

# GeoGebra Automatisierte Begründungswerkzeuge

## Ein Tutorial

Zoltán Kovács, Tomás Recio und M. Pilar Vélez  
Übersetzt von Katharina Schiffler

16. Februar 2018

## 1 Einführung

Das Softwaretool GeoGebra (<http://www.geogebra.org>) ist geeignet zur Unterstützung des Unterrichts der zweidimensionalen euklidischen Geometrie Theorien mit symbolischen Berechnungen. Zu diesem Zweck wurden einige Werkzeuge für den automatischen Beweis und zur Entdeckung von geometrischen Lehrsätzen entwickelt.

Die neuen Technologien im Klassenzimmer befinden sich noch in einer experimentellen Phase. Dieses Dokument fasst die technischen Möglichkeiten zusammen und zeigt auch einige Beispiele.

## 2 GeoGebra starten

GeoGebra ist auf vielen Plattformen verfügbar, wie zum Beispiel

- Computer mit verschiedenen Betriebssystemen
- Tablets, und
- Smartphones.

Darüber hinaus sind eingebettete GeoGebra Applets auf Websites verfügbar, speziell auf GeoGebra Materials (<https://www.geogebra.org/materials/>) mit Millionen kostenlos verfügbarer Unterrichtsmaterialien.

Die verfügbaren Werkzeuge können unter Umständen variieren. Auch die Benutzererfahrung kann auf den verschiedenen Plattformen unterschiedlich sein: Die symbolischen Berechnungen können ein hohes Ausmaß an Kalkulationen benötigen und die unterliegenden Hardware- und Softwarekomponenten unterstützen möglicherweise nicht alle Schritte vollständig.

Die vorgeschlagene Plattform für Klassen kann variieren. Die schnellsten Resultate können mit schnellen Computern erzielt werden, aber in diesem Fall muss die Software vom Benutzer heruntergeladen und installiert werden. Manche Beispiele in diesem Tutorial funktionieren unter Umständen nur in der „Desktop“ Version (welche für Microsoft Windows, Apple Macintosh und Linux Computer verfügbar ist). Auf der anderen Seite benötigt die „Web“ Version keine Installation durch den Benutzer: Diese funktioniert mit einem aktuellen Web Browser und die Lehrerin/der Lehrer kann zum Beispiel eine Liste von Beispielen als GeoGebra Applets für die Klasse unter Verwendung von GeoGebra Materials vorbereiten. Die „Web“ Version ist allerdings langsamer: Die symbolische Berechnung kann etwas langsamer sein.

GeoGebra funktioniert mittlerweile auch auf Tablets und Smartphones. In manchen Fällen können diese Plattformen schnellere Ergebnisse liefern als die Web Version, aber der kleinere Bildschirm könnte die Benutzer davon abhalten, geometrische Lehrsätze im Detail zu untersuchen. Lehrerinnen und Lehrer sollen bestärkt werden, mit dem Umgang dieser modernen Hilfsmittel zu experimentieren, aber der Einsatz für automatische Beweisführung ist noch nicht erforscht.

An den Werkzeugen zur automatischen Beweisführung wird kontinuierlich gearbeitet. Es ist zu bevorzugen, immer die aktuellste Version zu verwenden. Ein wöchentliches Update kann generell für alle Versionen empfohlen werden, abgesehen vom Mac App Store, welcher ungefähr monatlich aktualisiert wird. Die Liste der neuesten Änderungen kann auf <http://dev.geogebra.org/trac/timeline> eingesehen werden; das ist nur für fortgeschrittene Benutzer und Entwickler.

### 3 Automatische Beweisführung Werkzeuge

Automatische Beweisführung Werkzeuge sind eine Sammlung von GeoGebra Werkzeugen und Befehlen zur Mutmaßung, Entdeckung, Abstimmung und Beweis geometrischer Axiome in einer dynamischen geometrischen Konstruktion.

Am Anfang muss der Benutzer eine geometrische Figur mit bestimmten Werkzeugen auf dem Hauptfenster in GeoGebra zeichnen. Nach der Konstruktion der Figur kann GeoGebra auf verschiedene Arten die geometrischen Eigenschaften mit verschiedenen Werkzeugen und Einstellungen prüfen:

1. Mit *freie Objekte ziehen* ihre abhängigen Objekte können sichtbar erforscht werden.
2. Das *Beziehung* Werkzeug hilft beim Vergleichen von Objekten und erkennen von Beziehungen.
3. Mit dem Setzen der *Spur* eines konstruierten Objektes mit/ohne Bewegung eines Objektes wird sichtbar wenn seine vorhergehenden Objekte geändert werden.
4. Das *Ortslinie* Werkzeug zeigt die Spur eines Objektes für alle möglichen Positionen eines vorhergehenden Objektes (wenn es sich auf einem Pfad bewegt).
5. Mit dem Eintippen von *Beziehung* oder *Ortslinie* Befehl in der GeoGebra Befehlszeile können genauere Informationen ausgewählt werden.

Diese Methoden sind normalerweise bekannt in der GeoGebra community, deshalb sind sie gut dokumentiert und es können einige Beispiele auf GeoGebra Materials (<https://www.geogebra.org/materials/>) gefunden werden. Auf der anderen Seite bietet GeoGebra zurzeit auch symbolische automatische Beweisführung Werkzeuge für vermutete geometrische Eigenschaften:

6. *Beziehung* Werkzeug und Befehl können zum symbolischen Nachrechnen verwendet werden.
7. Der *Ortsliniengleichung* Befehl berechnet das Ergebnis der Ortsliniengleichung mit dem Zeigen der algebraischen Gleichung der graphischen Darstellung genauer.
8. Der *Ortsliniengleichung* Befehl kann implizite Ortslinien prüfen.
9. Der *Einhüllende* Befehl berechnet die Gleichung der Tangente mehrerer Objekte während ein bestimmter Vorgänger sich auf einer Bahn bewegt.

#### 3.1 Einfache und schwierige Werkzeuge

GeoGebra liefert die obenstehenden schwierigen Methoden um geometrische Theorien zu unterstützen. Diese Icons sind als schwierige Werkzeuge gedacht, weil sie aufgrund ihrer einfachen Benutzung direkt in der Klasse gezeigt werden können. Es gibt auch andere Wege, um mehr über den mathematischen Hintergrund zu lernen oder einfach zur Fehleranalyse. Diese einfachen Methoden sind im Zusatz aufgelistet und daher nicht empfohlen für den direkten Einsatz mit Schülerinnen und Schülern.

Offensichtlich sind einige der aufgelisteten Methoden einfach und andere schwierige. Bei der Befehlszeile in GeoGebra kann überlegt werden als ein schwierigerer Weg für die meisten Benutzer. Daher kann eine Lehrerin/ein Lehrer zuerst die einfacheren Methoden zeigen und nach einiger Erfahrung der Schülerinnen und Schüler die schwierigeren.

#### 3.2 Werkzeuge mit symbolischer Unterstützung

Wie oben erwähnt liefern einige automatische Beweisführung Werkzeuge symbolische Unterstützung. Dieses Feature erlaubt die Verifizierung einem mathematisch rigorosem Weg allgemeine Aussagen der Elementargeometrie, die vom Benutzer vermutet werden.

Ein allgemeiner Hinweis, dass der Benutzer sollte GeoGebra zur Inbetriebnahme mit "Grafikrechner" Modus starten. Diese Einstellung zeigt die Bezeichnung auf jedem neu hinzugefügten Objekt. Das kann für Beziehungswerkzeug und -befehl entscheidend sein, wenn die verschiedenen Beziehungen ausgewertet werden.

In den meisten Fällen sind die Achsen nicht wirklich notwendig: Diese können ausgeschaltet werden, wenn das *Bewegen* Werkzeug aktiv ist. (das ist das linke Icon, das einen Cursor zeigt) mit rechtsklick in der *Grafik Ansicht* und Auswahl der *Achsen* Einstellung.

In einigen Fällen ist es nicht notwendig, die *Algebra Ansicht* anzuzeigen. Wenn die Gleichungen der impliziten Kurven nicht im Detail betrachtet werden, kann dies auch durch die Änderung der Bezeichnung erreicht werden,

sodass auch der Wert angezeigt wird. (Um das durchzuführen, muss ein Rechtsklick und die Auswahl *Objekt Eigenschaften* durchgeführt werden, der Benutzer muss *Beschriftung anzeigen* zu *Wert* an dem *Basis* auswählen.)

GeoGebra Applets können komfortabel genutzt werden, wenn sie auf GeoGebra Materials hochgeladen sind. Wenn die Algebra Ansicht verwendet wird, kann es hilfreich sein, die Weite zu vergrößern, bevor das Applet auf GeoGebra Materials hochgeladen wird. Sonst könnte es schwierig für den Benutzer sein, die Befehle einzugeben. Nach dem hochladen in GeoGebra Materials ist es wichtig, die Funktionen des Applets *Erweiterte Einstellungen*, *Zeige Werkzeugleiste* und *Zeige Befehlszeile* zu überprüfen. Danach sollte eine adäquate Größe für das Applet gewählt werden.

### 3.2.1 Das Beziehungswerkzeug und Befehle

GeoGebras Beziehungswerkzeug und Befehle zeigt ein Fenster, das dem Benutzer Informationen über die Beziehung zwischen zwei oder mehr Objekten gibt. Dieser Befehl erlaubt dem Benutzer, numerisch herauszufinden, ob (für die gezeichnete Konstruktion mit zugehörigen Koordinaten)

- zwei Geraden normal aufeinander stehen,
- zwei Geraden parallel sind,
- zwei (oder mehr) Objekte (Punkte, Streckenlängen, Polygonflächen) gleich sind,
- ein Punkt auf einer Linie oder Kurve liegt,
- eine Gerade eine Tangent oder eine Passante einer Kurve ist,
- drei Punkte kollinear sind,
- drei Geraden übereinstimmend sind (oder parallel),
- vier Punkte auf einer Kreisbahn liegen (oder kollinear sind).

Manche dieser Überprüfungen können auch symbolisch ausgeführt werden, also der Satz kann generell verifiziert werden (mit beliebigen Koordinaten) und nicht nur für die gezeichnete konkrete Konstruktion.

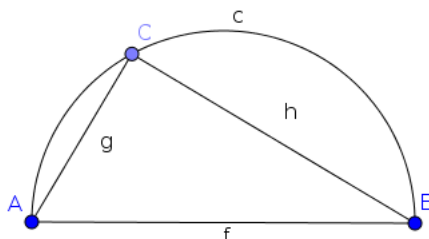
Mit der Verwendung des *Beziehungs Werkzeug* muss der Benutzer auf zwei Objekte klicken um die Mitteilungsbox zu sehen. Alternativ können zwei, drei oder vier Objekte mit dem Auswahlrechteck ausgewählt werden, um die Mitteilungsbox zu erhalten. Um nicht ungewollte Objekte im Applet auszuwählen ist es möglich, diese von der Auswahl auszunehmen, indem man mit einem Rechtsklick und der Auswahl *Objekt Eigenschaften Erweitert* auswählt und das Häkchen bei *Auswahl erlaubt* entfernt.

Bei der Verwendung des *Beziehungs Befehls* gibt der Benutzer eine der folgenden Formlen in die Befehlszeile ein:

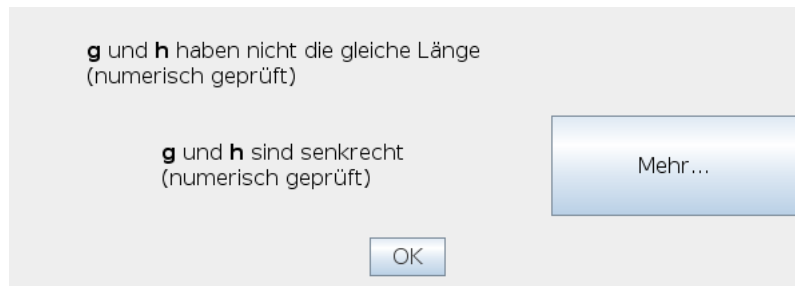
- `Beziehung[ <Objekt>, <Objekt> ]`
- `Beziehung[ { <Objekt>, <Objekt> } ]`
- `Beziehung[ { <Objekt>, <Objekt>, <Objekt> } ]`
- `Beziehung[ { <Objekt>, <Objekt>, <Objekt>, <Objekt> } ]`

Wenn die Mitteilungsbox eines oder mehrere zutreffende numerische Aussagen zu dem Objekt zeigt, gibt es unter Umständen eine Schaltfläche “Mehr...”, welche die symbolische Unterstützung für die gegebene Aussage zeigt. Bei Klick auf “Mehr...”, wird in Kürze die numerische Aussage zu einer allgemeinen Aussage aktualisiert.

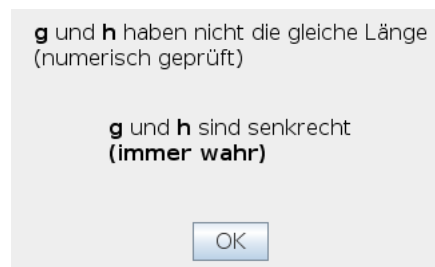
#### Beispiel (Thaleskreis)



1. Mit dem *Strecke* Werkzeug wird  $AB$  konstruiert.
2. Mit der Auswahl *Halbkreis durch zwei Punkte* Werkzeug wird  $c$  erstellt.
3. Der Punkt  $C$  wird auf  $c$  erstellt.
4. Konstruieren Sie die Strecken  $AC$  und  $BC$  und benennen Sie diese  $g$  und  $h$ .
5. Vergleichen Sie  $g$  und  $h$  unter Verwendung des Beziehungswerkzeugs durch Anklicken von  $g$  und  $h$ , oder durch Eintippen von `Beziehung[g,h]` in der Befehlszeile. Die folgende Mitteilung wird angezeigt:



6. Klicken Sie auf “Mehr...” – die Mitteilung wird sich wie folgt ändern:



Beachten Sie, dass Beziehung (Schritt 5) sucht nach Beziehungen zwischen  $g$  und  $h$  von den Koordinaten und Gleichungen abhängig von der gezeichneten Konstruktion. Durch Klicken auf “Mehr...” (Schritt 6) kann verifiziert werden, dass  $g$  und  $h$  normal aufeinander stehen und für alle Punkte kann  $A$  und  $B$  kann man Schritt 1 auswählen.

Beziehung zwischen bestimmten Objekten kann unter bestimmten Bedingungen wahr sein, aber nicht zwingend “immer”. In diesen Fällen werden genug Voraussetzungen angezeigt. Sonst zeigt GeoGebra nur die Aussage “unter bestimmten Voraussetzungen”. Das muss als “allgemein wahr” verstanden werden, aber in manchen Einzelfällen (welche eine ‘minimale Anzahl von Fällen’ ist, verglichen mit den allgemeinen Fällen) kann die Aussage falsch sein.

Das symbolische Ergebnis der Beziehung kann auch negativ sein, auch wenn die numerische Überprüfung positiv ist. Zum Beispiel, wenn man zwei Punkte  $P = (0,0)$  und  $Q = (0,0)$  definiert, kann die Beziehung diese numerisch vergleichen, aber das symbolische Ergebnis kann ergeben, dass “ $P$  und  $Q$  gleich sind (aber nicht allgemein wahr)”.

Eine komplette Übersicht der verschiedenen Ergebnisse der Beziehung können im Anhang gefunden werden. 6.1.3 auf der Seite 13.

### 3.2.2 Die Ortsliniengleichung Ortsliniengleichung Befehl

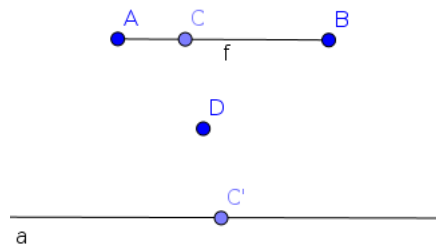
Dieser Befehl berechnet die Gleichung einer Ortslinie und zeichnet diese als implizite Kurve. Es gibt zwei Möglichkeiten der Nutzung:

- **Explizite Ortslinie.** Gegeben ist ein Anfangspunkt  $\mathcal{I}$  auf einer Strecke  $\mathcal{P}$ , einige Konstruktionsschritte und ein Endpunkt  $\mathcal{O}$ . Die Aufgabe ist, die Gleichung von  $\mathcal{E}$  of  $\mathcal{O}$  zu bestimmen, wenn sich  $\mathcal{I}$  auf  $\mathcal{P}$  bewegt, und dann  $\mathcal{E}$  zu bestimmen.  $\mathcal{I}$  ist für gewöhnlich *beweger* genannt,  $\mathcal{O}$  ist die *Spur*.  $\mathcal{E}$  heißt *Ortsliniengleichung* und ihre grafische Visualisierung ist die *Ortslinie*.

Die Schreibweise des Befehls ist:

`Ortsliniengleichung[ <Punkt Spur>, <Punkt Beweger> ].`

## Beispiel

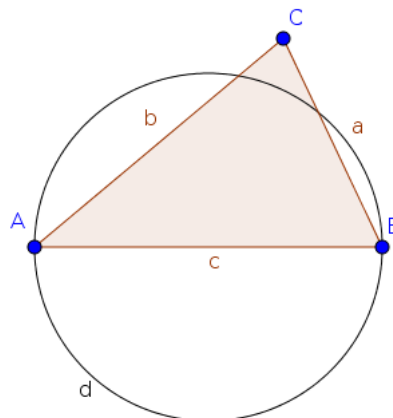


1. Unter Verwendung des *Strecke* Werkzeug wird  $AB$  konstruiert. Es wird eine automatische Strecke  $f$  konstruiert.
  2. Setzen Sie Punkt  $C$  auf  $f$ .
  3. Erstellen Sie Punkt  $D$  mit dem *Punkt* Werkzeug.
  4. Mit dem *über Punkt spiegeln* Werkzeug spiegeln Sie  $C$  über  $D$ . Damit erstellen Sie Punkt  $C'$ .
  5. Geben Sie Ortsliniengleichung  $[C', C]$  in die Befehlszeile ein. Nun wird die implizite Kurve  $a$  berechnet und gezeichnet. Das sollte eine Strecke (das Spiegelbild von  $f$  über  $D$  sein), aber GeoGebra zeichnete  $f$  als Gerade statt einer Strecke (aus algebraischen geometrischen Gründen), da das Spiegelbild ebenfalls eine Gerade ist.
  6. Versuchen Sie, die verschiebbaren Objekte zu ziehen. Es kann gezeigt werden, dass das Spiegelbild einer Strecke über einen Punkt immer parallel zum Urbild ist.
- **Implizite Ortslinie.** Gegeben ist ein Anfangspunkt  $\mathcal{J}$ , entweder als beliebiger Punkt oder auf einer Strecke  $\mathcal{P}$ . Es sind auch einige Konstruktionsschritte gegeben. Der Benutzer behauptet eine Boolesche Bedingung  $\mathcal{C}$  auf einige Objekte der Konstruktion. Die Aufgabe ist, eine Gleichung  $\mathcal{E}$  für alle Punkte  $\mathcal{J}'$  aufzustellen, sodass  $\mathcal{J} = \mathcal{J}'$ ,  $\mathcal{C}$  erfüllt. Wieder heißt  $\mathcal{E}$  Ortsliniengleichung und ihre grafische Interpretation ist die Ortslinie.

Die Schreibweise des Befehls ist:

Ortsliniengleichung[ <Boolescher Ausdruck>, <Punkt> ].

## Beispiel



1. Mit dem *Polygon* Werkzeug wird ein Dreieck  $ABC$  konstruiert. Die Strecken  $a$ ,  $b$  und  $c$  werden automatisch von GeoGebra erstellt.
2. Es wird Ortsliniengleichung  $[a^2 + b^2 == c^2, C]$  in der Werkzeugleiste eingegeben. Jetzt wird die implizite Kurve  $d$  berechnet und gezeichnet, welche wie ein Kreis aussieht. Bedenken Sie, dass *zwei* gleiche Zeichen eingegeben werden müssen; eine andere Möglichkeit ist,  $\hat{=}$  zu verwenden. (durch anklicken von  $\alpha$  Icon, oder durch Einfügen des Symbols von einer externen Applikation mit Copy and Paste).

3. Versuchen Sie nun, die beweglichen Objekte zu ziehen. Es kann gezeigt werden, dass wenn  $C$  auf einem Kreis liegt, dessen Durchmesser auf  $AB$  liegt, wegen der rechtwinkligen Eigenschaft des Dreiecks,  $a^2 + b^2 = c^2$  gilt.

Eine Boolesche Bedingung kann sein:

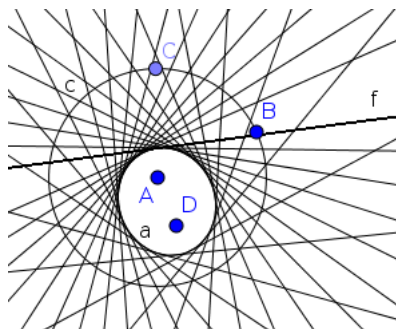
- Eine *algebraische* Gleichung mit der Bezeichnung der Strecken, e.g.  $a^2 + b^2 = c^2$ .
- Eine Gleichheit von zwei Geometrischen Objekten, e.g.  $A = B$ . Beachten Sie, dass *zwei* gleiche Zeichen eingegeben werden müssen; es gibt auch andere Möglichkeiten
  - $\stackrel{?}{=}$  (durch Anklicken des  $\alpha$  Icon, oder durch Einfügen des Symbols von einer externen Applikation mit Copy and Paste.
  - Alternativ, `SindGleich1[A,B]` für die gesamte Boolesche Bedingung.
- Eine Überprüfung, ob zwei geometrische Objekte kongruent sind, e.g. `SindKongruent[c,d]`.
- Eine Überprüfung, ob ein Punkt auf einer Strecke, einer Gerade oder einem Kreis liegt, e.g.  $A \in c$ .
- Eine Überprüfung, ob zwei Geraden oder Strecken parallel sind, e.g.  $p \parallel q$ . Hier kann auch `SindParallel[p,q]` verwendet werden.
- Eine Überprüfung, ob zwei Geraden oder Strecken normal aufeinander stehen, e.g.  $p \perp q$ . Hier kann auch `ArePerpendicular[p,q]` verwendet werden.
- `SindKollinear[A,B,C]` prüft, ob die Punkte  $A, B$  und  $C$  kollinear sind.
- `SindKongruent[d,e,f]` prüft, ob die Geraden  $d, e$  und  $f$  kongruent sind.
- `SindKonzentrisch[A,B,C,D]` prüft, ob die Punkte  $A, B, C$  und  $D$  konzentrisch sind.

### 3.2.3 Der Einhüllende Befehl

Dieser Befehl berechnet die Gleichung einer Tangente zu einer Gruppe von Objekten mit einem bestimmten Vorgänger auf einer Strecke.

Präziser gesagt, ist ein Anfangspunkt  $\mathcal{J}$  auf einer Strecke  $\mathcal{P}$ , einige Konstruktionsschritte und ein Endpunkt  $\mathcal{O}$  gegeben, entweder eine Gerade oder ein Kreis. Die Aufgabe besteht darin, die Gleichung  $\mathcal{E}$  einer Kurve  $\mathcal{C}$ , welche die Tangente zu  $\mathcal{O}$  ist, aufzustellen, wenn sich  $\mathcal{J}$  auf  $\mathcal{P}$ . Am Ende ergibt sich  $\mathcal{E}$ .  $\mathcal{J}$  wird der *Beweger* genannt.  $\mathcal{E}$  wird die *Einhüllende Gleichung* genannt, und ihre graphische Darstellung ist die *Einhüllende*.

#### Beispiel



1. Mit dem *Kreis mit Mittelpunkt durch Punkt* Werkzeug, wird ein Kreis  $c$  mit dem Mittelpunkt  $A$  und dem Umkreispunkt  $B$  konstruiert.
2. Setzen Sie Punkt  $C$  auf  $c$ .
3. Erstellen Sie einen beliebigen Punkt  $D$  in  $c$ .
4. Konstruieren Sie die *Mittelsenkrechte*  $f$  der Strecke  $CD$  unter Verwendung der Endpunkte.
5. Geben Sie `Einhüllende[f,C]` in der Befehlszeile ein. Die implizite Kurve  $a$  wird berechnet und gezeichnet. Diese sieht aus wie eine Ellipse.

### 3.3 Technische Notizen

Die folgenden Notizen sind wichtige Einschränkungen für jedes Automatische Begründungswerkzeug in GeoGebra, symbolische Berechnungen nutzt:

- Es werden nicht alle GeoGebra Werkzeuge und Konstruktionsschritte unterstützt.
- Die unterstützten Werkzeuge funktionieren nur für eine beschränkte Menge an geometrischen Objekten, zum Beispiel die Verwendung von Punkten, Geraden, Kreisen, Kegel.
- Abschnitte von Strahlen und Geraden werden als unendliche Geraden behandelt. Kreisbögen werden wie Kreise behandelt.
- Zu komplizierte Ortslinien- oder Einhüllende-Berechnungen können in der Algebra Ansicht als „undefiniert“ dargestellt werden.
- Beziehungs-Beweise, deren Inhalt zu kompliziert ist, werden die Mitteilung möglicherweise allgemein wahr anzeigen. Die Interpretation dafür ist, dass es GeoGebra nicht möglich ist, zu entscheiden, ob die Beziehung im allgemeinen wahr ist, aber die numerischen Ergebnisse lassen darauf schließen. Das heißt, die Beziehung kann auch im allgemeinen falsch sein.
- Wenn es keine Ortslinie oder Einhüllende gibt, ist die implizite Kurve die leere Menge  $0 = 1$ . Zum Beispiel: Für einen beliebigen Punkt  $P$

`Ortsliniengleichung[false,P]`

zeigt die leere Menge.

- Wenn die Ortslinie oder die Einhüllende die ganze Ebene ist, hat die implizite Kurve die Gleichung  $0 = 0$ . Zum Beispiel: Für einen beliebigen Punkt  $P$

`Ortsliniengleichung[true,P]`

zeigt die ganze Ebene.

- Manchmal werden zusätzliche Verzweigungen erscheinen, die nicht in der ursprünglichen Ortslinie oder Einhüllenden waren.
- Der Graph der impliziten Kurve kann in manchen Fällen ungenau sein.

## 4 Verwendung in der Klasse: Vermutung, Beweis und Verallgemeinerung

Technisch gesehen ist das Beziehungswerkzeug das einfachsten symbolische Werkzeug aus der obigen Liste. Auf der anderen Seite können manche Unterrichtssituationen den Einsatz andere Werkzeuge, oder den Einsatz mehr als eines Werkzeuges erfordern, möglicherweise auch in einer anderen Reihung als in der obigen Liste.

### 4.1 Satz des Thales

In vielen traditionellen Mathematikstunden wird der Satz des Thales in einer expliziten Form unterrichtet: Wenn  $C$  auf einem Halbkreis liegt, sind die Strecken  $g$  und  $h$  normal aufeinander. Tatsächlich kann dieser Lehrsatz mit einer offenen Frage formuliert werden: *Sei  $ABC$  ein beliebiges Dreieck. Was ist die geometrische Ortslinie von  $C$ , wenn der Winkel bei  $C$  ein rechter Winkel sein soll?*

Bei diesem Zugang kann es sinnvoller sein, den technisch anspruchsvolleren `Ortsliniengleichung[g⊥h,C]` Befehl zuerst zu verwenden und dann die Konstruktion mit dem Beziehungswerkzeug oder -befehl zu vollenden. Der Output des `Ortsliniengleichung` Befehl kann den Schülerinnen und Schülern zeigen, dass die Kurve ein Kreis ist. Die Algebra Ansicht zeigt die Gleichung der Kurve, das könnte unter Umständen für jüngere Schülerinnen und Schüler verwirrend sein.

Schließlich kann der Satz des Thales durch das eingeschriebene Dreieck in einen Halbkreis verallgemeinert werden. In diesem Fall ist die Bedingung nicht mehr  $g \perp h$ , sondern der Winkel der beiden veränderlichen Strecken zu einer fixen. GeoGebra unterstützt diese Art der Erhebung mit der Schreibweise.

`Ortsliniengleichung[SindKongruent[α,β],C]`

wenn  $\alpha$  fix ist und  $\beta = \angle ACB$ .

Zusammenfassend, bei diesem Zugang

1. wird eine implizite Ortslinie mit GeoGebra berechnet,
2. wird von den Schülerinnen und Schülern eine Vermutung aufgestellt, wie die Kurve aussieht,
3. wird die Vermutung mit dem Beziehungswerkzeug oder -befehl in GeoGebra überprüft,
4. kann der Beweis von den Schülerinnen und Schülern optional handschriftlich durchgeführt werden,
5. wird der Satz des Thales mit den impliziten Ortslinien in GeoGebra verallgemeinert – als zukünftiges Experiment für die Schülerinnen und Schüler.

## 4.2 Weitere Beispiele

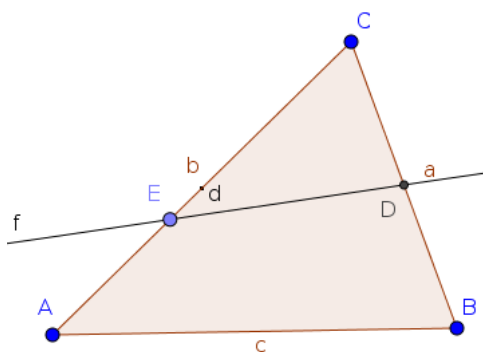
Die Dreiecksungleichung kann in eine Gleichung übersetzt werden, die in eine Erforschung der degenerierten Dreiecke geändert werden kann. Als Verallgemeinerung kann die künstliche Definition von konischen Segmenten erwähnt werden.

Eine andere Anwendung ist eine Verzweigung der Ortsliniengleichung im Dreieck  $ABC$  mit der Bedingung  $a \stackrel{?}{=} b$ , dazu muss  $C$  gefunden werden (Schritt 1). Natürlich muss  $C$  auf der Streckensymmetrale  $AB$  liegen (Schritt 2). Dass  $C$  explizit auf der Streckensymmetrale liegt, bestätigt GeoGebra, als  $AC = BC$ , wenn man die Berechnung mit dem Beziehungswerkzeug startet (Schritt 3). Nach dem Beweis des Satzes mit traditionellen Methoden (Schritt 4) kann eine Verallgemeinerung durch das Eintippen des Befehls `Ortsliniengleichung[a==2b,C]` erhalten werden. Dies kann auch ein interessantes Experiment für fortgeschrittene Schülerinnen und Schüler sein (Schritt 5).

## 4.3 Ein ausgearbeitetes Beispiel: Der Satz von der Mittelparallelen im Dreieck

Diese Schritt für Schritt Anleitungen sind ein möglicher Weg, den Satz von der Mittelparallelen im Dreieck mit den automatischen Beweiswerkzeugen in GeoGebra durchzuführen.

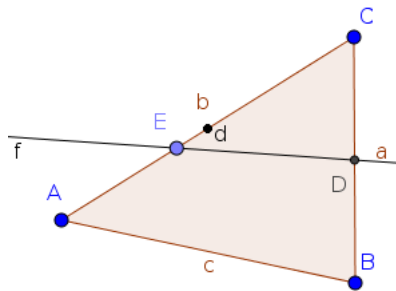
### Schritt 1



1. Mit Verwendung des *Polygon* Werkzeuges wird ein Dreieck  $ABC$  konstruiert. Dabei werden automatisch die Strecken  $a$ ,  $b$  und  $c$  konstruiert.
2. Mit Verwendung des *Mittelpunkt* Werkzeuges, wird der Mittelpunkt  $D$  von  $a$  konstruiert..
3. Erstellen Sie Punkt  $E$  auf  $b$ .
4. Erstellen Sie die Gerade  $f$ , die  $D$  und  $E$  verbindet.
5. Fragen Sie GeoGebra, wie  $E$  sein muss, damit  $f$  parallel zu  $c$  ist: Tippen Sie `Ortsliniengleichung[c||f,E]` in die Eingabezeile. Dadurch wird die implizite Kurve  $d$  berechnet und konstruiert. Diese scheint ein einziger Punkt zu sein. Achtung: Es könnte hilfreich sein, die Linienstärke der impliziten Kurve  $d$  zu ändern und auch die Schriftgröße zu ändern, damit diese nicht von anderen Objekten versteckt wird. Beide Einstellungen können im Fenster Grundeinstellungen geändert werden.

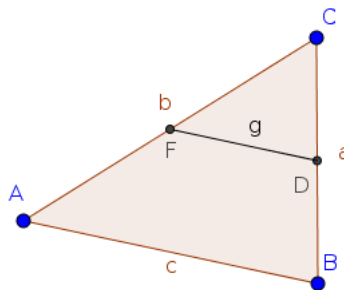


## Schritt 2

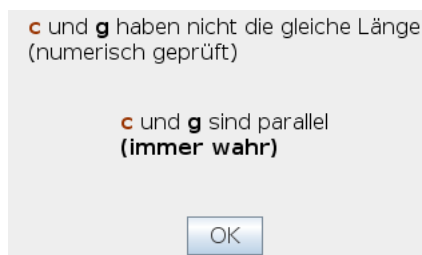


6. Ziehen Sie an den beweglichen Objekten und zeigen Sie, dass  $E$  der Mittelpunkt von  $b$  sein muss.
7. Um diese Vermutung zu bestätigen, erstellen Sie den Mittelpunkt  $F$  der Strecke  $b$  (und passen Sie die Beschriftung von  $d$  und  $F$ , um Überschneidungen zu vermeiden.). Ziehen Sie erneut an den beweglichen Objekten.

## Schritt 3



8. Machen Sie die Objekte  $E$ ,  $f$  und  $d$  unsichtbar, indem Sie diese verstecken.
9. Verbinden Sie  $D$  und  $F$  mit einer Strecke  $g$ .
10. Verwenden Sie das *Beziehung* Werkzeug, um  $c$  und  $g$  zu vergleichen. Sie sind parallel.
11. Klicken Sie auf "Mehr ..." im Pop Up Fenster und überprüfen Sie symbolisch, dass sie parallel sind.



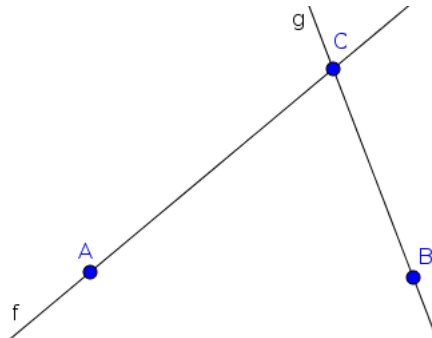
Die Schülerinnen und Schüler machen mit Schritt 4 weiter, wenn sie einen eleganten Weg benötigen, diese Behauptung zu beweisen, oder hören an dieser Stelle auf, wenn keine weitere Zeit in der Klasse für dieses Problem aufgewendet werden soll.

Auch in Schritt 5 können weitere Fragen auftreten.  $c$  und  $g$  haben nicht die selbe Länge, können aber unter Verwendung der Länge von  $c$  berechnet werden? Vielleicht  $c = 1.5 \cdot g$ , oder vielleicht mehr? Der GeoGebra Befehl `Relation[c, 1.5g]` ergibt, dass  $c$  und  $1.5g$  nicht gleich sind, aber es vielleicht eine andere Konstante als 1.5 gibt, die eine positive Antwort erzeugt. ... Auch wenn keine Zeit für weitere Arbeiten in der Klasse bleibt, werden manche Schülerinnen und Schüler diese Fragen interessant finden und alleine oder in Gruppen darüber nachdenken, sinnvollerweise *independently*, unter Verwendung des Computers als Experten.

## 5 Beschränkung: Eine Fallstudie des Satzes von Thales

Intuitive Benutzung des GeoGebra automatischen Beweiswerkzeuges kann unter Umständen unerwartete Ergebnisse hervorbringen. Diese Unterteilung erklärt einige häufige Fehler bei der Benutzung.

Der Satz von Thales wird diskutiert.

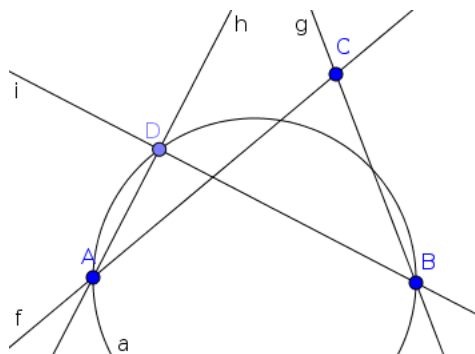


1. Erstellen Sie die Punkte  $A$ ,  $B$  und  $C$ .
2. Erstellen Sie die Geraden  $f = AC$ ,  $g = BC$ .
3. Überprüfen Sie das Ergebnis des Befehls `Beziehung[f,g]`: “ $f$  schneidet  $g$ ”.
4. Fragen Sie GeoGebra über geometrische Voraussetzungen von  $f \perp g$ :

Ortsliniengleichung [ $f \perp g, C$ ].

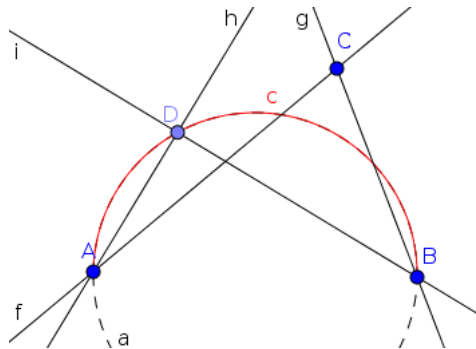
Eine implizite Kurve  $a$ , die kreisförmig aussieht, wird gezeigt.

5. Bewegen Sie  $C$  so nahe wie möglich an die implizite Kurve. Nun wird `Relation[f,g]` noch immer aussagen, dass “ $f$  schneidet  $g$ ”. **Warum? Weil der Punkt  $C$  nicht direkt auf der Kreisbahn liegt. Wir müssen den Punkt exakt auf die Kreisbahn legen.**
  - (a) Versuchen Sie, den Punkt  $C$  direkt auf die implizite Kurve  $a$  zu setzen. Verwenden Sie dazu das *Attach / Detach Point* Werkzeug. **GeoGebra erlaubt das nicht, weil per Definition  $a$  auf  $C$  liegt, und das würde keinen Sinn ergeben.**
  - (b) Erstellen Sie stattdessen einen neuen Punkt  $D$ , den Sie direkt auf die implizite Kurve setzen. Das erlaubt GeoGebra. Erstellen Sie auch die Geraden  $h = AD$ ,  $i = BD$ .



- (c) Überprüfen Sie das Ergebnis des Befehls `Beziehung[h,i]`: “ $h$  und  $i$  sind normal aufeinander” mit einer numerischen Überprüfung. Durch Klicken auf “Mehr...” ist das Ergebnis “vielleicht immer richtig”. **Warum? Weil GeoGebra die darunterliegenden implizite Kurve als das Ergebnis einer teilweisen Konfiguration der Konstruktion interpretiert. Mit anderen Worten: eine implizite Kurve ist ein numerisches Objekt, sie hat keine symbolische Bedeutung. Daher sind symbolische Überprüfungen basierend auf einer impliziten Kurve nicht möglich. Hier ist GeoGebra nur optimistisch, was die Wahrheit der Vermutung betrifft, aber die Software ist nicht fähig, dies zu prüfen.**

- (d) Der eigentliche Weg, die Schritte in dieser Vermutung zu einem Ende zu führen ist, einen Kreis mit dem Durchmesser  $AB$  mit dem Kreiswerkzeug zu konstruieren, zum Beispiel mit dem *Halbkreis durch 2 Punkte* Werkzeug, nachdem  $D$  von  $a$  abgetrennt wird und dann  $a$  unsichtbar zu machen. Dann kann  $D$  zu dem Halbkreis hinzugefügt werden.



(Optional kann die implizite Kurve auch durch eine andere Formatierung sichtbar gemacht werden. In diesem Beispiel wurde eine andere Formatierung auch für den Halbkreis verwendet. Zum Schluss zeigt Beziehung  $[h, i]$  positive Ergebnisse, sowohl numerisch, als auch symbolisch.

## 6 Anhang

### 6.1 Einfache GeoGebra Werkzeuge

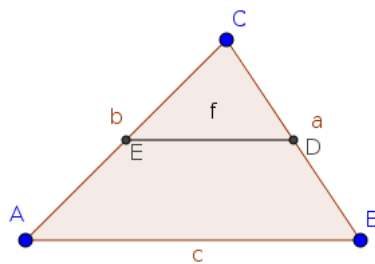
Automatische Beweisführungswerkzeuge in GeoGebra werden abgerundet durch einige einfache Werkzeuge, die zum zusätzlichen Lernen über exaktere geometrische Eigenschaften.

#### 6.1.1 Der Prüfe Befehl

Der Prüfe Befehl entscheidet, ob eine geometrische Aussage im allgemeinen wahr ist. Es hat drei mögliche Ergebnisse:

- *wahr* bedeutet, dass die Aussage immer wahr ist, oder wahr unter bestimmten, nicht-degenerierten [6] oder essentiellen [14] Bedingungen oder auf einigen Komponenten [12].
- *falsch* bedeutet, dass die Aussage im allgemeinen falsch ist.
- *nicht definiert* bedeutet, dass GeoGebra aufgrund verschiedener Ursachen keine Entscheidung treffen kann:
  - Die Aussage kann nicht in ein Modell übersetzt werden, das erforscht werden kann. Das bedeutet normalerweise, dass die Algebraisierung der Aussage nicht durchgeführt werden kann, aufgrund
    - \* theoretischer Unmöglichkeit (e.g. Verwendung einer transzendenten Funktion als Konstruktions-schritt, z.B Sinus von  $x$ ),
    - \* fehlende Implementation in GeoGebra.
  - Die übersetzte Aussage in algebraischer Geometrie ist zu schwierig zu lösen. Das kann einerseits bedeuten, dass es zu viele Variablen gibt, oder die Gleichungen sind mit einem Algorithmus schwierig zu bearbeiten. Das führt entweder zu einem Timeout oder der Platz im Hauptspeicher reicht nicht aus.
  - Der Lösungsalgorithmus konnte die Situation prüfen, aber das Ergebnis ist nicht eindeutig: entweder ist die Behauptung falsch, oder sie ist wahr unter bestimmten Bedingungen. Aber der Algorithmus konnte nicht feststellen, welcher Fall vorliegt.
  - Es gab eine interne Störung in GeoGebra während der Berechnungen.

## Beispiel



1. Konstruieren Sie das Dreieck  $ABC$  mit dem *Vieleck* Werkzeug.
2. Konstruieren Sie die Mittelpunkte  $D$  und  $E$  der Seiten  $a$  und  $b$ , mit dem *Mittelpunkt* Werkzeug.
3. Unter Verwendung des *Strecke* Werkzeug, konstruieren Sie  $f$  durch Verbinden mit  $D$  und  $E$ .
4. Geben Sie `Prüfe[f||c]` ein, um zu zeigen, dass *wahr* in der Algebra Ansicht als Boolescher Wert *d* angezeigt wird. Achten Sie darauf, dass das Parallel Zeichen eingegeben werden muss, entweder durch:
  - die Liste der mathematischen Symbole durch Klicken auf das  $\alpha$  Icon in der Eingabezeile, oder
  - externe Eingabe des Symbols mit Copy and Paste.
  - Alternativ kann `f||c` auch durch `AreParallel[f, c]` ersetzt werden.
5. Geben Sie `Prüfe[c==3f]` ein. Jetzt ist die Antwort *undefiniert*, weil GeoGebra nicht entscheiden kann, ob die Aussage falsch oder wahr ist unter bestimmten Bedingungen. In diesen Fällen kann der Befehl `PrüfeDetails` helfen. (siehe unten). Beachten Sie, dass *zwei* gleiche Zeichen eingegeben werden müssen; andere Möglichkeiten der Verwendung sind:
  - `Prüfe[c?=3f]`, oder
  - `Prüfe[AreEqual[c, 3f]]`.

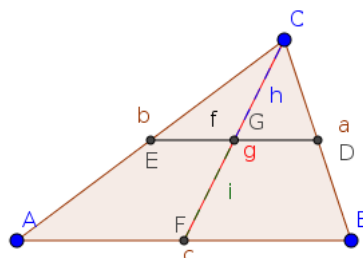
### 6.1.2 The PrüfeDetails Befehl

Der `PrüfeDetails` Befehl hat ähnliches Verhalten wie der `Prüfe` Befehl, aber es kann verschiedene Algorithmen im Entscheidungsprozess verwenden und könnte mehr Informationen des Ergebnisses zeigen. Es gibt drei mögliche Ergebnisse:

- `{true}` bedeutet, dass die Aussage immer wahr ist.
- `{true, {...}}` wenn die Aussage unter bestimmten, nicht-degenerierten [6] oder essentiellen [14] Bedingungen oder auf einigen Komponenten [12] wahr ist: Diese Bedingungen sind in Klammern geschrieben. (Wenn die Liste bleibt, bedeutet das, dass keine künstliche Übersetzung gefunden werden konnte "..."). Wenn die Verbindung der Verneinten Bedingungen wahr ist, dann ist die Aussage wahr.
- `{false}` bedeutet, dass die Aussage im Allgemeinen falsch ist.

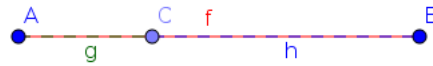
### Beispiel (fortgesetzt)

6. Geben Sie `PrüfeDetails[c==3f]` ein. Jetzt ist die Antwort `{false}`.
7. Geben Sie `PrüfeDetails[c==2f]` ein. Jetzt ist die Antwort `{true}`.



8. Sei  $F$  der Mittelpunkt von  $c$ , und die Strecke  $CF$  wird in  $g$  umbenannt. Sei  $G$  der Schnittpunkt von  $f$  und  $g$ . Zum Schluss werden noch die Strecken  $CG$  und  $FG$  in  $h$  und  $i$  umbenannt. In diesem Falle wird  $\text{PrüfeDetails}[h==i]$  wieder  $\{true, \{ "AreCollinear[A,B,C]" \}\}$ , was bedeutet, dass, wenn  $A$ ,  $B$  und  $C$  nicht kollinear sind, ist  $h = i$ .

### Ein weiteres Beispiel



1. Sei  $AB$  eine Strecke, umbenannt in  $f$ .
2. Sei  $C$  ein Punkt auf  $f$ .
3. Die Strecken  $AC$  und  $BC$  werden in  $g$  und  $h$  umbenannt.
4. Geben Sie  $\text{PrüfeDetails}[f==g+h]$  ein. Jetzt ist die Antwort

$$\{true, \{ "g+f=h", "h+f=g" \}\}$$

was bedeutet, dass, wenn  $g + f \neq h$  und  $h + f \neq g$ , ist  $f = g + h$ .

### 6.1.3 Ein Vergleich von Prüfe, PrüfeDetails und Beziehung

Die folgende Tabelle erklärt die Bedeutung der Ergebnisse der Befehle Prüfe, PrüfeDetails und Beziehung.

GeoGebra Outputs			Konklusion
Prüfe	PrüfeDetails	Symbolisches Fenster von <b>Beziehung</b>	
true	{true}	immer wahr	Die Aussage ist wahr.
	{true,{Bedingungen}}	wahr unter <i>Vereinigung von speziellen Bedingungen</i>	Die Aussage ist wahr, wenn die speziellen <i>Bedingungen</i> stimmen. Diese Bedingungen sind ausreichend, aber möglicherweise nicht notwendig. Es könnte andere ausreichende Bedingungen geben, die die Aussage wahr machen.
	{true,{...}}	wahr unter bestimmten Bedingungen	Die Aussage ist wahr, wenn gewisse Gleichungen stimmen. Diese Gleichungen haben keine klar ersichtliche Bedeutung für GeoGebra.
	{}	im Allgemeinen wahr	Die Aussage ist wahr, wenn gewisse Gleichungen stimmen. GeoGebra konnte diese Bedingungen nicht ermitteln, weil die Berechnungen zu kompliziert waren.
false	{false}	nicht im Allgemeinen wahr	Die Aussage ist falsch.
	{}	möglicherweise im Allgemeinen wahr	GeoGebra konnte nicht entscheiden, ob die Aussage wahr oder falsch ist. Die numerische Überprüfung zeigt, dass die Aussage wahr ist, aber die symbolische war konnte nicht durchgeführt werden, weil die Berechnungen zu kompliziert waren.
nicht definiert	{true}	immer wahr	Die Aussage ist wahr.
	{true,{Bedingungen}}	wahr unter <i>Vereinigung von bestimmten Bedingungen</i>	Die Aussage ist wahr, wenn die bestimmten <i>Bedingungen</i> stimmen. Diese Bedingungen sind ausreichend, aber möglicherweise nicht notwendig. Es könnte andere ausreichende Bedingungen geben, die die Aussage wahr machen.
	{true,{...}}	wahr unter bestimmten Bedingungen	Die Aussage ist wahr, wenn bestimmte Bedingungen stimmen. Diese Gleichungen haben keine klar ersichtliche Bedeutung für GeoGebra.
	{false}	im Allgemeinen falsch	Die Aussage ist falsch.
	{}	möglicherweise im Allgemeinen wahr	GeoGebra konnte nicht entscheiden, ob die Aussage wahr oder falsch ist. Die numerische Überprüfung vermutet, dass die Aussage wahr ist, aber die symbolische Überprüfung konnte nicht durchgeführt werden, weil die Berechnungen zu kompliziert waren, oder die gegebene Aussage ist noch nicht in GeoGebra eingebaut.

## 6.2 Fehlerbeseitigung

Wenn Sie GeoGebra mit einem Befehl in der Befehlszeile starten, gibt es mehr Möglichkeiten, die Ergebnisse zu überprüfen. Hier ist eine Methode in einer typischen Linux-Installation gezeigt.

Der Benutzer muss GeoGebra mit dem folgenden Befehl starten:

```
geogebra --logfile=/dev/stdout --logshowcaller=false \
--logshowtime=false --logshowlevel=false
```

Eine typische Ausgabe sieht wie folgt aus:

```

Using AUTO
Using BOTANAS_PROVER
A = (3.42, 1.86) /* free point */
// Free point A(v1,v2)
B = (10.48, 3.1) /* free point */
// Free point B(v3,v4)
f = Segment[A, B] /* Segment [A, B] */
C = Point[f] /* Point on f */
// Constrained point C(v5,v6)
Hypotheses:
1. -v5*v4+v6*v3+v5*v2-v3*v2-v6*v1+v4*v1
g = Segment[A, C] /* Segment [A, C] */
h = Segment[C, B] /* Segment [C, B] */
Processing numerical object
Hypotheses have been processed.
giac evalRaw input: evalfa(expand(ggbtmpvarf))
giac evalRaw output: ggbtmpvarf
input = expand(ggbtmpvarf)
result = ggbtmpvarf
eliminate([ggbtmpvarf-((ggbtmpvarg)+(ggbtmpvarh))=0,ggbtmpvarh^2=v11^2,ggbtmpvarg^2=v12^2,ggbtmpvarf^2=v13^2],[ggbtmpvarh,
ggbtmpvarg,ggbtmpvarf])
giac evalRaw input: evalfa(eliminate([ggbtmpvarf-((ggbtmpvarg)+(ggbtmpvarh))=0,ggbtmpvarh^2=v11^2,ggbtmpvarg^2=v12^2,
ggbtmpvarf^2=v13^2],[ggbtmpvarh,ggbtmpvarg,ggbtmpvarf]))
Running a probabilistic check for the reconstructed Groebner basis. If successfull, error probability is less than 1e-07
and is estimated to be less than 10^-18. Use proba_epsilon:=0 to certify (this takes more time).
// Groebner basis computation time 0.000448 Memory -1e-06M
giac evalRaw output: {v11^4-2*v11^2*v12^2+v12^4-2*v11^2*v13^2-2*v12^2*v13^2+v13^4},[
input = eliminate([ggbtmpvarf-((ggbtmpvarg)+(ggbtmpvarh))=0,ggbtmpvarh^2=v11^2,ggbtmpvarg^2=v12^2,ggbtmpvarf^2=v13^2],[
ggbtmpvarh,ggbtmpvarg,ggbtmpvarf])
result = {v11^4-2*v11^2*v12^2+v12^4-2*v11^2*v13^2-2*v12^2*v13^2+v13^4}
giac evalRaw input: evalfa(eliminate([ggbtmpvarf-((ggbtmpvarg)+(ggbtmpvarh))=0,ggbtmpvarh=v11,ggbtmpvarg=v12,ggbtmpvarf=
v13],[ggbtmpvarh,ggbtmpvarg,ggbtmpvarf]))
Running a probabilistic check for the reconstructed Groebner basis. If successfull, error probability is less than 1e-07
and is estimated to be less than 10^-18. Use proba_epsilon:=0 to certify (this takes more time).
// Groebner basis computation time 0.000592 Memory -1e-06M
giac evalRaw output: {v11+v12-v13}
input = eliminate([ggbtmpvarf-((ggbtmpvarg)+(ggbtmpvarh))=0,ggbtmpvarh=v11,ggbtmpvarg=v12,ggbtmpvarf=v13],[ggbtmpvarh,
ggbtmpvarg,ggbtmpvarf])
result = {v11+v12-v13}
giac evalRaw input: evalfa(simplify({v11^4-2*v11^2*v12^2+v12^4-2*v11^2*v13^2-2*v12^2*v13^2+v13^4}/{v11+v12-v13}))
giac evalRaw output: {v11^3-v11^2*v12+v11^2*v13-v11*v12^2-2*v11*v12*v13-v11*v13^2+v12^3+v12^2*v13-v12*v13^2-v13^3}
input = simplify({v11^4-2*v11^2*v12^2+v12^4-2*v11^2*v13^2-2*v12^2*v13^2+v13^4}/{v11+v12-v13})
result = {v11^3-v11^2*v12+v11^2*v13-v11*v12^2-2*v11*v12*v13-v11*v13^2+v12^3+v12^2*v13-v12*v13^2-v13^3}
giac evalRaw input: evalfa(factor(v11^3-v11^2*v12+v11^2*v13-v11*v12^2-2*v11*v12*v13-v11*v13^2+v12^3+v12^2*v13-v12*v13^2-
v13^3))
giac evalRaw output: (v11-v12-v13)*(v11-v12+v13)*(v11+v12+v13)
input = factor(v11^3-v11^2*v12+v11^2*v13-v11*v12^2-2*v11*v12*v13-v11*v13^2+v12^3+v12^2*v13-v12*v13^2-v13^3)
result = (v11-v12-v13)*(v11-v12+v13)*(v11+v12+v13)
Trying to detect polynomial -v13-v12+v11
-v13-v12+v11 means h = f + g
Trying to detect polynomial v13-v12+v11
v13-v12+v11 means f + h = g
Trying to detect polynomial v13+v12+v11
v13+v12+v11 means f + g + h = 0, uninteresting
Thesis equations (non-denied ones):
2. v11^2-v6^2-v5^2+2*v6*v4-v4^2+2*v5*v3-v3^2
3. v12^2-v6^2-v5^2+2*v6*v2-v2^2+2*v5*v1-v1^2
4. v13^2-v4^2-v3^2+2*v4*v2-v2^2+2*v3*v1-v1^2
Thesis reductio ad absurdum (denied statement), product of factors:
(v13^4-2*v13^2*v12^2+v12^4-2*v13^2*v11^2-2*v12^2*v11^4)*v14-1
that is,
5. -1+v14*v13^4-2*v14*v13^2*v12^2+v14*v12^4-2*v14*v13^2*v11^2-2*v14*v12^2*v11^4+v14*v11^4
substitutions: {v1=0, v2=0}
Eliminating system in 8 variables (5 dependent)
giac evalRaw input: evalfa([[[ff:="\\"],[aa:=eliminate2([v12^2-v6^2-v5^2,v11^2-v6^2-v5^2+2*v6*v4-v4^2+2*v5*v3-v3^2,-1+v14*
v13^4-2*v14*v13^2*v12^2+v14*v12^4-2*v14*v13^2*v11^2-2*v14*v12^2*v11^4,v13^2-v4^2-v3^2,-v5*v4+v6*v3],
revlist([v6,v11,v12,v13,v14]))],[bb:=size(aa)],[for ii from 0 to bb-1 do ff+=\"\\\"+(ii+1)+\\\": [1]:
unicode95uunicode91u1]=1\\\";cc:=factors(aa[ii]);dd:=size(cc);for jj from 0 to dd-1 by 2 do ff+=\"\\
unicode95uunicode91u\\\"+(jj/2+2)+\\\"=\\\"+cc[jj]]; od; ff+=\"\\\" [2]: \\\"+cc[1]);for kk from 1 to dd-1 by 2 do ff+=\"\\\",\\\"+
cc[kk]];od;od],[if(ff=\\\"\\\" begin ff:=[0] end],[ff][5])
Running a probabilistic check for the reconstructed Groebner basis. If successfull, error probability is less than 1e-07
and is estimated to be less than 10^-7. Use proba_epsilon:=0 to certify (this takes more time).
// Groebner basis computation time 0.000249 Memory -1e-06M
giac evalRaw output: "[1]: [1]: unicode95uunicode91u1]=1 unicode95uunicode91u2]=1 [2]: 1,1"
Considering NDG 1...
Found a better NDG score (0.0) than Infinity
Statement is GENERALLY TRUE
Benchmarking: 38 ms
STATEMENT IS TRUE (yes/no: TRUE)
OUTPUT for ProveDetails: null = {true, {"f + h = g", "h = f + g"}}

```

Es wird absichtlich kein einfacherer Weg gezeigt, die Ausgabe zu erstellen. Die letzten paar Zeilen der Information zur Fehlerbeseitigung ist in GeoGebra verfügbar im *Hilfe* Menü, durch Auswählen von *Über/Lizenz*, und durch Anklicken von *Systeminformation* Dies kopiert die aktuellsten Fehlerbeseitigungs-Nachrichten in die Zwischenablage.

### 6.3 Übersetzung von GeoGebra Befehlen

Die Bezeichnungen von GeoGebra automatischen Beweisführungs-Werkzeugen könnten auch als Übersetzungen in andere Sprachen benötigt werden. Zum Beispiel ist die deutsche Übersetzung von Prüfe Prüfe. Um die übersetzten Befehlsbezeichnungen zu lernen, sind folgende Schritte empfohlen:

1. Erstellen Sie eine GeoGebra Datei, die die benötigten Befehle in der Algebra Ansicht beinhaltet.
2. Ändern Sie die Sprache in GeoGebra im *Optionen* Menü durch Auswählen von *Sprache*.
3. Die Befehlsbezeichnungen werden in der Algebra Ansicht automatisch geändert.
4. Bewegen Sie die Maus über einen Befehl in der Algebra Ansicht und lesen Sie die übersetzte Bezeichnung ab.

### Literatur

- [1] Artigue, M.: ¿Qué la educación matemático basada en la en la indagación? La Gaceta de la Real Sociedad Matemática Española 20(3). 2017.
- [2] Abánades, M.A., Botana, F., Kovács, Z., Recio, T. and Sólyom-Gecse, C.: Development of automatic reasoning tools in GeoGebra. ACM Communications in Computer Algebra 50(3), p. 85–88. 2016.
- [3] Botana, F., Hohenwarter, M., Janičić, Kovács, Z., Petrović, I., Recio, T. and Weitzhofer, S.: Automated Theorem Proving in GeoGebra: Current Achievements. Journal of Automated Reasoning 55.1, p. 39–59. 2015.
- [4] Botana, F. and Recio, T.: On the Unavoidable Uncertainty of Truth in Dynamic Geometry Proving. Mathematics in Computer Science 10(1), p. 5–25. 2016.
- [5] Botana, F. and Recio, T.: Computing envelopes in dynamic geometry environments. Annals of Mathematics and Artificial Intelligence 80(1), p. 3–20. 2017.
- [6] Chou, S.-C.: *Mechanical geometry theorem proving*. Kluwer Academic Publishers, Norwell, MA, USA. 1987.
- [7] Cox, D.A., Little, J., O’Shea, D.: *Ideals, varieties, and algorithms. An introduction computational algebraic geometry and commutative algebra*. 4th revised ed. Undergraduate Texts in Mathematics, Springer, Cham. 2015.
- [8] Davis, P.: The rise, fall, and possible transfiguration of triangle geometry: a mini-history. The American Mathematical Monthly 102(3), p. 204–214. 1995.
- [9] Howson, G. and Wilson, B.: *ICMI Study series: School mathematics in the 1990’s*. Kuwait: Cambridge University Press. 1986.
- [10] Kovács, Z.: *Computer Based Conjectures and Proofs in Teaching Euclidean Geometry*. PhD thesis. Johannes Kepler University, Linz. 2015.
- [11] Kovács, Z.: Automated reasoning Tools in GeoGebra: A new approach for experiments in planar geometry. South Bohemia Mathematical Letters 25(1), p. 48–65. <http://home.pf.jcu.cz/~sbml/wp-content/uploads/Kovacs.pdf>. 2017.
- [12] Kovács, Z., Recio, T. and Vélez, M.P.: Detecting truth on components. CoRR abs/XXXX.XXXXX. 2018.
- [13] Kovács, Z., Recio, T., Richard, P.R. and Vélez, M.P.: GeoGebra Automated Reasoning Tools: A Tutorial with Examples. In: Aldon, G. and Trgalova, J. (Eds.): *Proceedings of the 13th International Conference on Technology in Mathematics Teaching*. <https://hal.archives-ouvertes.fr/hal-01632970>. 2017.
- [14] Kovács, Z., Recio, T. and Sólyom-Gecse, C.: Automatic Rewrites of Input Expressions in Complex Algebraic Geometry Provers. In: Narboux, J., Schreck, P. and Streinu, E. (Eds.): *Proceedings of ADG 2016*, p. 137–143. 2016.
- [15] Quaresma, P.: Towards an Intelligent and Dynamic Geometry Book. Mathematics in Computer Science 11, p. 427–437. 2017.



- [16] Recio, T. and Vélez, M.P.: Automatic discovery of theorems in elementary geometry. *Journal of Automated Reasoning*, 23, p. 63–82. 1999.
- [17] Richard, P.R., Oller, A.M. and Meavilla, V.: The Concept of Proof in the Light of Mathematical Work. *ZDM – The International Journal on Mathematics Education*, 48(6), p. 843–859. 2016.
- [18] Sinclair, N., Bartolini Bussi, M.G., de Villiers, M. and Owens, K.: Recent research on geometry education: an ICME-13 survey team report. *ZDM Mathematics Education*, 48(5), p. 691–719. 2016.