Intersecting Chords Theorem

Lukas Bulwahn

December 14, 2021

Abstract

This entry provides a geometric proof of the intersecting chords theorem. The theorem states that when two chords intersect each other inside a circle, the products of their segments are equal.

After a short review of existing proofs in the literature [1, 3–5], I decided to use a proof approach that employs reasoning about lengths of line segments, the orthogonality of two lines and Pythagoras Law. Hence, one can understand the formalized proof easily with the knowledge of a few general geometric facts that are commonly taught in high-school.

This theorem is the 55th theorem of the Top 100 Theorems list.

Contents

1 Introduction

The intersecting chords theorem states:

> When two chords intersect each other inside a circle, the products of their segments are equal.

To prove this theorem in Isabelle, I reviewed existing formalizations in theorem provers and proofs in the literature [1, 3–5]. At the time of this AFP submission, the formalization of geometry in Isabelle is limited to only a few concepts and theorems. Hence, I selected to formalize the proof approach that fitted best to the already existing geometry formalizations.

The proof in HOL Light [3] simply unfolds the involved geometric predicates and then proves the theorem using only algebraic computations on

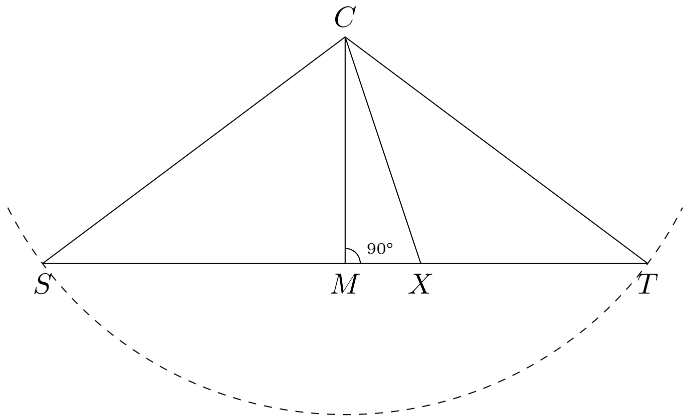

Figure 1: Key Lemma states $\overline{SX} \cdot \overline{XT} = \overline{SC}^2 - \overline{XC}^2$

real numbers. By a quick and shallow inspection of the proof script without executing the proof script step by step in HOL Light, I could not understand the proof script well enough to re-write the proof in Isabelle. As running the script in HOL Light seemed too involved to me, I ignored HOL Light's proof approach and considered the other approaches in the literature.

The first proof approach [5] that I found in the literature employs similarity of triangles, the inscribed angle theorem, and basic reasoning with angles. The intersecting chords theorem only consists of two reasoning steps after stating the geometric observations about angles. However, the proof requires to formalize the concept of similarity of triangles, extend the existing formalization of angles, and prove the inscribed angle theorem. So, I abandoned this proof approach and considered the second proof approach.

The second proof approach [5] needs only basic geometric reasoning about lengths of line segments, the orthogonality of two lines and Pythagoras Law. More specifically, one must prove that the line that goes through the apex and the midpoint of the base in an isosceles triangle is orthogonal to the base. This is easily derived from the property of an isosceles triangle using the congruence properties of triangles, which is already formalized in AFP's Triangle entry [2]. Furthermore, Pythagoras Law is a special case of the Law of Cosines, which is already formalized in AFP's Triangle entry.

Ultimately, I decided to use this second proof approach, which I sketch in more detail in the next subsection.

1.1 Informal Proof Sketch

The proof of the intersecting chords theorem relies on the following observation which is depicted in Figure 1.

Instead of considering *two* arbitrary chords intersecting, consider *one* arbitrary chord with endpoints S and T on a circle with center C and one arbitrary point X on the chord ST. This point X on the chord creates two line segments on this chord, the left part SX, and the right part XT. Without loss of generality, we can assume that SX is longer that XT, as shown in Figure 1.

The key lemma for the intersecting chords theorem provides a closed expression for the length of these two line segment using the distances of the chord endpoints and the point to the center C, i.e., the lemma states:

$$\overline{SX} \cdot \overline{XT} = \overline{SC}^2 - \overline{XC}^2.$$

To prove this fact, we consider the midpoint M of the chord ST. First, as M is the midpoint, $\overline{SM}$ and $\overline{TM}$ are equal. Second, we observe that the lengths of the line segments SX and XT are:

$$\overline{SX} = \overline{SM} + \overline{MX} \text{ and } \overline{XT} = \overline{TM} - \overline{MX} = \overline{SM} - \overline{MX}.$$

Third, the Pythagoras law for the triangles SMC and XMC states:

$$\overline{SM}^2 + \overline{MC}^2 = \overline{SC}^2 \text{ and } \overline{XM}^2 + \overline{MC}^2 = \overline{XC}^2.$$

Finally, the product can be expressed as:

$$\overline{SX} \cdot \overline{XT} = (\overline{SM} + \overline{MX}) \cdot (\overline{TM} - \overline{MX}) = \overline{SM}^2 - \overline{MX}^2 = (\overline{SC}^2 - \overline{MC}^2) - (\overline{XC}^2 - \overline{MC}^2) = \overline{SC}^2 - \overline{XC}^2.$$

The intersecting chord theorem now follows directly from this lemma: as the distances SC and XC for two arbitrary chords intersecting at X are equal, also the products of the chord segments are equal.

2 Intersecting Chord Theorem

theory *Chord-Segments*
imports *Triangle.Triangle*
begin

2.1 Preliminaries

lemma *betweenE-if-dist-leq*:
  fixes $A\ B\ X$ :: *'a::euclidean-space*
  assumes *between* $(A,\ B)\ X$
  assumes *dist* $A\ X \leq$ *dist* $B\ X$
  obtains u **where** $1\ /\ 2 \leq u\ u \leq 1$ **and** $X = u *_R A + (1 - u) *_R B$
proof (*cases* $A = B$)
  assume $A \neq B$
  from ‹*between* $(A,\ B)\ X$› **obtain** u **where** u: $u \geq 0\ u \leq 1$ **and** X: $X = u *_R A + (1 - u) *_R B$

    by (*metis add.commute betweenE between-commute*)
  from X **have** $X = B + u *_R (A - B)$ **and** $X = A + (u - 1) *_R (A - B)$
    by (*simp add*: *scaleR-diff-left real-vector.scale-right-diff-distrib*)+
  from ‹$X = B + u *_R (A - B)$› **have** *dist-B*: $dist\ B\ X = norm\ (u *_R (A - B))$
    by (*auto simp add*: *dist-norm*)
  from ‹$X = A + (u - 1) *_R (A - B)$› **have** *dist-A*: $dist\ A\ X = norm\ ((u - 1) *_R (A - B))$
    by (*auto simp add*: *dist-norm*)
  from ‹$A \neq B$› **have** $norm\ (A - B) > 0$ **by** *auto*
  from *this* ‹$dist\ A\ X \leq dist\ B\ X$› **have** $u \geq 1 / 2$
    using *dist-A dist-B* **by** *simp*
  from *this* ‹$u \leq 1$› X *that* **show** *thesis* **by** *blast*
next
  assume $A = B$
  define $u :: real$ **where** $u = 1$
  from ‹$between\ (A, B)\ X$› ‹$A = B$› **have** $1 / 2 \leq u$ $u \leq 1$ $X = u *_R A + (1 - u) *_R B$
    unfolding *u-def* **by** *auto*
  with *that* **show** *thesis* **by** *blast*
qed

lemma *dist-geq-iff-midpoint-in-between*:
  fixes $A\ B\ X :: 'a::euclidean\text{-}space$
  assumes $between\ (A, B)\ X$
  shows $dist\ A\ X \leq dist\ B\ X \longleftrightarrow between\ (X, B)\ (midpoint\ A\ B)$
proof
  assume $dist\ A\ X \leq dist\ B\ X$
  from ‹$between\ (A, B)\ X$› *this* **obtain** u
    where u: $1 / 2 \leq u$ $u \leq 1$ **and** X: $X = u *_R A + (1 - u) *_R B$
    using *betweenE-if-dist-leq* **by** *blast*
  have M: $midpoint\ A\ B = (1 / 2) *_R A + (1 / 2) *_R B$
    unfolding *midpoint-def* **by** (*simp add*: *scaleR-add-right*)
  from ‹$1 / 2 \leq u$› **have** 1: $midpoint\ A\ B = (1 / (2 * u)) *_R X + (1 - (1 / (2 * u))) *_R B$
  proof −
    have $(2 - u * 2) / (2 * u) = 1 / u - u / u$
      using $u(1)$ **by** (*simp add*: *diff-divide-distrib*)
    also have $\ldots = 1 / u - 1$ **using** $u(1)$ **by** *auto*
    finally have $(2 - u * 2) / (2 * u) = 1 / u - 1$ **.**
    from ‹$1 / 2 \leq u$› *this* **show** *?thesis*
      using X M **by** (*simp add*: *scaleR-add-right scaleR-add-left*[*symmetric*])
  qed
  moreover from u **have** 2: $(1 / (2 * u)) \geq 0$ $(1 / (2 * u)) \leq 1$ **by** *auto*
  ultimately show $between\ (X, B)\ (midpoint\ A\ B)$
    using *betweenI* [*of concl*: B X] **by** (*metis add.commute between-commute*)
next
  assume $between\ (X, B)\ (midpoint\ A\ B)$
  then have $between\ (A, midpoint\ A\ B)\ X$
    using ‹$between\ (A, B)\ X$› *between-midpoint*(*1*) *between-swap* **by** *blast*

then have *dist A X* $\leq$ *dist A* (*midpoint A B*)
using *between zero-le-dist* **by** *force*
also have *dist A* (*midpoint A B*) $\leq$ *dist B* (*midpoint A B*)
by (*simp add*: *dist-midpoint*)
also from ‹*between* (*X*, *B*) (*midpoint A B*)› **have** *dist B* (*midpoint A B*) $\leq$ *dist B X*
using *between zero-le-dist* **by** (*metis add.commute dist-commute le-add-same-cancel1*)
finally show *dist A X* $\leq$ *dist B X* **.**
qed

2.2 Properties of Chord Segments

lemma *chord-property*:
fixes *S C* :: *'a* :: *euclidean-space*
assumes *dist C S* = *dist C T*
assumes *between* (*S*, *T*) *X*
shows *dist S X* $*$ *dist X T* = (*dist C S*) ^ *2* − (*dist C X*) ^ *2*
proof −
define *M* **where** *M* = *midpoint S T*
have *between* (*S*, *T*) *M*
unfolding *M-def* **by** (*simp add*: *between-midpoint*(*1*))
have *dist T M* = *dist S M*
unfolding *M-def* **by** (*simp add*: *dist-midpoint*)

have *distances*: *max* (*dist S X*) (*dist X T*) = (*dist S M*) + (*dist X M*) ∧
min (*dist S X*) (*dist X T*) = (*dist S M*) − (*dist X M*)
proof *cases*
assume *dist S X* $\leq$ *dist X T*
then have *between* (*X*, *T*) *M*
using ‹*between* (*S*, *T*) *X*› *M-def*
by (*simp add*: *dist-geq-iff-midpoint-in-between dist-commute*)
then have *between* (*S*, *M*) *X*
using ‹*between* (*S*, *T*) *X*› ‹*between* (*S*, *T*) *M*› *between-swap* **by** *blast*
from ‹*between* (*X*, *T*) *M*› **have** *dist X T* = *dist X M* + *dist M T*
using *between* **by** *auto*
moreover from ‹*between* (*S*, *M*) *X*› **have** *dist S X* = *dist S M* − *dist M X*
using *between dist-commute* **by** *force*
ultimately show *?thesis*
using ‹*dist S X* $\leq$ *dist X T*› ‹*dist T M* = *dist S M*›
by (*simp add*: *add.commute dist-commute max-def min-def*)
next
assume ¬ (*dist S X* $\leq$ *dist X T*)
then have *dist T X* $\leq$ *dist S X* **by** (*simp add*: *dist-commute*)
then have *between* (*S*, *X*) *M*
using ‹*between* (*S*, *T*) *X*› *M-def*
by (*simp add*: *dist-geq-iff-midpoint-in-between midpoint-sym between-commute*)
then have *between* (*T*, *M*) *X*
using ‹*between* (*S*, *T*) *X*› ‹*between* (*S*, *T*) *M*› *between-swap between-commute* **by** *metis*

from ‹*between* (S, X) *M*› **have** *dist S X* = *dist S M* + *dist M X*
  using *between* **by** *auto*
moreover from ‹*between* (T, M) *X*› **have** *dist T X* = *dist T M* − *dist M X*
  using *between dist-commute* **by** *force*
ultimately show *?thesis*
  using ‹¬ *dist S X* ≤ *dist X T*› ‹*dist T M* = *dist S M*›
  by (*metis dist-commute max-def min-def*)
qed

have *orthogonal* $(C - M)$ $(S - M)$
  using ‹*dist C S* = *dist C T*› *M-def*
  by (*auto simp add*: *isosceles-triangle-orthogonal-on-midpoint*)
have *orthogonal* $(C - M)$ $(X - M)$
proof −
  have *between* (S, T) *M*
    using *M-def between-midpoint*(*1*) **by** *blast*
  obtain *c* **where** $(X - M) = c *_R (S - M)$
  proof (*cases* $S = M$)
    assume $S \neq M$
    then obtain *c* **where** $(X - M) = c *_R (S - M)$
      using *between-implies-scaled-diff* [*OF* ‹*between* (S, T) *X*› ‹*between* (S, T)
M›] **by** *metis*
    from *this that* **show** *thesis* **by** *blast*
  next
    assume $S = M$
    from *this* ‹*between* (S, T) *X*› **have** $X = M$
      by (*simp add*: *midpoint-between M-def*)
    from ‹$X = M$› ‹$S = M$› **have** $(X - M) = 0 *_R (S - M)$ **by** *simp*
    from *this that* **show** *thesis* **by** *blast*
  qed
  from *this* ‹*orthogonal* $(C - M)$ $(S - M)$› **show** *?thesis*
    by (*auto intro*: *orthogonal-clauses*(*2*))
qed
from ‹*orthogonal* $(C - M)$ $(S - M)$› ‹*orthogonal* $(C - M)$ $(X - M)$› **have**
  (*dist S M*) ^ *2* + (*dist M C*) ^ *2* = (*dist C S*) ^ *2*
  (*dist X M*) ^ *2* + (*dist M C*) ^ *2* = (*dist C X*) ^ *2*
  by (*auto simp only*: *Pythagoras*)
then have *geometric-observation*:
  (*dist S M*) ^ *2* = (*dist C S*) ^ *2* − (*dist M C*) ^ *2*
  (*dist X M*) ^ *2* = (*dist C X*) ^ *2* − (*dist M C*) ^ *2*
  by *auto*

have *dist S X* ∗ *dist X T* = *max* (*dist S X*) (*dist X T*) ∗ *min* (*dist S X*) (*dist X
T*)
  by (*auto split*: *split-max*)
also have ... = ((*dist S M*) + (*dist X M*)) ∗ ((*dist S M*) − (*dist X M*))
  using *distances* **by** *simp*
also have ... = (*dist S M*) ^ *2* − (*dist X M*) ^ *2*
  by (*simp add*: *field-simps power2-eq-square*)

```
  also have ... = ((dist C S) ^ 2 − (dist M C) ^ 2) − ((dist C X) ^ 2 − (dist
M C) ^ 2)
    using geometric-observation by simp
  also have ... = (dist C S) ^ 2 − (dist C X) ^ 2 by simp
  finally show ?thesis .
qed

theorem product-of-chord-segments:
  fixes S₁ T₁ S₂ T₂ X C :: 'a :: euclidean-space
  assumes between (S₁, T₁) X between (S₂, T₂) X
  assumes dist C S₁ = r dist C T₁ = r
  assumes dist C S₂ = r dist C T₂ = r
  shows dist S₁ X * dist X T₁ = dist S₂ X * dist X T₂
proof −
  from ‹dist C S₁ = r› ‹dist C T₁ = r› ‹between (S₁, T₁) X›
  have dist S₁ X * dist X T₁ = r ^ 2 − (dist C X) ^ 2
    by (subst chord-property) auto
  also from ‹dist C S₂ = r› ‹dist C T₂ = r› ‹between (S₂, T₂) X›
  have ... = dist S₂ X * dist X T₂
    by (subst chord-property) auto
  finally show ?thesis .
qed

end
```

References

[1] G. D. Birkhoff and R. Beatley. *Basic Geometry*, page 150. AMS Chelsea Publishing, Providence, Rhode Island, third edition, 1959.

[2] M. Eberl. Basic geometric properties of triangles. *Archive of Formal Proofs*, Dec. 2015. http://isa-afp.org/entries/Triangle.shtml, Formal proof development.

[3] J. Harrison. Theorem on product of segments of chords (in HOL light), 2010. https://github.com/jrh13/hol-light/blob/master/100/chords.ml.

[4] W. R. Knorr. *Textual Studies in Ancient and Medieval Geometry*, pages 34–36. Birkhäuser, Boston, first edition, 1989.

[5] Proofwiki. Intersecting chord theorem, 2014. https://proofwiki.org/wiki/Intersecting_Chord_Theorem [Online; accessed = 10-October-2016].