \{\} & \text{sonst} \end{cases}$$

Klassische Interpretation von Gleichheit

- Typen interpretieren wir als: Mengen
  „ist vom Typ“ interpretieren wir als: „ist Element von“
  Außerdem: Mengen sind im mengentheoretischen Sinn gleich, wenn sie dieselben Elemente enthalten.
- Sei X eine Menge und seien x und y Elemente von X.
  Dann interpretieren wir $x =_X y$ als die Menge

$$\begin{cases} \{*\} & \text{falls } x \text{ und } y \text{ im mengentheoretischen Sinn gleich sind} \\ \{\} & \text{sonst} \end{cases}$$

- Ist X eine Menge und x ein Element von X,
  so interpretieren wir $\mathrm{refl}_X x$ als das Element

$$* \quad \text{in} \quad \{*\}$$

Klassische Interpretation von Gleichheit

- Typen interpretieren wir als: Mengen
  „ist vom Typ" interpretieren wir als: „ist Element von"
  Außerdem: Mengen sind im mengentheoretischen Sinn gleich, wenn sie dieselben Elemente enthalten.
- Sei X eine Menge und seien x und y Elemente von X.
  Dann interpretieren wir $x =_X y$ als die Menge

$$\begin{cases} \{*\} & \text{falls } x \text{ und } y \text{ im mengentheoretischen Sinn gleich sind} \\ \{\} & \text{sonst} \end{cases}$$

- Ist X eine Menge und x ein Element von X,
  so interpretieren wir $\mathrm{refl}_X x$ als das Element

$$* \quad \text{in} \quad \{*\}$$

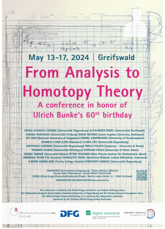
- In dieser Interpretation ist das Rekursionsprinzip für Gleichheit erfüllt.

Homotopietheoretische Interpretation von Gleichheit

Es gibt weitere Interpretationen von Typtheorie und Gleichheit!
Zum Beispiel: Das homotopietheoretische Modell.

Homotopietheoretische Interpretation von Gleichheit

Es gibt weitere Interpretationen von Typtheorie und Gleichheit!
Zum Beispiel: Das homotopietheoretische Modell.

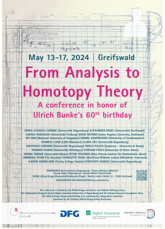


Homotopietheorie:

- „Geometrie“ ohne Längen/Winkel
- bis auf Homotopie, d.h. bis auf „stetige Deformation“ von Objekten und Transformationen.

Homotopietheoretische Interpretation von Gleichheit

Es gibt weitere Interpretationen von Typtheorie und Gleichheit!
Zum Beispiel: Das homotopietheoretische Modell.



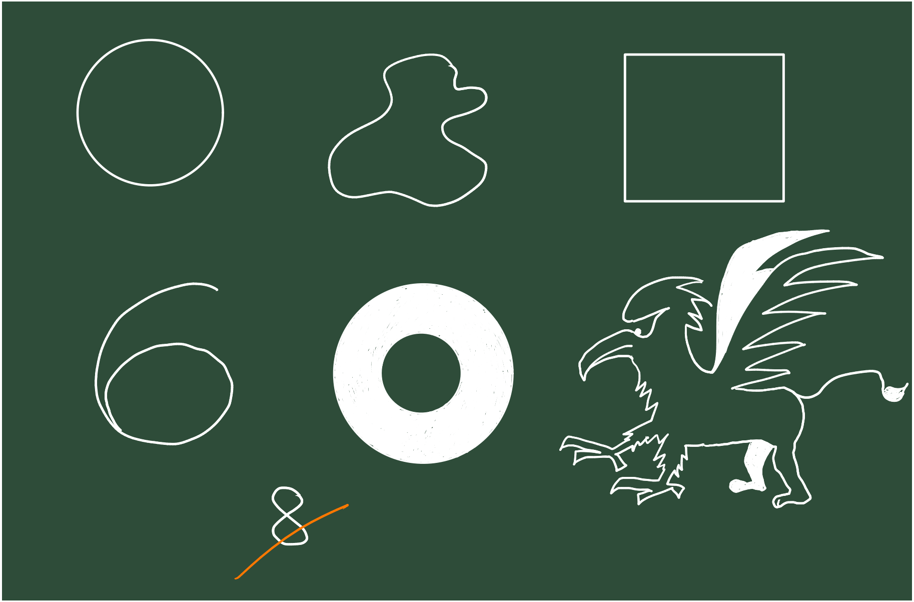
Homotopietheorie:

- „Geometrie“ ohne Längen/Winkel
- bis auf Homotopie, d.h. bis auf „stetige Deformation“ von Objekten und Transformationen.

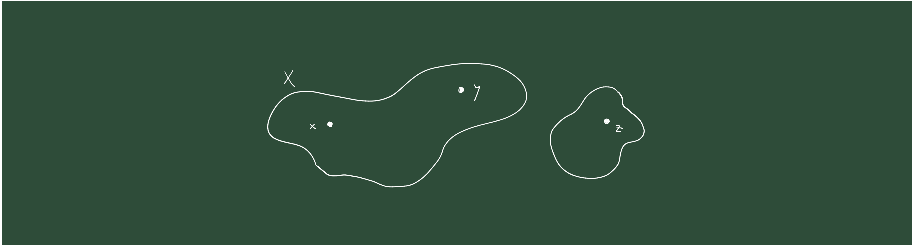
Anwendungen der Homotopietheorie:

- algebraische Topologie, globale Analysis, algebraische Geometrie, ...
- topologische Datenanalyse, Beweisassistenten, ...

Homotopietheorie



Homotopietheoretische Interpretation von Gleichheit



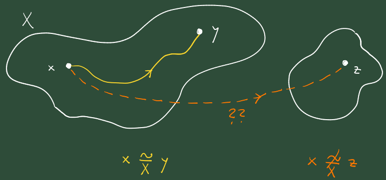
- Typen interpretieren wir als: topologische Räume
  „ist vom Typ“ interpretieren wir als: „ist Punkt in“

Homotopietheoretische Interpretation von Gleichheit

- Typen interpretieren wir als: topologische Räume
  „ist vom Typ“ interpretieren wir als: „ist Punkt in“
- Sei X ein topologischer Raum und seien x und y Punkte in X.
  Interpretation von $x =_X y$:

  Raum aller stetigen Wege von x nach y in X

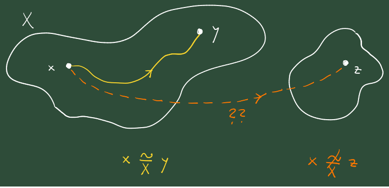
Homotopietheoretische Interpretation von Gleichheit



- Typen interpretieren wir als: topologische Räume
  „ist vom Typ“ interpretieren wir als: „ist Punkt in“
- Sei X ein topologischer Raum und seien x und y Punkte in X.
  Interpretation von $x =_X y$:

  Raum aller stetigen Wege von x nach y in X

Homotopietheoretische Interpretation von Gleichheit



- Typen interpretieren wir als: topologische Räume
  „ist vom Typ" interpretieren wir als: „ist Punkt in"
- Sei X ein topologischer Raum und seien x und y Punkte in X.
  Interpretation von $x =_X y$:

  Raum aller stetigen Wege von x nach y in X

- Interpretation von $\mathrm{refl}_X x$:

  der konstante Weg von x nach x

Vergleich der Modelle

Vergleich der Modelle

klassische Interpretation

alle Gleichheiten sind gleich!

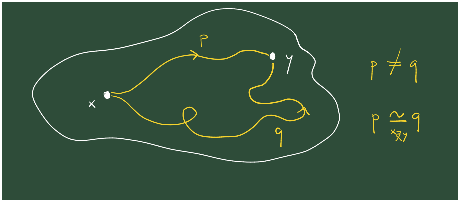
Vergleich der Modelle

klassische Interpretation

alle Gleichheiten sind gleich!

homotopietheoretische Interpretation

nicht alle Gleichheiten sind gleich!



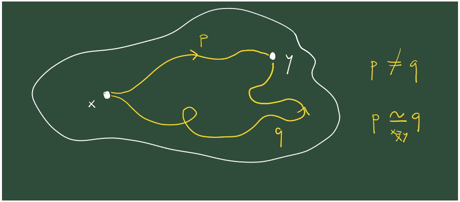
Vergleich der Modelle

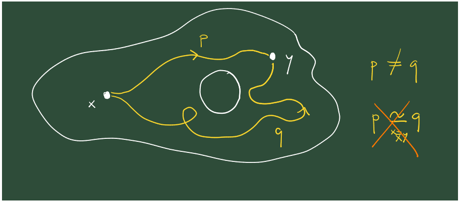
klassische Interpretation

alle Gleichheiten sind gleich!

homotopietheoretische Interpretation

nicht alle Gleichheiten sind gleich!





Wozu diese exotischen Modelle?

Die Behandlung von Gleichheit ist zentral

- in der Programmierung:
  z.B. Gleichheit von Pointern vs. strukturelle Gleichheit,
  Verallgemeinerung von Konzepten auf andere Kontexte
  (Substitutionsprinzip!)

Wozu diese exotischen Modelle?

Die Behandlung von Gleichheit ist zentral

- in der Programmierung:
  z.B. Gleichheit von Pointern vs. strukturelle Gleichheit, Verallgemeinerung von Konzepten auf andere Kontexte (Substitutionsprinzip!)
- in der Mathematik:
  wie kann man Strukturen robust und effizient identifizieren?
  z.B. ist $\mathbb{N} = \{x \in \mathbb{Z} \mid x \geq 0\}$?!

Wozu diese exotischen Modelle?

Die Behandlung von Gleichheit ist zentral

- in der Programmierung:
  z.B. Gleichheit von Pointern vs. strukturelle Gleichheit, Verallgemeinerung von Konzepten auf andere Kontexte (Substitutionsprinzip!)
- in der Mathematik:
  wie kann man Strukturen robust und effizient identifizieren?
  z.B. ist $\mathbb{N} = \{x \in \mathbb{Z} \mid x \geq 0\}$?!

Wunschzettel für

- geschickte Abstraktion/Verallgemeinerung: flexible Gleichheit
- Robustheit: starre Gleichheit

Interessante Balance: homotopietheoretische Modelle für Gleichheit
Konsequente Fortführung: Univalenz

Computerverifikation mithilfe von Beweisassistenten

- Ziel: Formalisierung von

  mathematischen Definitionen, Sätzen, Beweisen

  in einer geeigneten Programmiersprache,

  so dass Korrektheit von Beweisen maschinell zertifiziert werden kann.

  Es geht zunächst *nicht* um das Finden,

  sondern „nur" um das Überprüfen von Beweisen!

Computerverifikation mithilfe von Beweisassistenten

- Ziel: Formalisierung von

  mathematischen Definitionen, Sätzen, Beweisen

  in einer geeigneten Programmiersprache,
  so dass Korrektheit von Beweisen maschinell zertifiziert werden kann.
  Es geht zunächst *nicht* um das Finden,
  sondern „nur“ um das Überprüfen von Beweisen!
- Warum ist Computerverifikation interessant?
  - innermathematisch: Verifikation sehr komplexer Beweise
    Vierfarbensatz, algebraische Geometrie, LiquidTensor-Experiment, ...
  - außermathematisch: Analyse komplexer Systeme
    Verifikation sicherheitskritischer Hardware/Software,
    als unterliegendes rationales Schlussmodell für zukünftige KI-Systeme

Computerverifikation mithilfe von Beweisassistenten

- Ziel: Formalisierung von

  mathematischen Definitionen, Sätzen, Beweisen

  in einer geeigneten Programmiersprache,
  so dass Korrektheit von Beweisen maschinell zertifiziert werden kann.
  Es geht zunächst *nicht* um das Finden,
  sondern „nur“ um das Überprüfen von Beweisen!
- Warum ist Computerverifikation interessant?
  - innermathematisch: Verifikation sehr komplexer Beweise
    Vierfarbensatz, algebraische Geometrie, LiquidTensor-Experiment, ...
  - außermathematisch: Analyse komplexer Systeme
    Verifikation sicherheitskritischer Hardware/Software,
    als unterliegendes rationales Schlussmodell für zukünftige KI-Systeme
- Beispiele für Beweisassistenten:
  Coq (insbesondere das von Voevodsky initiierte UniMath-Projekt),
  Cubical Agda,
  Isabelle (nicht homotopietheoretisch),
  Lean 4 (nicht homotopietheoretisch), ...

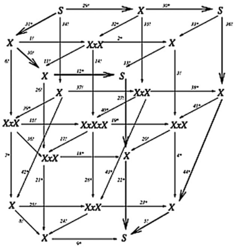
Wozu neue Grundlagen und Computerverifikation in der Mathematik?

The Origins and Motivations of Univalent Foundations

A Personal Mission to Develop Computer Proof Verification to Avoid Mathematical Mistakes

Vladimir Voevodsky • Published 2014

Email Share Share Tweet

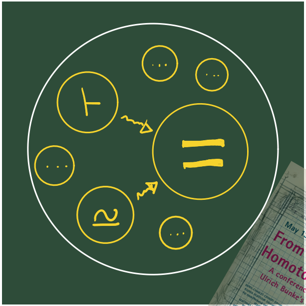


For the convenience of further reference we numbered all the arrows. The right vertical face of the diagram is the diagram (2) defining the 2-morphism $Id \to \Omega\Sigma$ and the upper horizontal face is the diagram (1) defining the 2-morphism $\Sigma\Omega \to Id$. The whole diagram is the union of the front part which

This three-dimensional diagram is an example of the kind of "formulas" that Voevodsky would have to use to support his arguments about 2-theories.

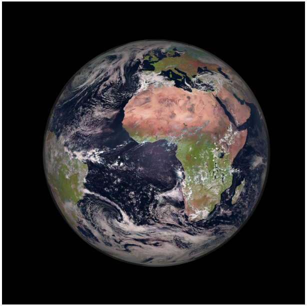
Zusammenfassung

Mathematik



Homotopietheorie
Typtheorie

Anwendung



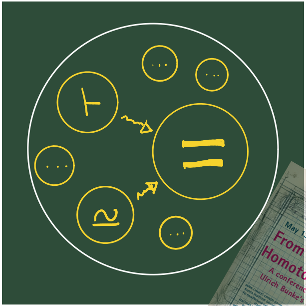
©ESA: EUMETSAT/ESA

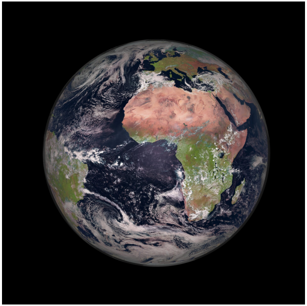
Computerverifikation

Zusammenfassung

Mathematik

Anwendung





©ESA: EUMETSAT/ESA

Homotopietheorie
Typtheorie

Computerverifikation

Beispiel dafür, dass theoretische Mathematik (z.B. Homotopietheorie) unerwartet hilft, praktische Probleme zu lösen (z.B. Computerverifikation) und umgekehrt

$$x = x$$

Ende

Literatur

- V. Voevodsky.
  The Origins and Motivations of Univalent Foundations. A Personal Mission to Develop Computer Proof Verification to Avoid Mathematical Mistakes, 2014
  https://www.ias.edu/ideas/2014/voevodsky-origins
  https://www.math.ias.edu/vladimir/sites/math.ias.edu.vladimir/files/2014_IAS.pdf
- V. Voevodsky.
  How I became interested in foundations of mathematics, 2014
  https://www.math.ias.edu/vladimir/sites/math.ias.edu.vladimir/files/2014_08_ASC_lecture.pdf
- D. R. Grayson.
  An introduction to univalent foundations for mathematicians,
  Bull. Am. Math. Soc., New Ser. 55, No. 4, 427-450, 2018.
- nLab.
  identity type
  https://ncatlab.org/nlab/show/identity+type