Geometry, Algebra and Computation

Wai-Shing Luk

Fudan University

April 25, 2019

Projective Geometry

Geometry and Algebra

- Geometry
  - Points, lines, triangles, circles, conic sections...
  - Collinear, concurrent, parallel, perpendicular...
  - Distances, angles, areas, quadrance, spread, quadrea...
  - Midpoint, bisector, orthocenter, pole/polar, tangent...
- Algebra
  - Addition, multiplication, inverse...
  - Elementary algebra: integer/rational/real/complex... numbers.
  - Abstract Algebra: rings, fields...
  - Linear algebra: vector, matrix, determinant, dot/cross product...
- Two subjects are related by coordinates.

Key points

- Our earth is non-flat and our universe is non-Euclidean.
- Non-Euclidean geometry is much easier to learn than you might think.
- Our curriculum in school is completely wrong.
- Euclidean geometry is non-symmetric. Three sides determine a triangle, but three angles do not determine a triangle. It might not be true in general geometries. Euclidean geometry is just a special case.
- Yet Euclidean geometry is more computationally efficient and is still used in our small-scale daily life.
- Incidenceship promotes integer arithmetic; non-oriented measurement promotes rational arithmetic; oriented measurement promotes floating-point arithmetic. Don't use a machine gun to hunt rabbit.

Projective Plane's Basic Elements

Projective Plane Concept

- Only involve "Points" and "Lines".
- "Points" (or "lines") are assumed to be distinguishable.
- Denote $A = B$ as A and B are referred to the same point.
- E.g., $(1/3, 2/3) = (10/30, 20/30)$
- We have the following rules:
  - $A = A$ (reflective)
  - If $A = B$, then $B = A$ (symmetric)
  - If $A = B$ and $B = C$, then $A = C$ (transitive)
- Unless mention specifically, objects in different names are assumed to be distinct, i.e. $A \neq B$.
- The idea can be generalized to higher dimensions. However, we restrict to 2D only here.

Incidence

- A point either lies on a line or not.
- If a point A lies on a line l, denote $l \circ A$.
- For convenience, we also denote as $A \circ l$.
- We have $A \circ l = l \circ A$

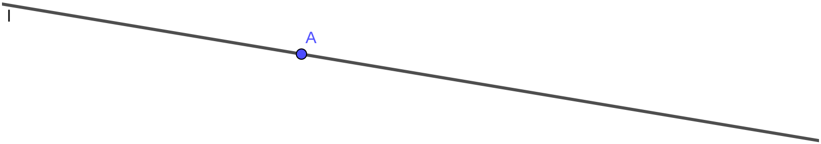


Figure 1: incident

Projective Point and Line

- Projective Point
  - Exactly one line passes through two distinct points.
  - Denote join(A, B) or simply AB as a line joined by A and B.
  - We have:
    - $AB = BA$
    - $AB \circ A$ and $AB \circ B$ are always true.
- Projective Line
  - Exactly one point met by two distinct lines.
  - Denote meet(l, m) or simply lm as a point met by l and m.
  - We have:
    - $lm = ml$
    - $lm \circ l$ and $lm \circ m$ are always true.
- Duality: "Point" and "Line" are interchangable here.
- "Projective geometry is all geometry." (Arthur Cayley)

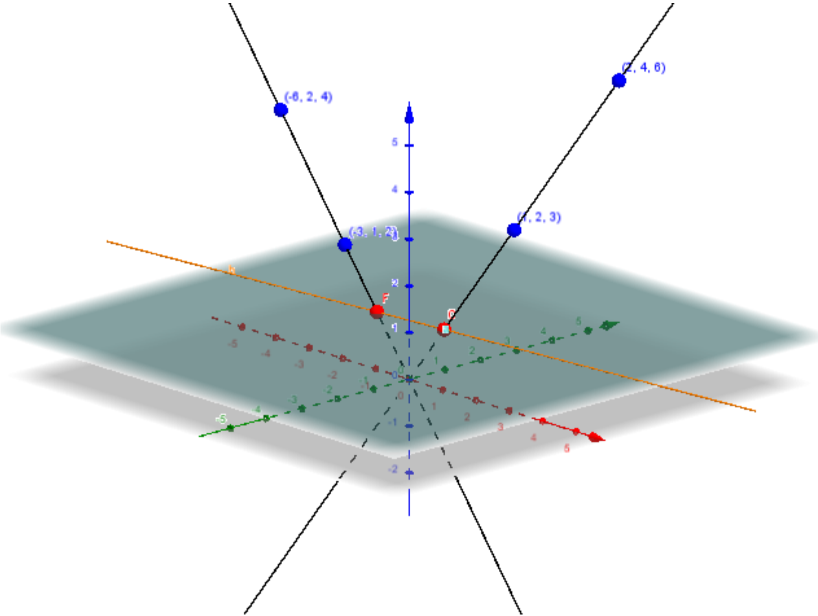
Example 1: Euclidean Geometry

- Point: projection of a 3D vector $p = [x, y, z]$ to 2D plane $z = 1$:

$$(x', y') = (x/z, y/z)$$

- $[\alpha x, \alpha y, \alpha z]$ for all $\alpha \neq 0$ are representing the same point.
- For instance, $[1, 5, 6]$ and $[-10, -50, -60]$ are representing the same point $(1/6, 5/6)$
- $p_\infty = [x, y, 0]$ is a point at *infinity*.
- Line: $ax' + by' + c = 0$, denoted by a vector $[a, b, c]$.
- $[\alpha a, \alpha b, \alpha c]$ for all $\alpha \neq 0$ are representing the same line.
- $l_\infty = [0, 0, 1]$ is the line at *infinity*.
- $[0, 0, 0]$ is not a valid point or line.

Euclidean 2D plane from 3D vector



Calculation by Vector Operations

- Let $v_1 = [x_1, y_1, z_1]$ and $v_2 = [x_2, y_2, z_2]$.
  - dot product $v_1 \cdot v_2 = v_1^\intercal v_2 = x_1 x_2 + y_1 y_2 + z_1 z_2$.
  - cross product $v_1 \times v_2 = [y_1 z_2 - z_1 y_2, -x_1 z_2 + z_1 x_2, x_1 y_2 - y_1 x_2]$
- Then, we have:
  - $A \circ a$ if and only if $[A] \cdot [a] = 0$
  - Join of two points: $[AB] = [A] \times [B]$
  - Meet of two lines: $[lm] = [l] \times [m]$
  - $A = B$ if and only if $[A] \times [B] = [0, 0, 0]$

Examples

- The linear equation that joins the point $(1/2, 3/2)$ and $(4/5, 3/5)$ is $[1, 3, 2] \times [4, 3, 5] = [9, 3, -9]$, or $9x + 3y - 9 = 0$, or $3x + y = 3$.
- The point $(1/2, 3/2)$ lies on the line $3x + y = 3$ because $[1, 3, 2] \cdot [3, 1, -3] = 0$.
- Exercise: Calculate the line equation that joins the points $(5/8, 7/8)$ and $(-5/6, 1/6)$.

Python Code (pg_object)

```
class pg_object(np.ndarray):
    @abstractmethod
    def dual(self):
        """abstract method"""
        pass

    def __new__(cls, inputarr):
*        obj = np.asarray(inputarr).view(cls)
        return obj

    def __eq__(self, other):
        if type(other) is type(self):
            return (np.cross(self, other) == 0).all()
        return False

    def __ne__(self, other):
        return not self.__eq__(other)

    def incident(self, l):
        return not self.dot(l)

    def __mul__(self, other):
        T = self.dual()
        return T(np.cross(self, other))
```

Python Code (pg_point and pg_line)

```
class pg_point(pg_object):
    def __new__(cls, inputarr):
        obj = pg_object(inputarr).view(cls)
        return obj

    def dual(self):
        return pg_line

class pg_line(pg_object):
    def __new__(cls, inputarr):
        obj = pg_object(inputarr).view(cls)
        return obj

    def dual(self):
        return pg_point

def join(p, q):
    assert isinstance(p, pg_point)
    return p * q

def meet(l, m):
    assert isinstance(l, pg_line)
    return l * m
```

Python Code (Example)

```
from __future__ import print_function
from pprint import pprint
import numpy as np

if __name__ == "__main__":
    p = pg_point([1, 3, 2])
    q = pg_point([4, 3, 5])
    print(join(p, q))

    l = pg_line([5, 7, 8])
    m = pg_line([-5, 1, 6])
    print(meet(l, m))

    p = pg_point([1-2j, 3-1j, 2+1j]) # complex number
    q = pg_point([-2+1j, 1-3j, -1-1j])
    assert p.incident(p*q)
```

Example 2: Perspective View of Euclidean Geometry

- It turns out that we can choose any line on a plane as the line of infinity.

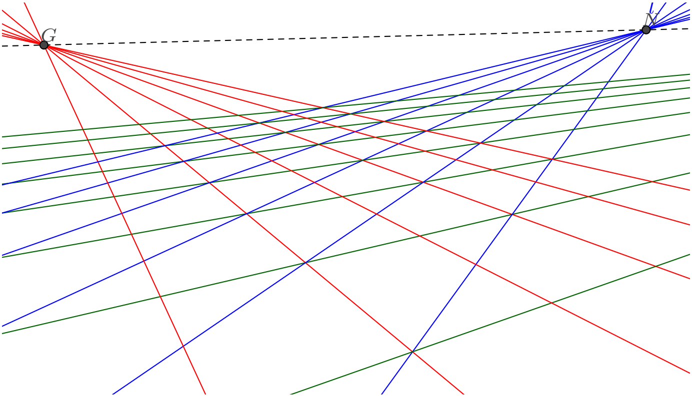


Figure 3: euclidean2

Example 3: Spherical/Elliptic Geometry

- Surprisingly, the vector notations and operators can also represent other geometries such as spherical/Elliptic geometry.
- "Point": projection of 3D vector $[x, y, z]$ to the unit sphere.

$$(x', y', z') = (x/r, y/r, z/r)$$

  where $r^2 = x^2 + y^2 + z^2$.
- Two points on the opposite poles are considered the same point here.
- "Line": $[a, b, c]$ represents the *great circle* intersected by the unit sphere and the plane $ax + by + cz = 0$.
- $[x, y, z]$ is called *Homogeneous Coordinates*.
- Here, the coordinates could be in integer numbers, rational numbers (ratio of two integers), real numbers, complex numbers, or finite field numbers, or even polynomial functions.

Spherical Geometry from 3D vector

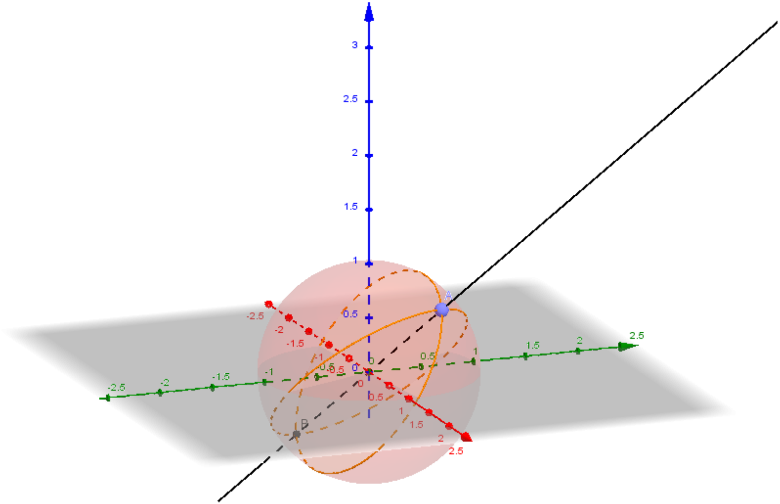


Figure 4: sphere

Example 4: Hyperbolic Geometry from 3D vector

- A velocity "point": projection of a 3D vector $[p] = [x, y, t]$ to 2D plane $t = 1$:
$$(v_x, v_y) = (x/t, y/t)$$

Counter-examples

- In some quorum systems, two “lines” are allowed to meet at more than one points. Therefore, only the very special case it is a projective geometry.
- In some systems, a line from A to B is not the same as the line from B to A, so they cannot form a projective geometry.
- “Symmetry” is an important keyword in projective geometry.

Number systems

- Integer number ($\mathbb{Z}$):
  - e.g. $0, 1, 2, 3, \ldots, -1, -2, -3, \ldots$
  - discrete, more computationally efficient.
- Rational number ($\mathbb{Q}[\mathbb{Z}]$):
  - e.g. $0/1, 2/3, -1/3, 1/0$ (i.e. infinity)
  - Multiplication/division is easier than addition/subtraction
- Real number ($\mathbb{R}$):
  - e.g. 0.3, $2^{1/2}$, π
  - May induce round-off errors.
- Finite field, $GF(n)$, where n is a prime number (e.g. $2, 3, 5, 7, 11, 13$) or prime powers (e.g. $4 = 2^2, 8 = 2^3, 9 = 3^2$).
  - Used in Coding Theory

Number systems (cont'd)

- Complex number ($\mathbb{C}$):
  - e.g $1 + \pi i$, $1 - 3\pi i$
  - Besides the identity (the only automorphism of the real numbers), there is also the automorphism τ that sends $x + iy$ to $x - iy$ such that $\tau(\tau(x)) = x$.
- Complex number over integer ($\mathbb{C}[\mathbb{Z}]$)
  - e.g. $1 + 2i$, $1 - 2i$
  - Also known as Gaussian integer.
- Complex number over Rational ($\mathbb{C}[\mathbb{Q}]$)
- Projective Geometry can work on all these number systems.
- In fact, Projective Geometry can work on any field number. Moreover, the multiplicative inverse is not required.
- "Continuity" is not assumed in Projective Geometry.

Example 4: Poker Card Geometry

- Even "coordinates" is **not** a necessary requirement of projective geometry.
- Consider the poker cards in Table 1:
  - For example, $\text{meet}(l_2, l_5) = \texttt{5}$, $\text{join}(\texttt{J}, \texttt{4}) = l_8$.
- We call this *Poker Card Geometry* here.

Table 1: Poker Card Geometry

l_1	l_2	l_3	l_4	l_5	l_6	l_7	l_8	l_9	l_{10}	l_{11}	l_{12}	l_{13}
A	2	3	4	5	6	7	8	9	10	J	Q	K
2	3	4	5	6	7	8	9	10	J	Q	K	A
4	5	6	7	8	9	10	J	Q	K	A	2	3
10	J	Q	K	A	2	3	4	5	6	7	8	9

Finite projective plane

- Yet we may assign the homogeneous coordinate to a finite projective plane, where the vector operations are in a finite field.
- E.g. The poker card geometry is a finite projective plane of order 3.
- The smallest finite projective plane (order 2) contains only 7 points and 7 lines.
- If the order is a prime number or prime powers, then we can easily construct the finite projective plane via finite field and homogeneous coordinate.
- The non-existence of finite projective plane of order 10 was proved in 1989. The proof took the equivalent of 2000 hours on a Cray 1 supercomputer.
- The existences of many other higher order finite projective planes are still an open question.

Not covered in this work

- Unless mention specifically, we don't discuss finite projective plane further more.
- Although the coordinate system is not a requirement in general projective geometry, practically all examples we are dealing with have homogeneous coordinates. All the proofs of theorems are based on the assumption of homogeneous coordinates.

Basic Properties

Collinear, Concurrent, and Coincidence

- Three points are called *collinear* if they all lie on the same line.
- Three lines are called *concurrent* if they all meet at the same point.
- Denote coincidence relation as $\mathrm{coI}(A, B, C)$.
- $\mathrm{coI}(A, B, C)$ is true if and only if $AB \circ C$ is true.
- Similarly, $\mathrm{coI}(a, b, c)$ is true if and only if $ab \circ c$ is true.
- In general, $\mathrm{coI}(A_1, A_2, \ldots, A_n)$ is true if and only if $A_1A_2 \circ X$ is true for all X in the rest of points $A_3, A_4, \ldots, A_n$.
- Unless mention specifically, $ABCD\ldots$ is assumed to be coincidence, while $\{ABCD\ldots\}$ is assumed none of three are coincident.

Parameterize a line

- The points on the line AB can be parameterized by $\lambda[A] + \mu[B]$ with λ and μ are not both zero.
- For integer coordinates, to show that $\lambda[A] + \mu[B]$ can span all the integer points on the line, we give the exact expression of λ/μ of a point C as follows.
- Let $l = [C] \times ([A] \times [B])$.
- Then
$$\lambda[A] + \mu[B] = (l^\mathsf{T}[B])[A] - (l^\mathsf{T}[A])[B]$$

Python Code

```python
def coincident(p, q, r):
    return r.incident(p * q)

def coI_core(l, Lst):
    for p in Lst:
        if not l.incident(p):
            return False
    return True

def coI(p, q, *rest):
    assert p != q
    return coI_core(p*q, rest )

# Note: `lambda` is a preserved keyword in python
def plucker(lambda1, p, mu1, q):
    T = type(p)
    return T(lambda1 * p + mu1 * q)
```

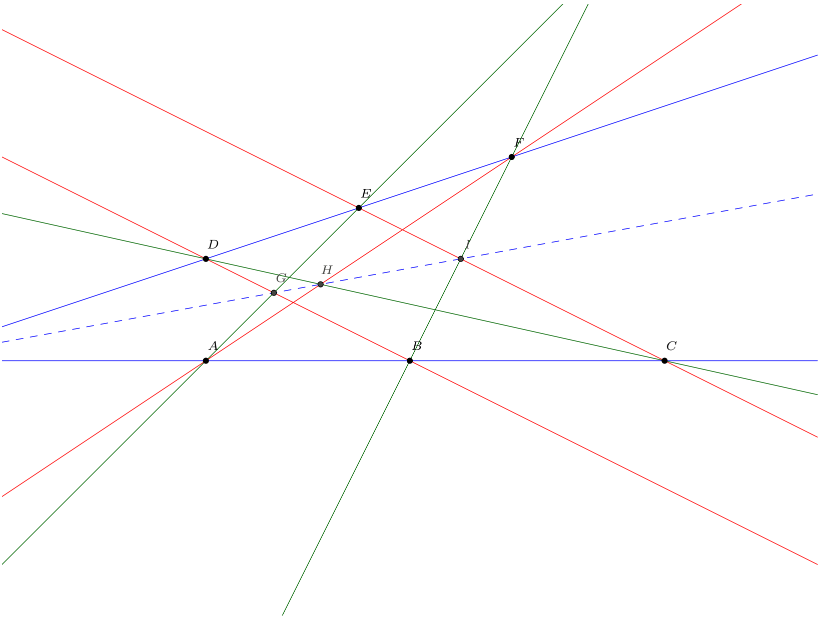
Pappus Theorem

- Theorem (Pappus): Given two lines ABC and DEF. Let G=meet(AE, BD), H=meet(AF, CD), I=meet(BF, CE). Then G, H, I are collinear.
- Sketch of the *proof*:
  - Let $[C] = \lambda_1[A] + \mu_1[B]$.
  - Let $[F] = \lambda_2[D] + \mu_2[E]$.
  - Express $[G], [H], [I]$ in terms of $[A], [B], \lambda_1, \mu_1, \lambda_2, \mu_2$.
  - Simplify the expression $[G] \cdot ([H] \times [I])$ and derive that it is equal to 0. (we may use the Python's symbolic package for the calculation.)
- Exercise: verify that this theorem holds for the poker card geometry with `3`, `6`, `Q` on l_3 and `8`, `9`, `J` on l_8.

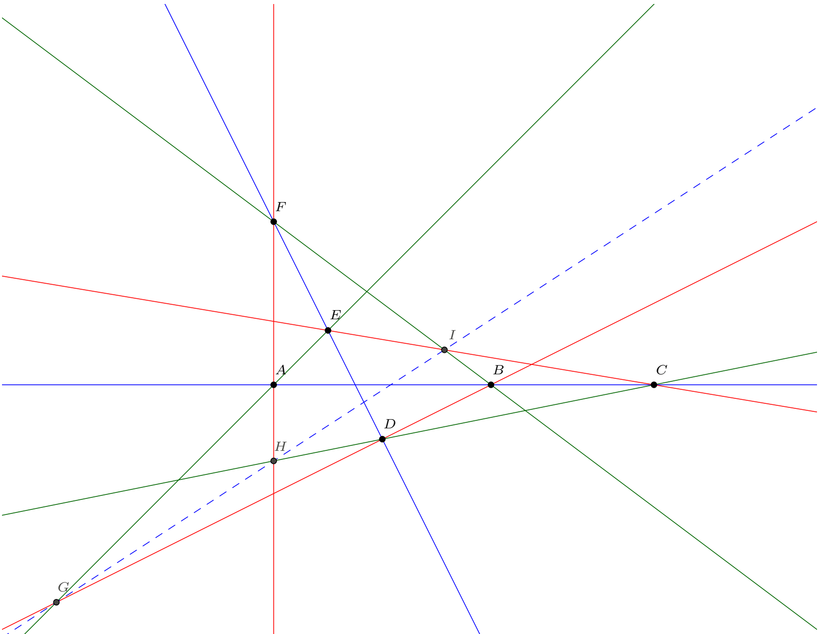
Python Code for the Proof

```
import sympy
sympy.init_printing()
pv = sympy.symbols("p:3", integer=True)
qv = sympy.symbols("q:3", integer=True)
lambda1, mu1 = sympy.symbols("lambda1 mu1", integer=True)
p = pg_point(pv); q = pg_point(qv)
r = plucker(lambda1, p, mu1, q)
sv = sympy.symbols("s:3", integer=True)
tv = sympy.symbols("t:3", integer=True)
lambda2, mu2 = sympy.symbols("lambda2 mu2", integer=True)
s = pg_point(sv); t = pg_point(tv)
u = plucker(lambda2, s, mu2, t)
G = (p * t) * (q * s)
H = (p * u) * (r * s)
I = (q * u) * (r * t)
ans = np.dot(G, H * I)
ans = sympy.simplify(ans)
print(ans) # get 0
```

An instance of Pappus' theorem



Another instance of Pappus' theorem



Triangles and Trilaterals

- If three points A, B, and C are not collinear, they form a triangle, denoted as $\{ABC\}$.
- If three lines a, b, and c are not concurrent, they form a trilateral, denoted as $\{abc\}$.
- Triangle $\{ABC\}$ and trilateral $\{abc\}$ are dual if $A = bc$, $B = ac$ and $C = ab$.

Python Code (II)

```python
def tri(T):
    a1, a2, a3 = T
    l1 = a2 * a3
    l2 = a1 * a3
    l3 = a1 * a2
    return l1, l2, l3

def tri_func(func, T):
    a1, a2, a3 = T
    m1 = func(a2, a3)
    m2 = func(a1, a3)
    m3 = func(a1, a2)
    return m1, m2, m3
```

An example of triangle and trilateral

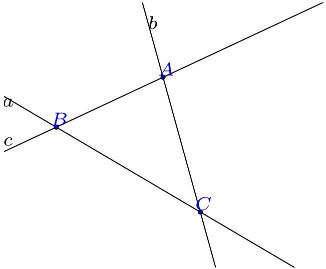


Figure 7: Triangle

Projectivities and Perspectivities

Projectivities

- An ordered set (A, B, C) (either collinear or not) is called a projective of a concurrent set abc if and only if $A \circ a$, $B \circ b$ and $C \circ c$.
- Denote this as $(A, B, C) \barwedge abc$.
- An ordered set (a, b, c) (either concurrent or not) is called a projective of a collinear set ABC if and only if $A \circ a$, $B \circ b$ and $C \circ c$.
- Denote this as $(a, b, c) \barwedge ABC$.
- If each ordered set is coincident, we may write:
  - $ABC \barwedge abc \barwedge A'B'C'$
  - Or simply $ABC \barwedge A'B'C'$

Perspectivities

- An ordered set (A, B, C) is called a perspectivity of an ordered set (A', B', C') if and only if $(A, B, C) \barwedge abc$ and $(A', B', C') \barwedge abc$ for some concurrent set abc.
- Denote this as $(A, B, C) \doublebarwedge (A', B', C')$.
- An ordered set (a, b, c) is called a perspectivity of an ordered set (a', b', c') if and only if $(a, b, c) \barwedge ABC$ and $(a', b', c') \barwedge ABC$ for some collinear set ABC.
- Denote this as $(a, b, c) \doublebarwedge (a', b', c')$.

An instance of perspectivity

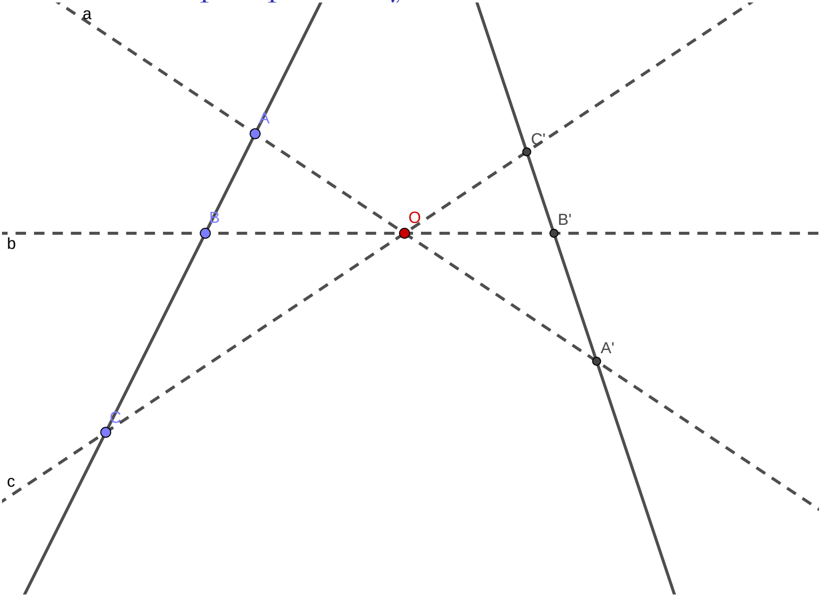


Figure 8: perspectivity

Another instance of perspectivity

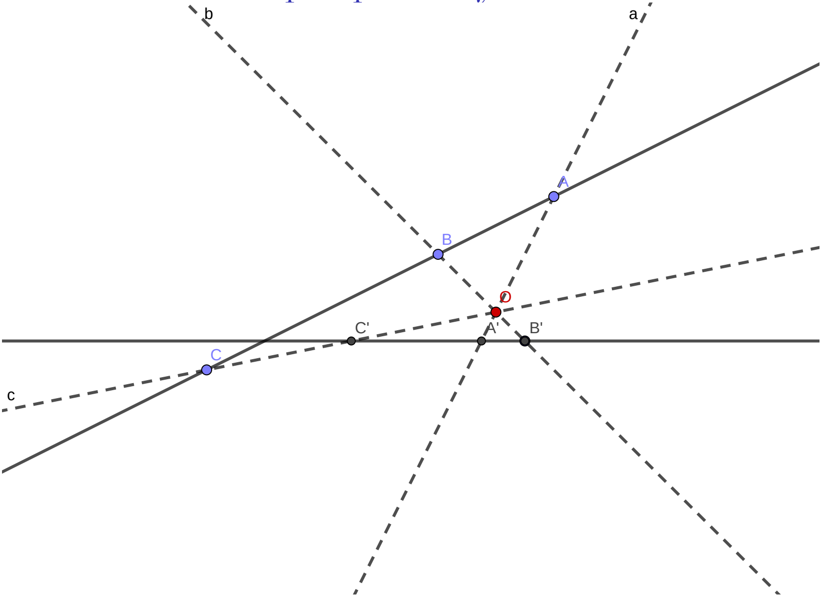


Figure 9: perspecitivity2

Perspectivity

- Similar definition for more than three points:
  - $(A_1, A_2, A_3, \ldots, A_n) \stackrel{=}{\wedge} (A'_1, A'_2, A'_3, \ldots, A'_n)$.
- To check perspectivity:
  - First construct a point $O := \text{meet}(A_1A'_1, A_2A'_2)$.
  - For the rest of points, check if X, X', O are collinear.
- Note that $(A, B, C) \stackrel{=}{\wedge} (D, E, F)$ and $(D, E, F) \stackrel{=}{\wedge} (G, H, I)$ does not imply $(A, B, C) \stackrel{=}{\wedge} (G, H, I)$.

Python Code (III)

```
def persp(L, M):
    if len(L) != len(M):
        return False
    if len(L) < 3:
        return True
    [pL, qL] = L[0:2]
    [pM, qM] = M[0:2]
    assert pL != qL
    assert pM != qM
    assert pL != pM
    assert qL != qM
    O = (pL * pM) * (qL * qM)
    for rL, rM in zip(L[2:], M[2:]):
        if not O.incident(rL * rM):
            return False
    return True
```

Desargues's Theorem

- Theorem (Desargues): Let trilateral $\{abc\}$ be the dual of triangle $\{ABC\}$ and trilateral $\{a'b'c'\}$ be the dual of triangle $\{A'B'C'\}$. Then, $\{ABC\} \overline{\overline{\wedge}} \{A'B'C'\}$ if and only if $\{abc\} \overline{\overline{\wedge}} \{a'b'c'\}$.
- Sketch of the *proof*:
  - Let O be the perspective point.
  - Let $[A'] = \lambda_1[A] + \mu_1[O]$.
  - Let $[B'] = \lambda_2[B] + \mu_2[O]$.
  - Let $[C'] = \lambda_3[C] + \mu_3[O]$.
  - Let $[G] = ([A] \times [B]) \times ([A'] \times [B'])$
  - Let $[H] = ([B] \times [C]) \times ([B'] \times [C'])$
  - Let $[I] = ([A] \times [C]) \times ([A'] \times [C'])$
  - Express $[G], [H], [I]$ in terms of $[A], [B], [C], [O], \lambda_1, \mu_1, \lambda_2, \mu_2, \lambda_3, \mu_3$.
  - Simplify the expression $[G] \cdot ([H] \times [I])$ and find that it is equal to 0. (we may use the Python's symbolic package for the calculation.)
  - Due to the duality, the only-if part can be proved using the same technique.

Python Code for the Proof (II)

```
# Define symbol points p, q, s, t as before
# Define symbol lambda1, mu1, lambda2, mu2 as before
# ...
lambda3, mu3 = sympy.symbols("lambda3 mu3", integer=True)
p2 = plucker(lambda1, p, mu1, t)
q2 = plucker(lambda2, q, mu2, t)
s2 = plucker(lambda3, s, mu3, t)
G = (p * q) * (p2 * q2)
H = (q * s) * (q2 * s2)
I = (s * p) * (s2 * p2)
ans = np.dot(G, H * I)
ans = sympy.simplify(ans)
print(ans) # get 0
```

An instance of Desargues's theorem

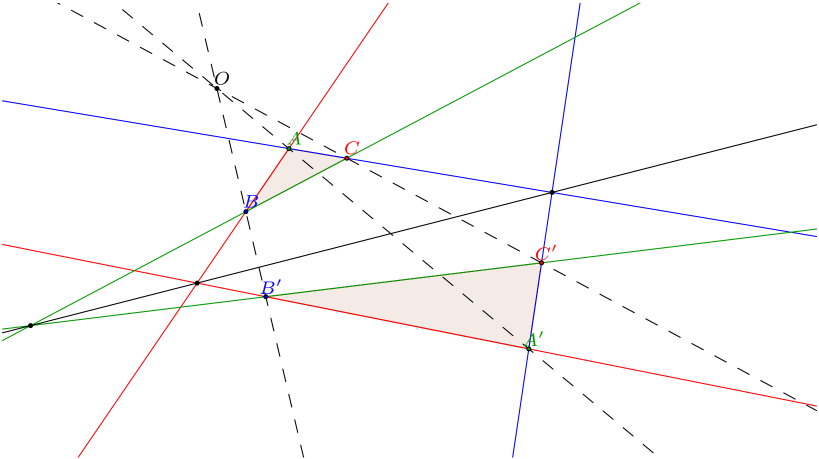


Figure 10: desargues

Another instance of Desargues's theorem

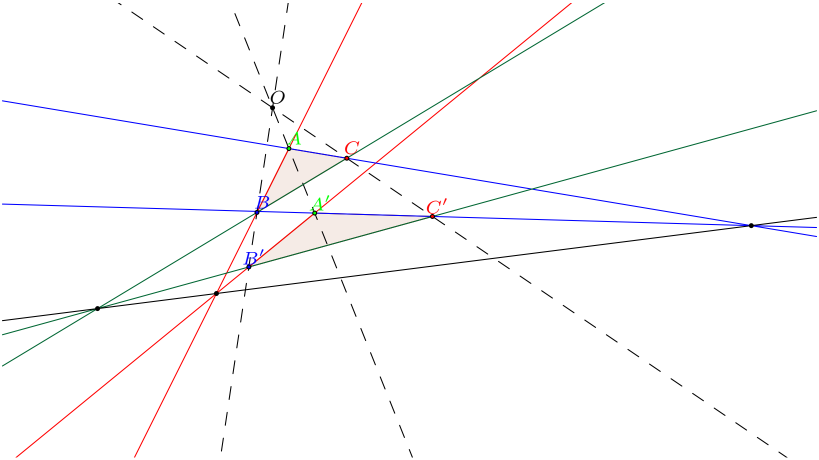


Figure 11: desargues2

Projective Transformation

- Given a function τ that transforms a point A to another point $\tau(A)$.
- If A, B, and C are collinear and we always have $\tau(A)$, $\tau(B)$, and $\tau(C)$ collinear. Then the function τ is called a projective transformation.
- In Homogeneous coordinate, a projective transformation is any non-singular matrix times a vector.

Quadrangles and Quadrilateral Sets

- If four points P, Q, R and S none of three are collinear, they form a quadrangle, denoted as $\{PQRS\}$.
- Note that Quadrangle here could be convex or self-intersecting.
- Totally there are six lines formed by $\{PQRS\}$.
- Assume they meet another line l at A, B, C, D, E, F, where
  - $A = \text{meet}(PQ, l)$, $D = \text{meet}(RS, l)$
  - $B = \text{meet}(QR, l)$, $E = \text{meet}(PS, l)$
  - $C = \text{meet}(PR, l)$, $F = \text{meet}(QS, l)$
- We call the six points as a quadrilateral set, denoted as $(AD)(BE)(CF)$.

Quadrilateral set

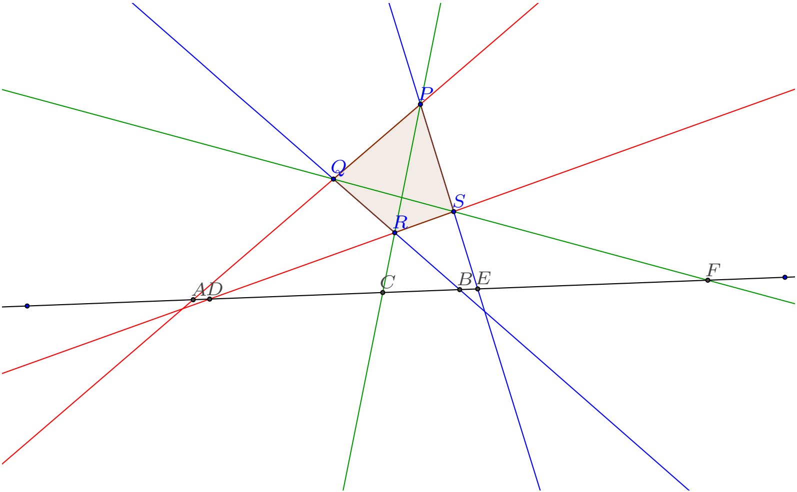


Figure 12: quad_set

Another quadrilateral set

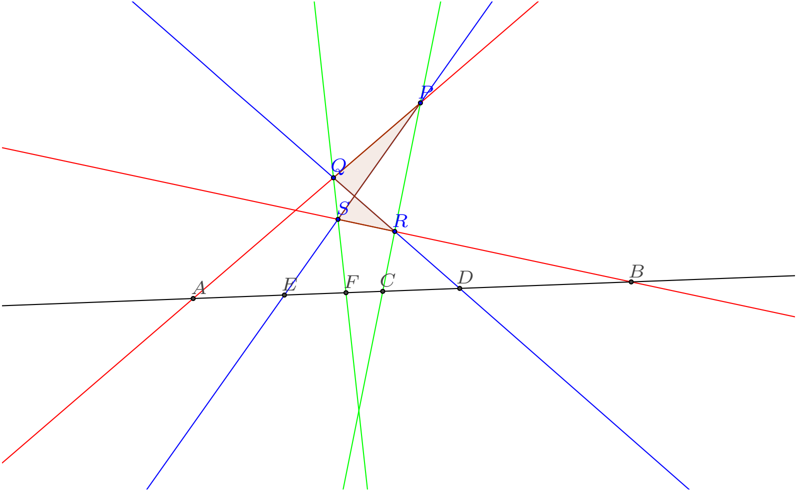


Figure 13: quad_set2

Harmonic Sets

- In a quadrilateral set $(AD)(BE)(CF)$, if $A = D$ and $B = E$, then it is called a harmonic set.
- The Harmonic relation is denoted by $H(AB, CF)$.
- Then C and F is called a harmonic conjugate.
- Theorem: If $ABCF \barwedge abcd$, then $H(AB, CF) = H(ab, cf)$.
- In other words, projectivity preserves harmonic relation.
- Theorem: If $ABCF \doublebarwedge A'B'C'F'$, then $H(AB, CF) = H(A'B', C'F')$.
- In other words, perspectivity preserves harmonic relation.

Basic measure between point and line

- A basic measure between p and l, denoted by $p \cdot l$ (inner product):
  - $p \cdot l$ can be positive, negative, and zero.
  - $p \cdot l = 0$ if and only if p lies on l.

Cross Ratio

- Given a line incident with $ABCD$. Arbitrary choose a point O not on the line.
- The cross ratio is defined as:

$$R(A, B; C, D) = (OA \cdot C)(OB \cdot D)/(OA \cdot D)(OB \cdot C)$$

- Note: the cross ratio does not depend on what O is chosen.

Python Code (IV)

```
from fractions import Fraction
import numpy as np

def ratio_ratio(a, b, c, d):
    if isinstance(a, (int, np.int64)):
        return Fraction(a, b) / Fraction(c, d)
    return (a * d) / (b * c)

def x_ratio(A, B, l, m):
    dAl = A.dot(l)
    dAm = A.dot(m)
    dBl = B.dot(l)
    dBm = B.dot(m)
    return ratio_ratio(dAl, dAm, dBl, dBm)

def R(A, B, C, D):
    O = (C*D).aux()
    return x_ratio(A, B, O*C, O*D)
```

Polarities

- A *polarity* is a projective correlation of period 2.
- We call a the *polar* of A, and A the pole of a.
- Denote $a = A^{\perp}$ and $A = a^{\perp}$.
- Except degenerate cases, $A = (A^{\perp})^{\perp}$ and $a = (a^{\perp})^{\perp}$.
- It may happen that A is incident with a so that each is *self-conjugate*.
- The locus of self-conjugate points defines a *conic*. However, the polarity is a more general concept than a conic, because some polarities may not have self-conjugate points (or their self-conjugate points are complex).

The Use of a Self-Polar Triangle

- Any projective correlation that relates the three vertices of one triangle to the respectively opposite sides is a polarity.
- Thus, any triangle $\{ABC\}$, any point P not on a side, and any line p not throughout a vertex, determine a definite polarity $(ABC)(Pp)$.

The Conic

- Historically *ellipse* (including *circle*), *parabola*, and *hyperbola*.
- The locus of self-conjugate points is a *conic*.
- Their polars are its *tangents*.
- Any other line is called a *secant* or a *nonsecant* according to as it meets the conic twice or not at all, i.e., according to as the involution of conjugate points on it is hyperbolic or elliptic.
- Note: Intersecting a conic with a line may result of an irrational intersection point.

Construct the polar of a point using a conic

- To construct the polar of a given point C, not on the conic, draw any two secants PQ and RS through C; then the polar joins the two points meet(QR, PS) and meet(RP, QS).

Example of constructing the polar of a point

Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

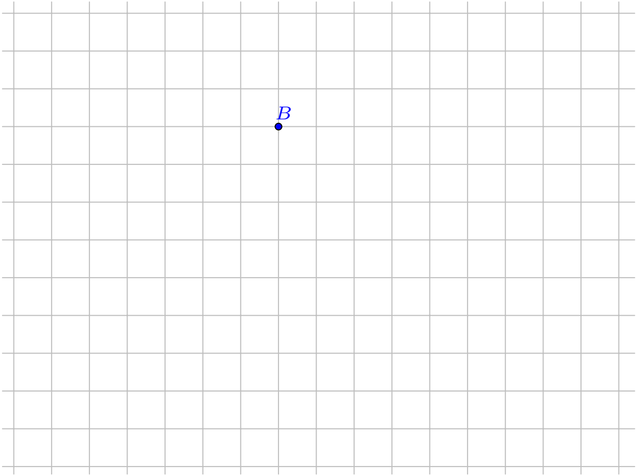


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

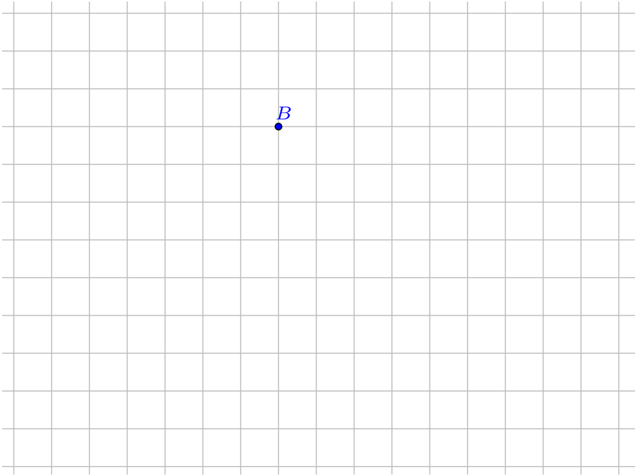


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

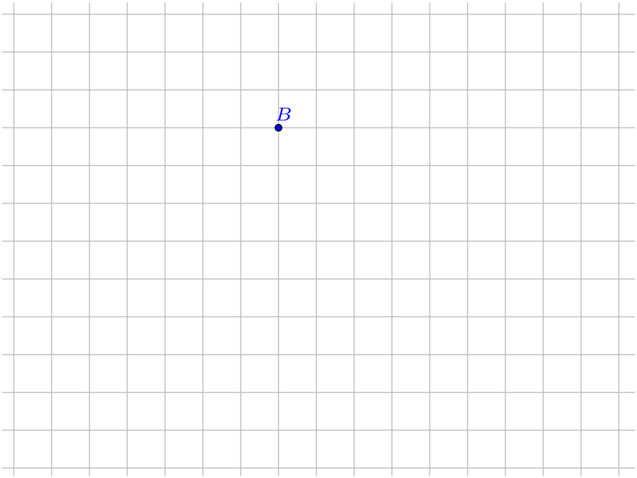


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

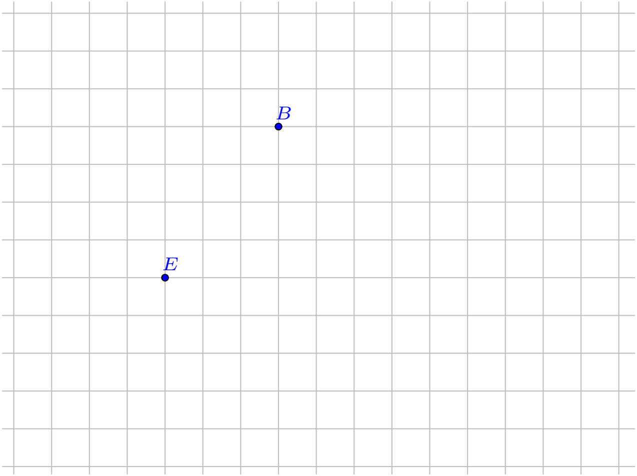


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

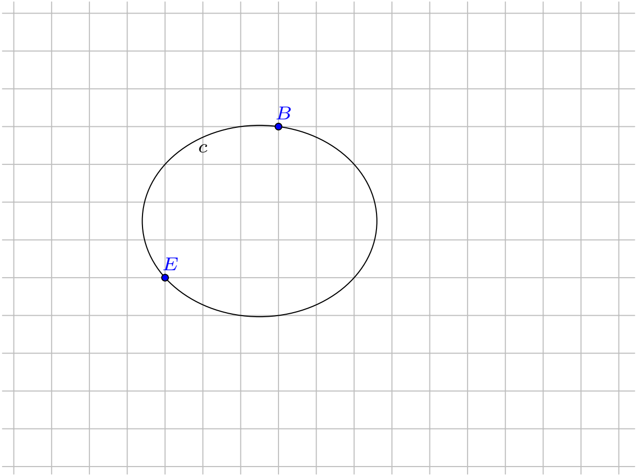


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

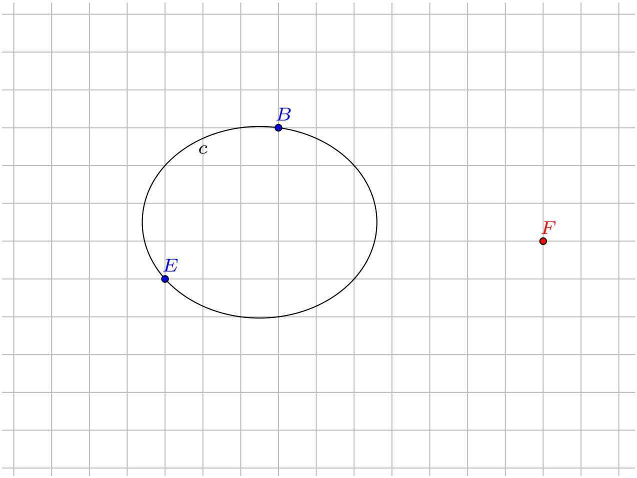


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

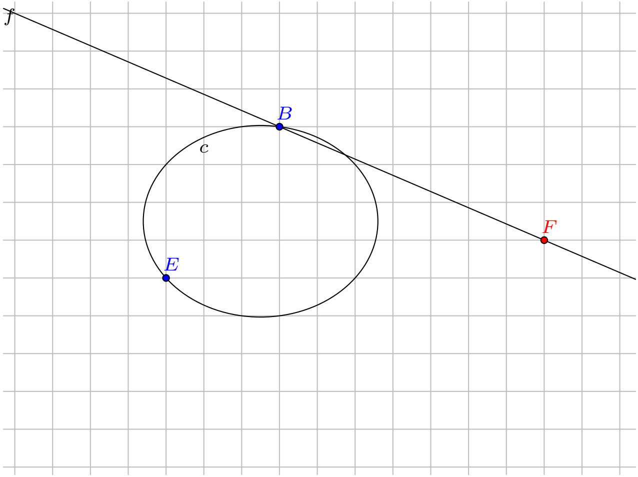


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

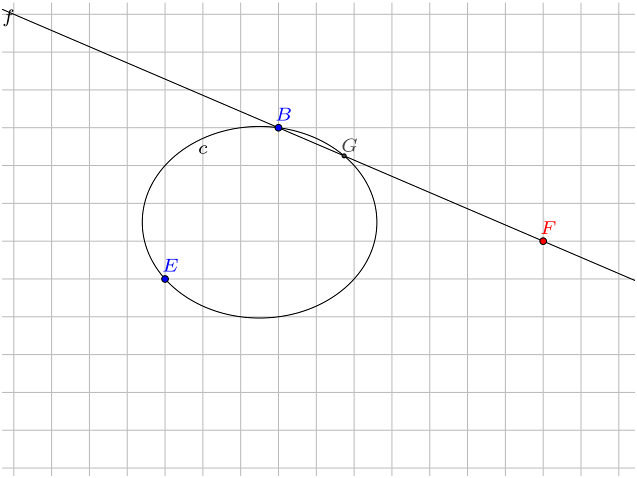


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

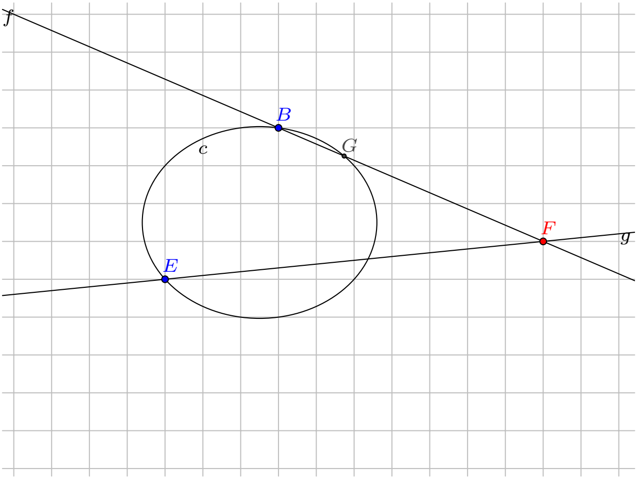


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

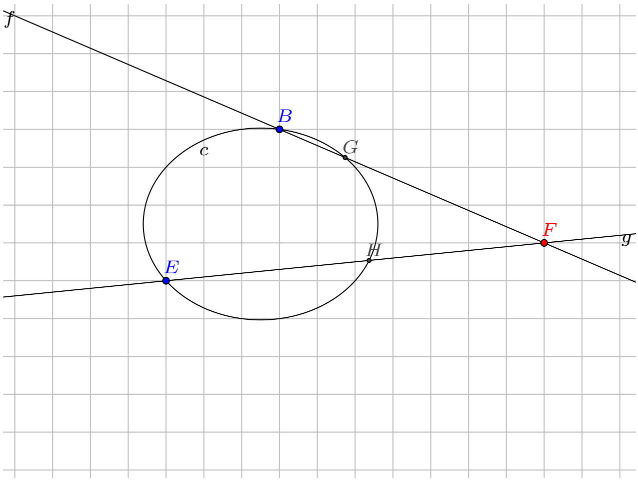


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

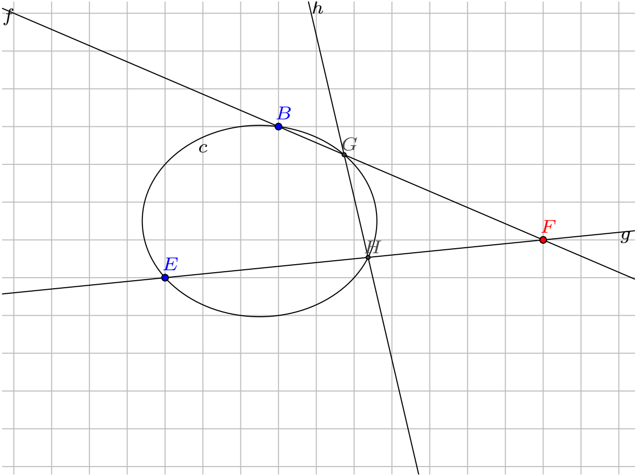


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

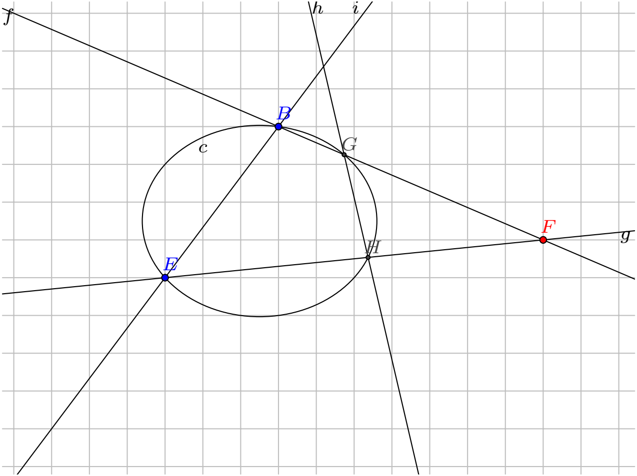


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

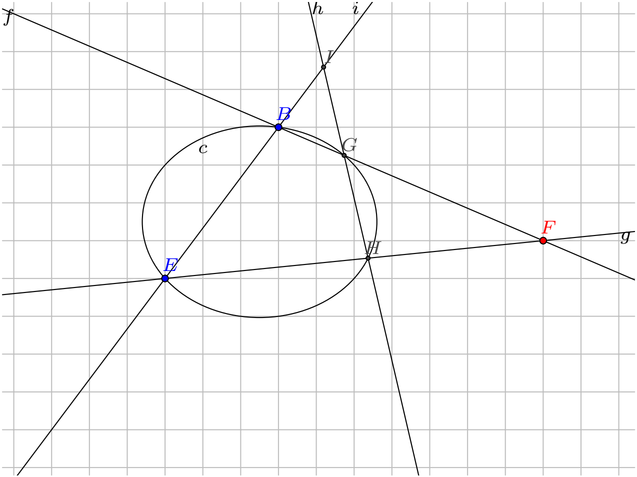


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

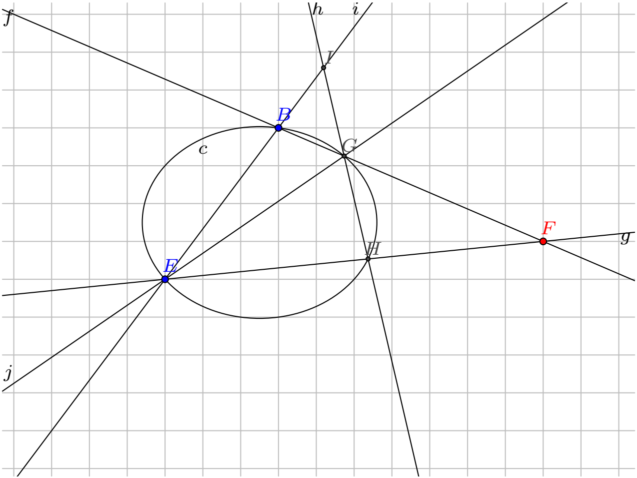


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

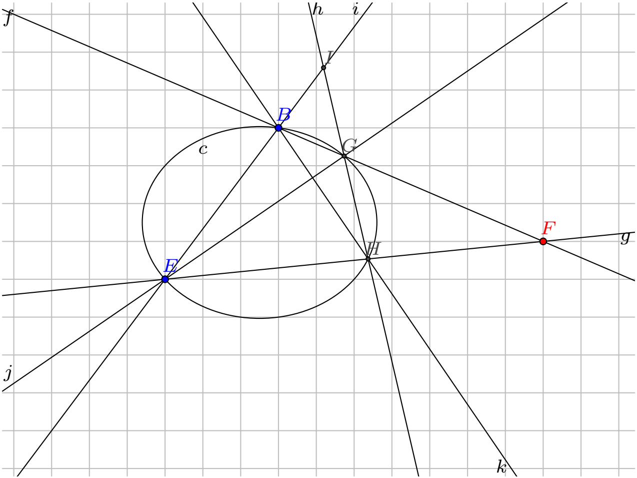


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

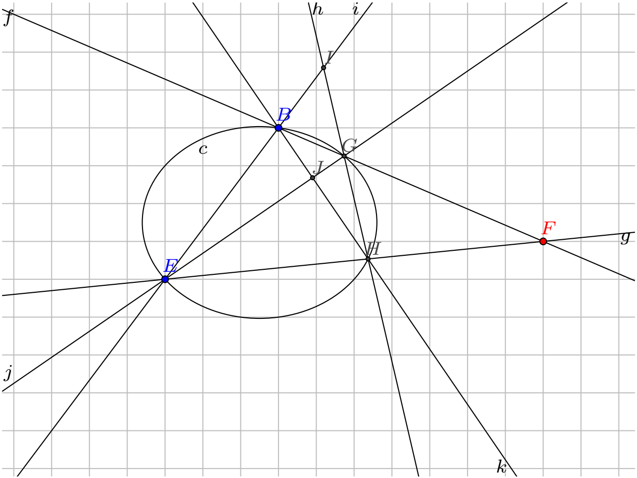


Figure 14: Example of constructing the polar of a point

Example of constructing the polar of a point

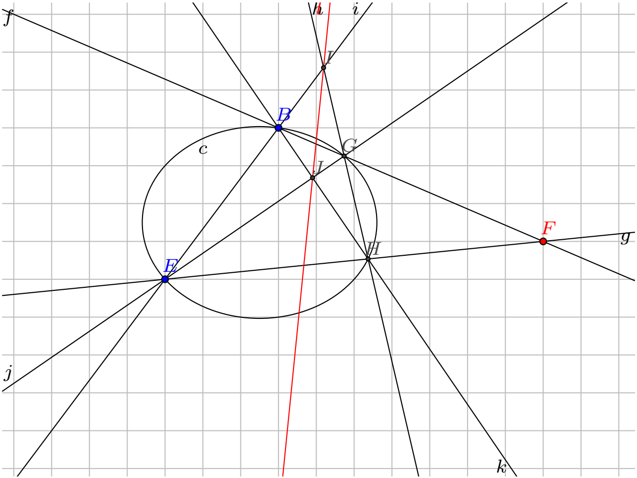


Figure 14: Example of constructing the polar of a point

Another example of constructing the polar of a point

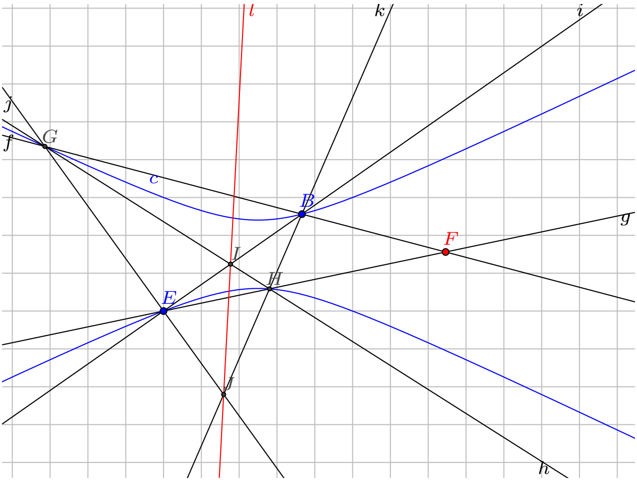


Figure 15: Another example of constructing the polar of a point

Construct the pole from a line

- To construct the pole of a given secant a, draw the polars of any two points on the line; then the common point of two polars is the pole of a.

Constructing the pole of a line

Figure 16: Constructing the pole of a line

Constructing the pole of a line

Figure 16: Constructing the pole of a line

Constructing the pole of a line

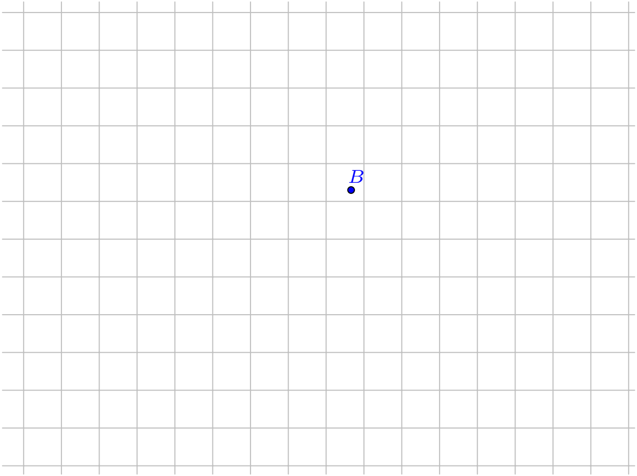


Figure 16: Constructing the pole of a line

Constructing the pole of a line

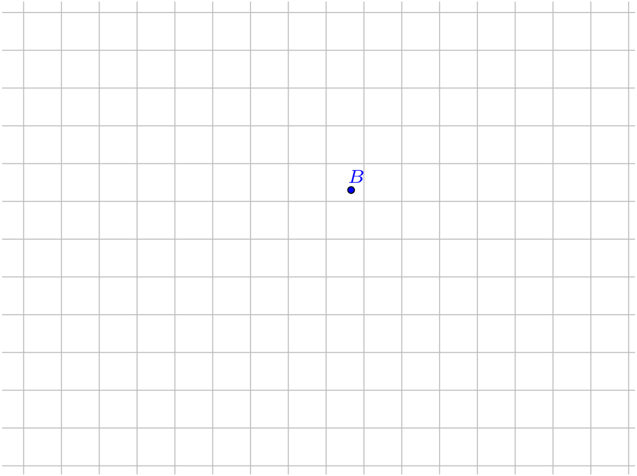


Figure 16: Constructing the pole of a line

Constructing the pole of a line

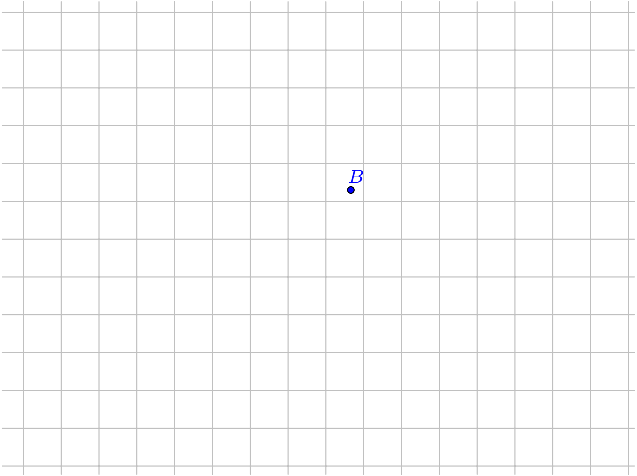


Figure 16: Constructing the pole of a line

Constructing the pole of a line

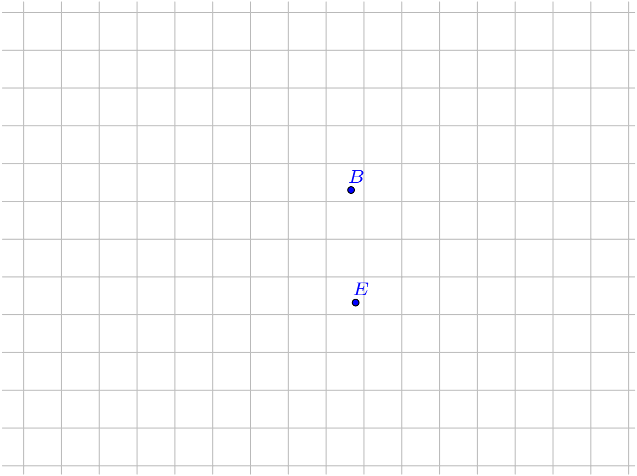


Figure 16: Constructing the pole of a line

Constructing the pole of a line

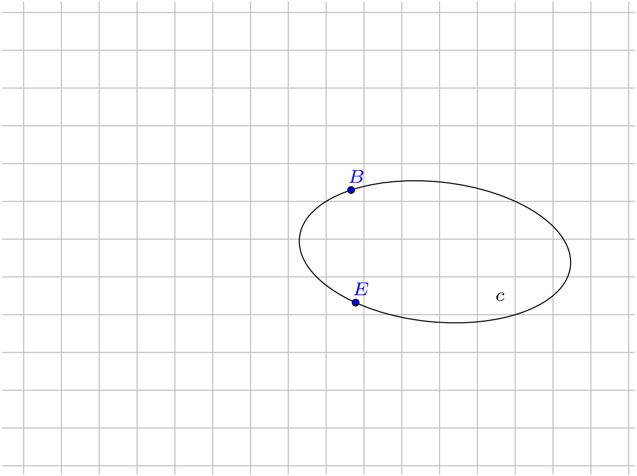


Figure 16: Constructing the pole of a line

Constructing the pole of a line

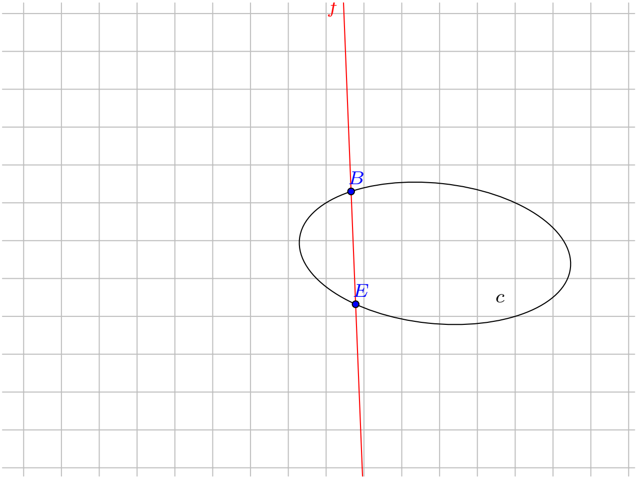


Figure 16: Constructing the pole of a line

Constructing the pole of a line

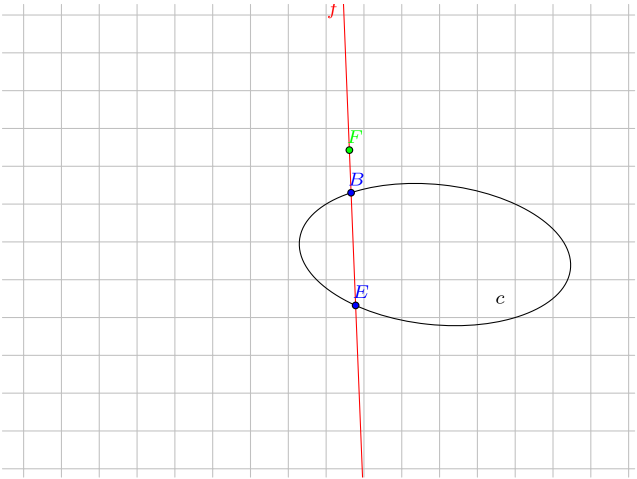


Figure 16: Constructing the pole of a line

Constructing the pole of a line

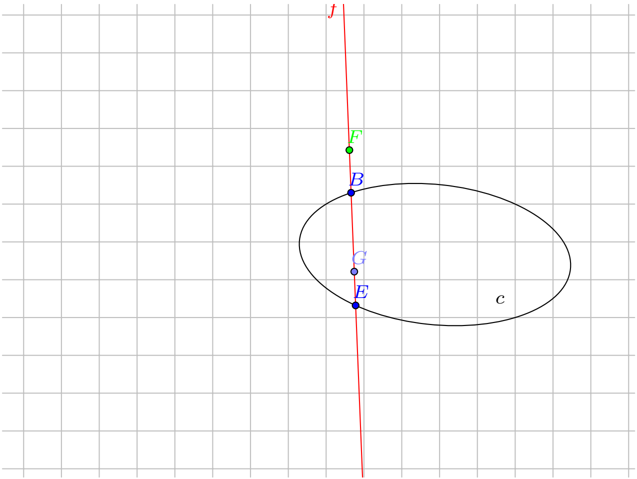


Figure 16: Constructing the pole of a line

Constructing the pole of a line

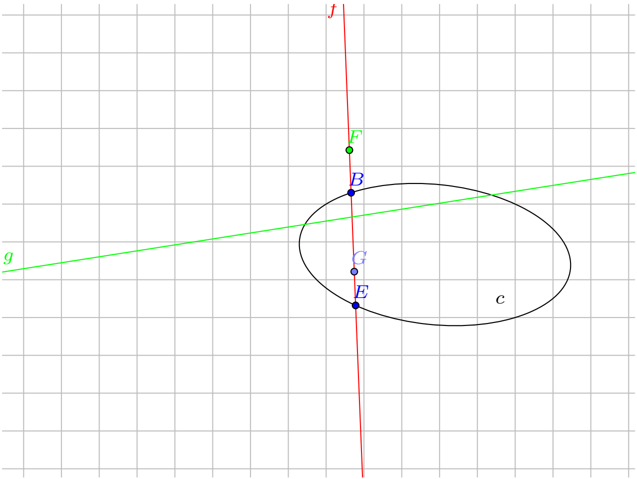


Figure 16: Constructing the pole of a line

Constructing the pole of a line

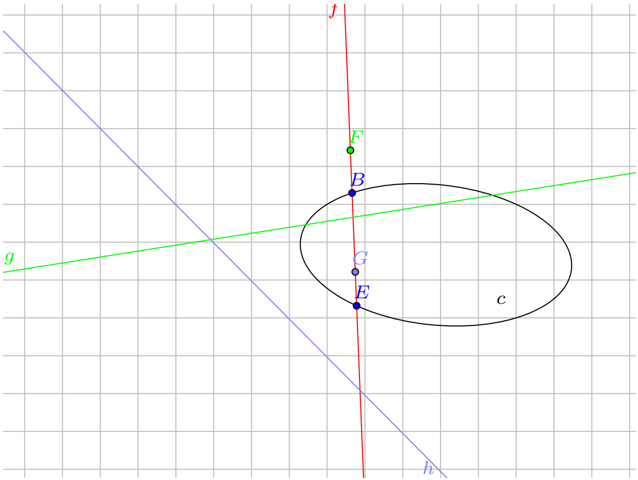


Figure 16: Constructing the pole of a line

Constructing the pole of a line

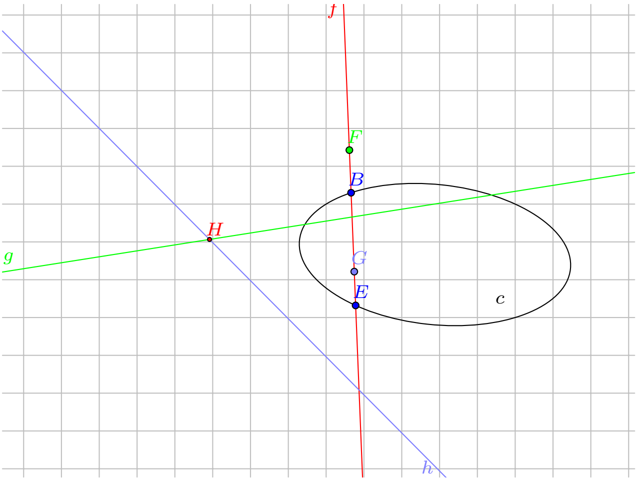


Figure 16: Constructing the pole of a line

Construct the tangent of a point on a conic

- To construct the tangent at a given point P on a conic, join P to the pole of any secant through P.

Example of construct the tangent of a point on a conic

Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

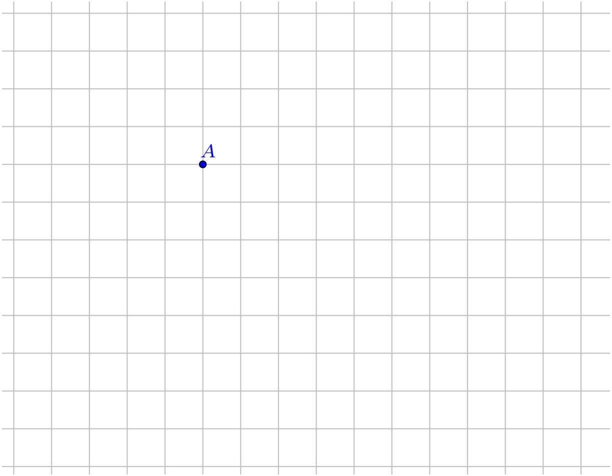


Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

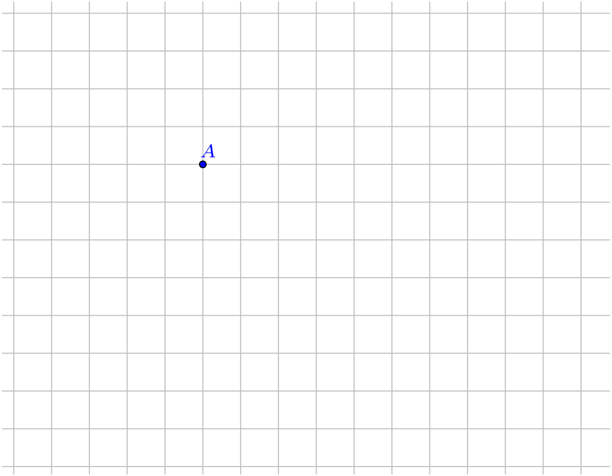


Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

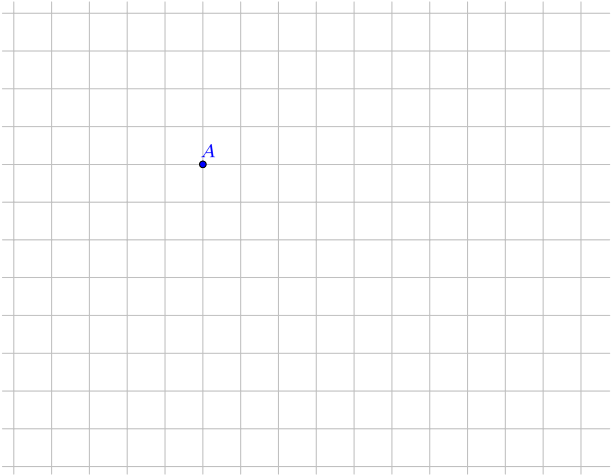


Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

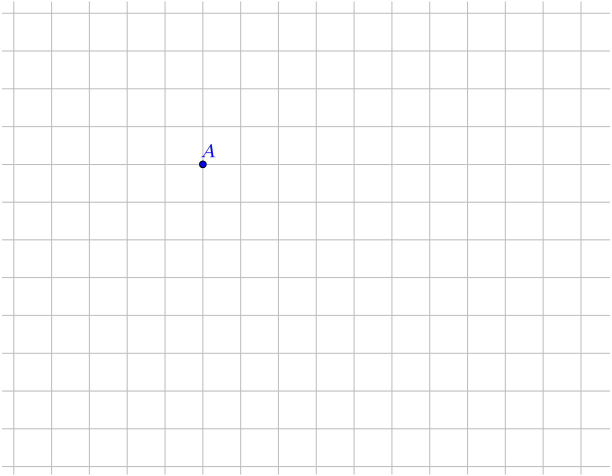


Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

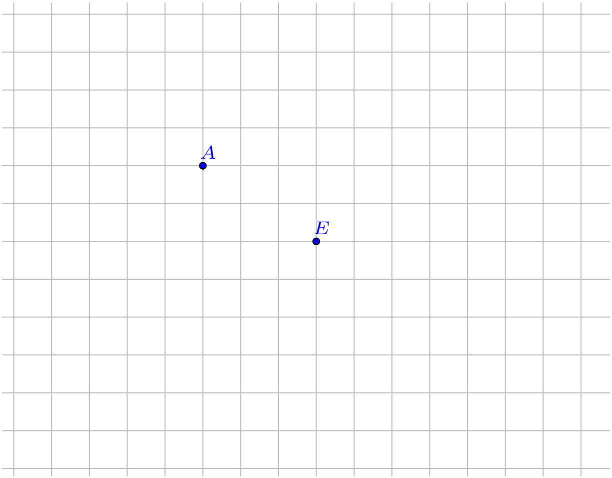


Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

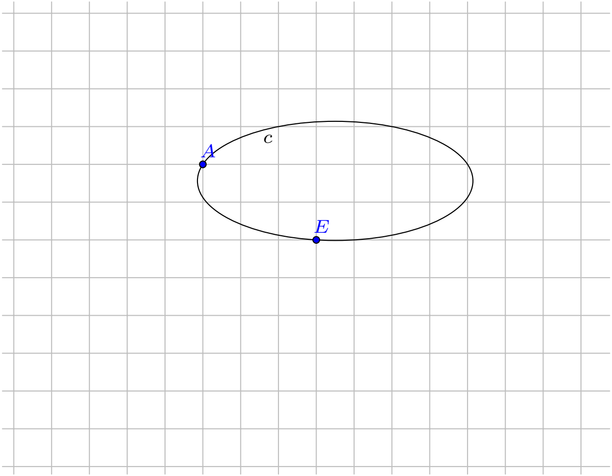


Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

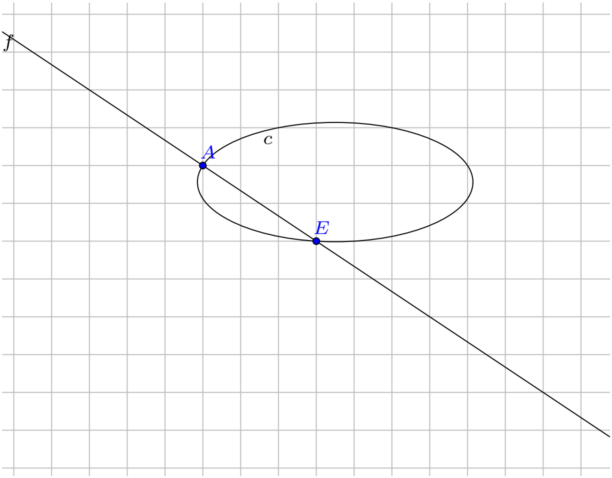


Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

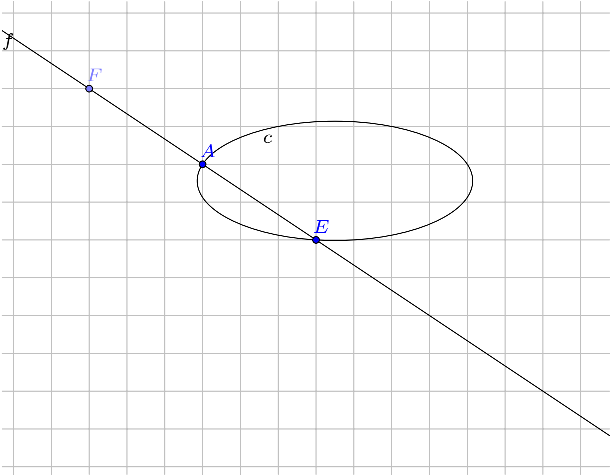


Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

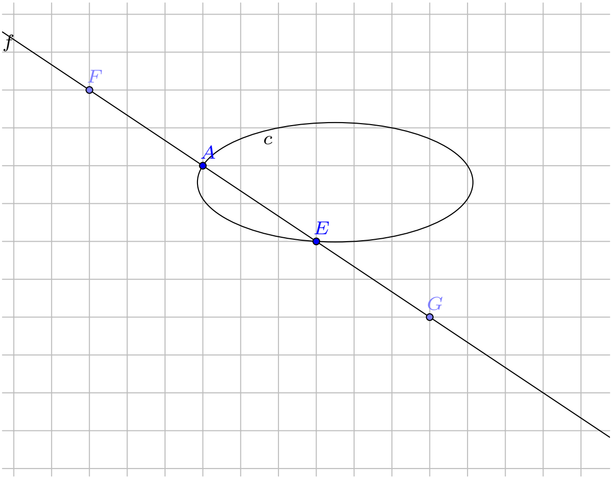


Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

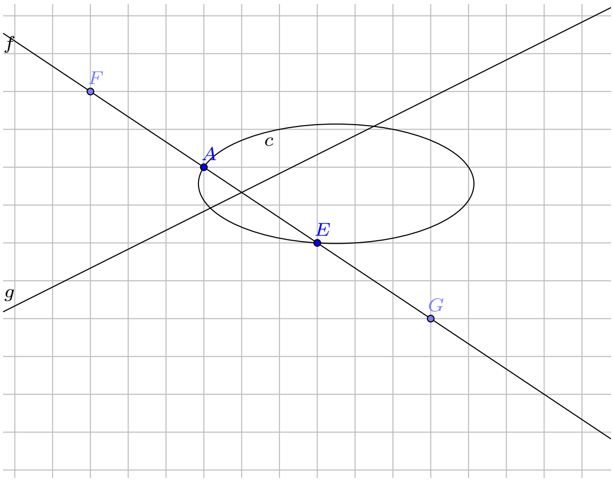


Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

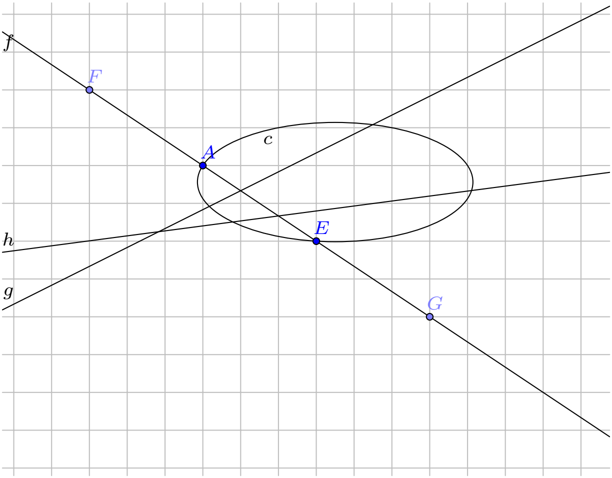


Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

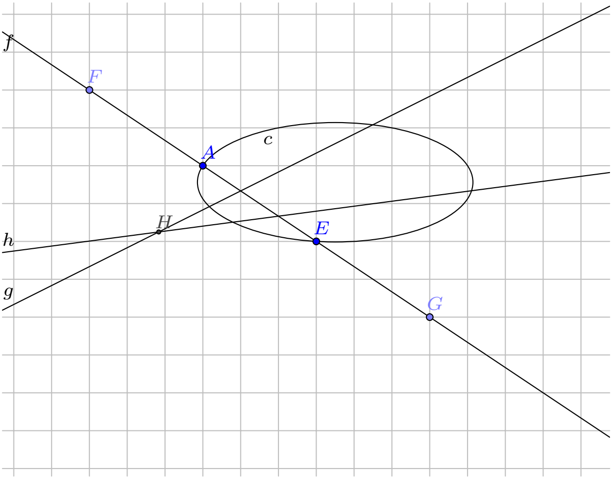


Figure 17: Construct the tangent of a point on a conic

Example of construct the tangent of a point on a conic

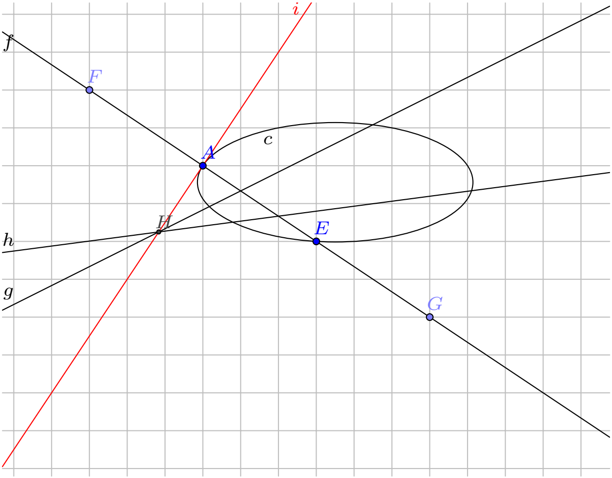


Figure 17: Construct the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

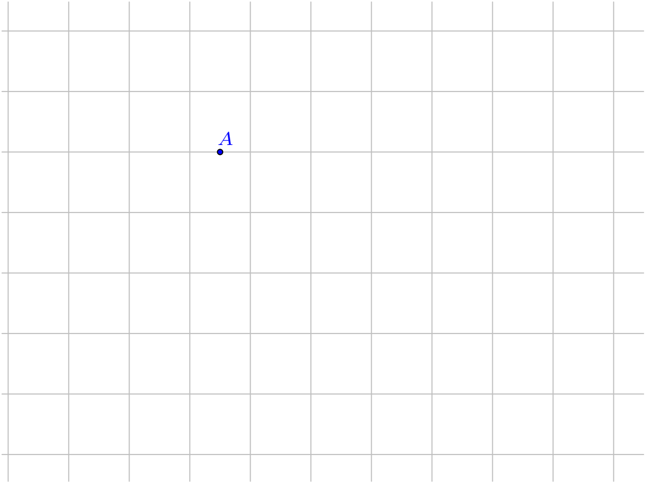


Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

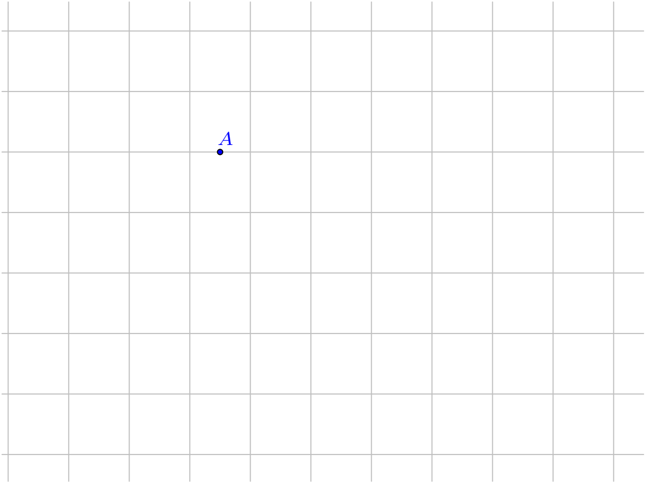


Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

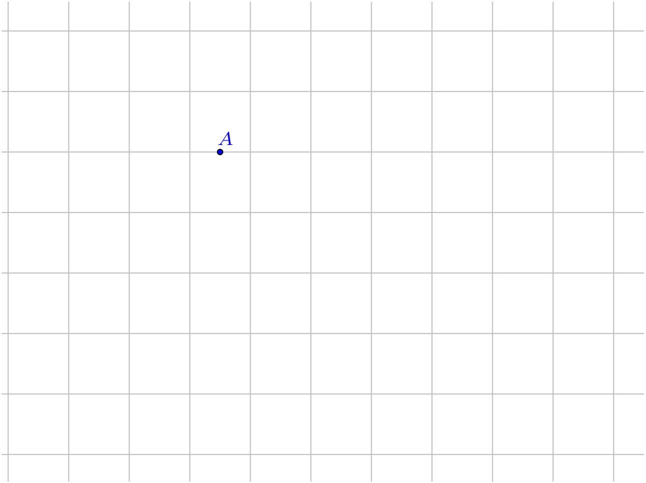


Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

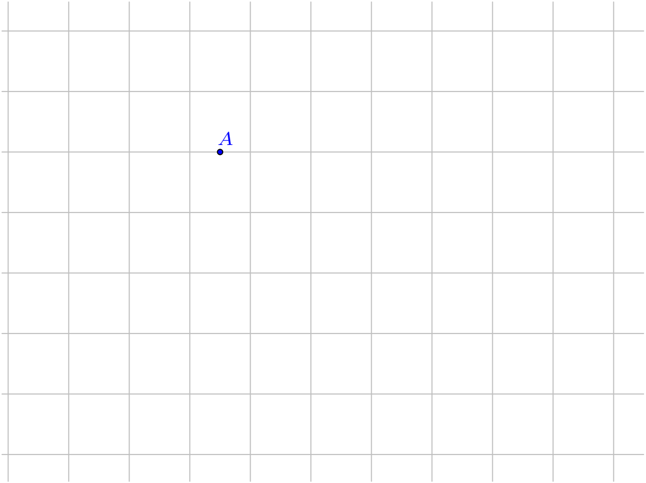


Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

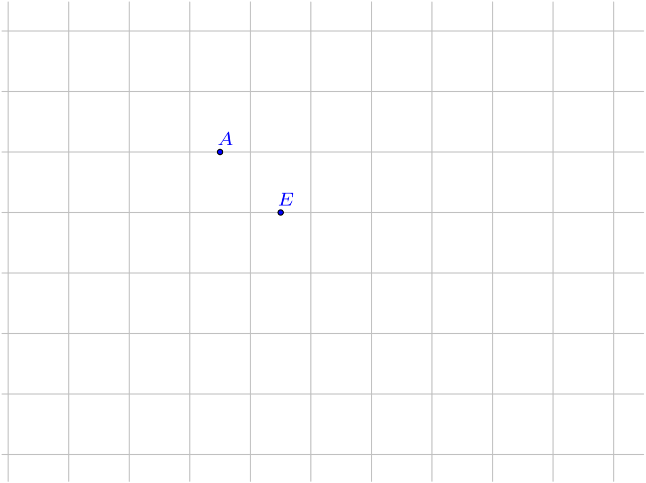


Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

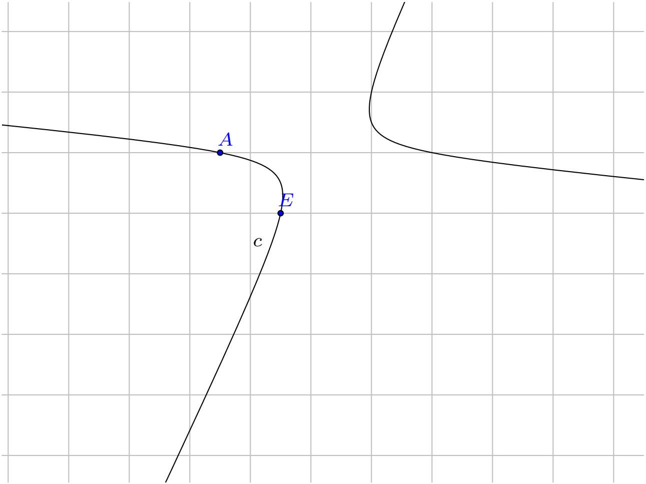


Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

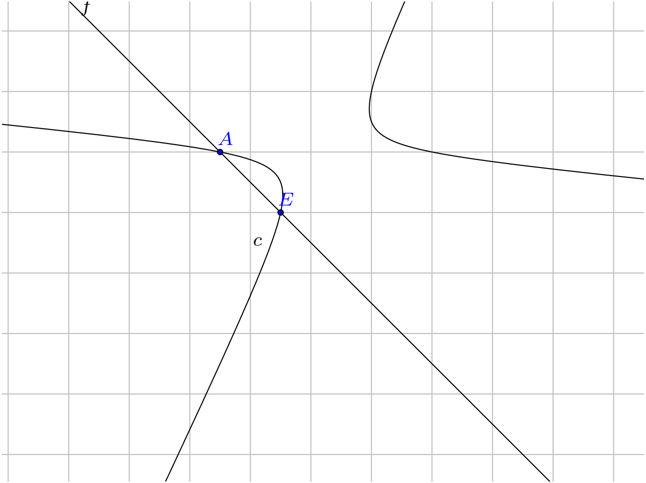


Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

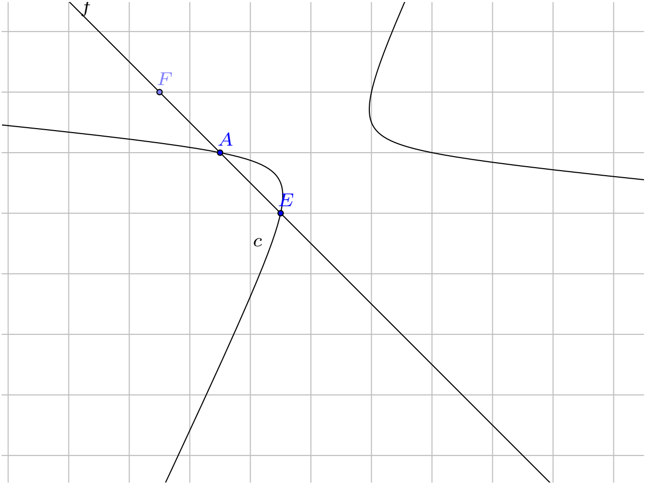


Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

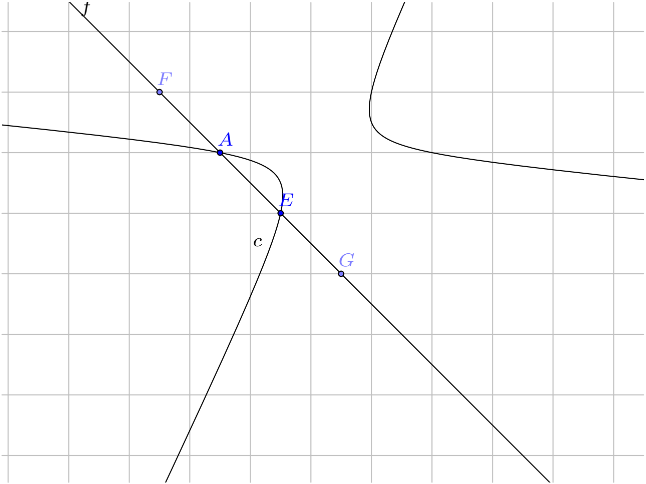


Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

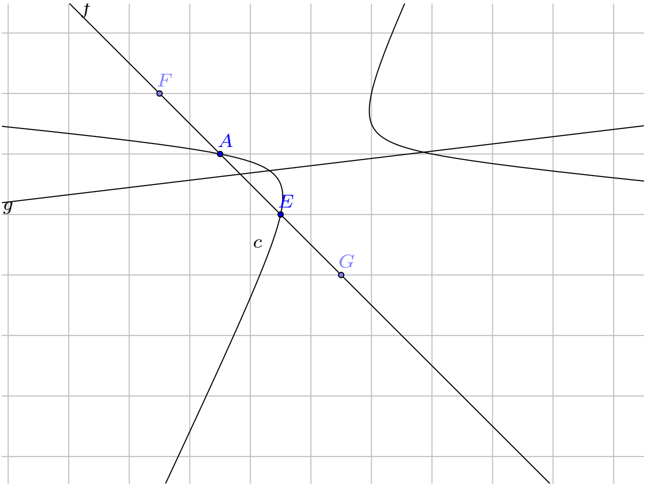


Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

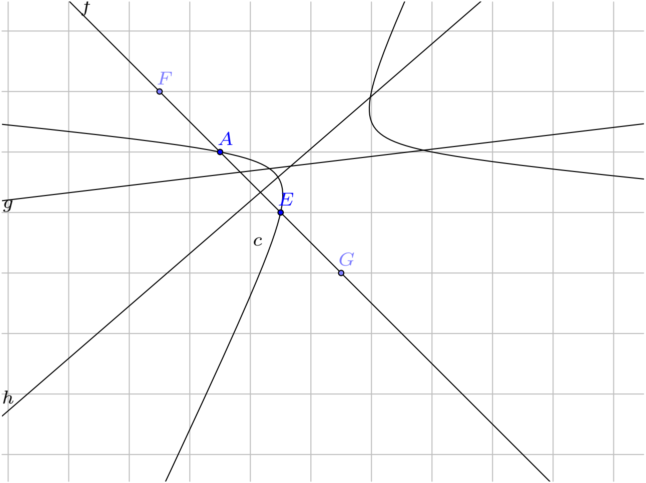


Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

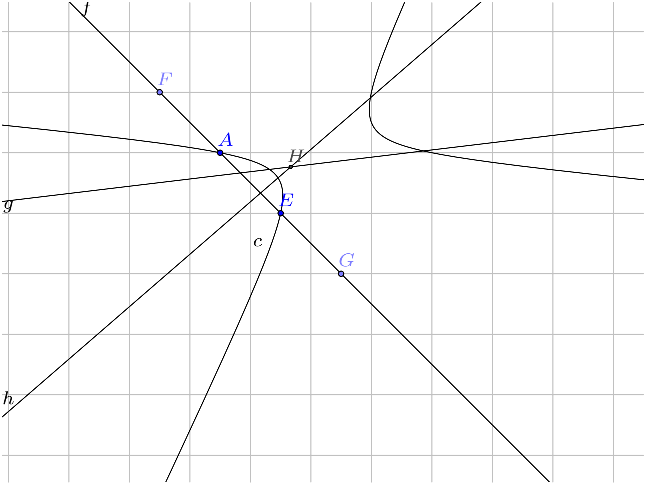


Figure 18: Another example of constructing the tangent of a point on a conic

Another example of constructing the tangent of a point on a conic

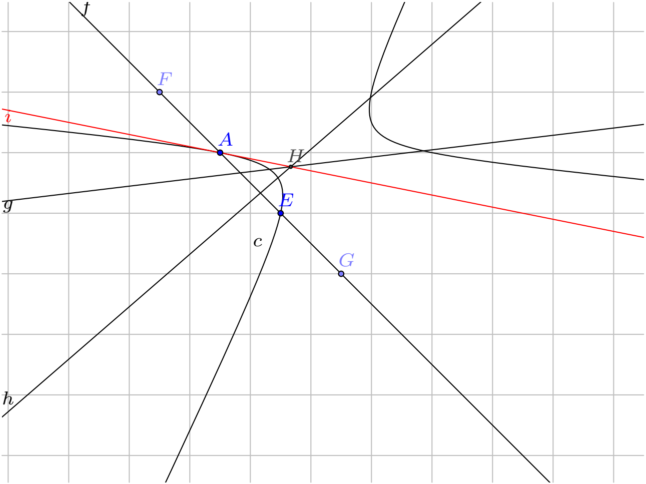


Figure 18: Another example of constructing the tangent of a point on a conic

Pascal's Theorem

- If a hexagon is inscribed in a conic, the three pairs of opposite sides meet in collinear points.

An instance of Pascal’ theorem

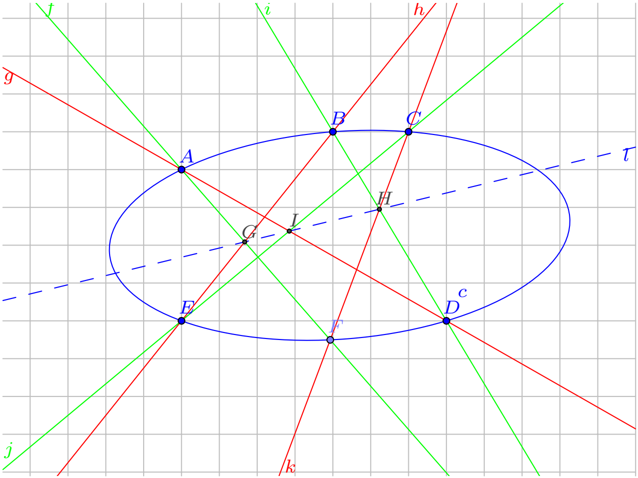


Figure 19: pascal

Another instance of Pascal' theorem

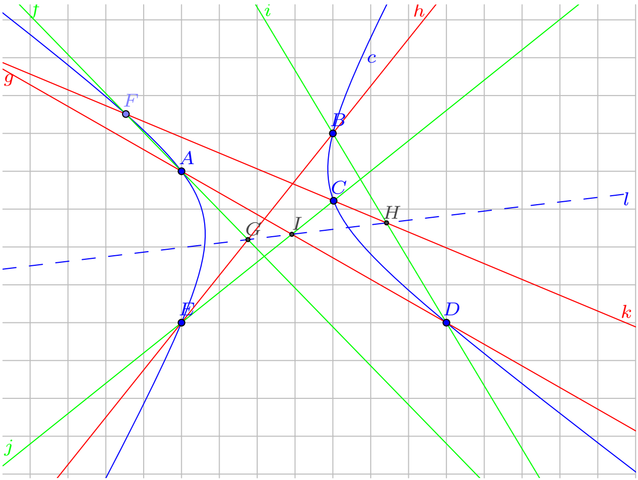


Figure 20: pascal2

Backup

melpon.org

```
> pandoc -s --mathjax -t revealjs -V theme=default -o proj_geom
> pandoc -t beamer -o proj_geom.pdf proj_geom.md beamer.yaml
> pandoc -o proj_geom.docx proj_geom.md
```