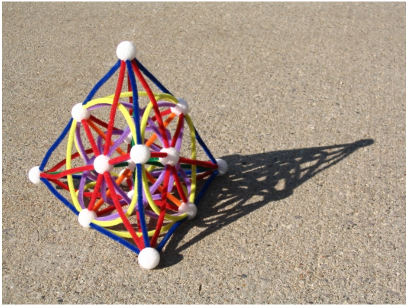
Proof Pearl: Formalizing Spreads and Packings of the Smallest Projective Space PG(3,2) Using the Coq Proof Assistant



Picture taken from David A. Richter
http://homepages.wmich.edu/~drichter/projectivespace.htm

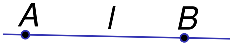
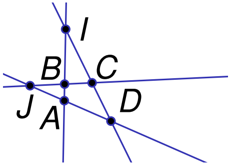
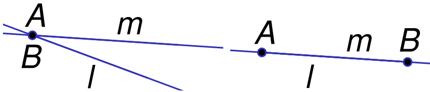
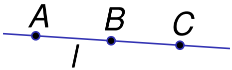
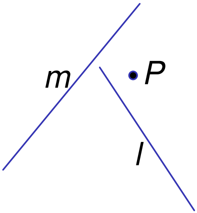
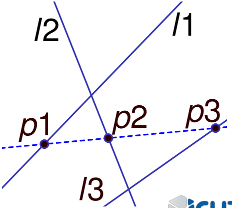
Nicolas Magaud
ICube, UMR 7357 CNRS - Université de Strasbourg, France

ITP 2022 / FLoC 2022 - Haifa - Israel - August 9th 2022

What is Projective (Space) Geometry ?

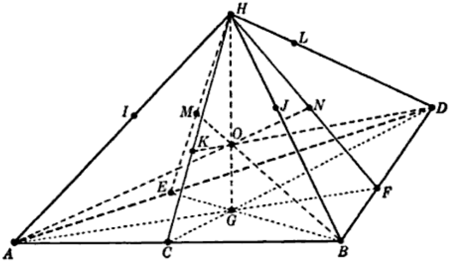
- Context
  - Incidence Geometry
    - only points, lines and an incidence relation
  - Projective Incidence Geometry
    - in 2D : 2 lines always intersect
    - in 3D : Pasch's axiom
  - Simple description : only 6 axioms for 3D
- Outline of this talk
  - Specifying the smallest projective space PG(3,2)
  - Proving that it verifies the axioms of projective geometry
  - Computing its Spreads and Packings and Prove their Properties

Axioms for Projective Space Geometry

A1P3	A2P3	A3P3
		
A4P3	**A5P3**	**A6P3**
		

The Smallest Projective Space PG(3,2)

- PG(3,2) :
  - 15 points
  - 35 lines
  - 15 planes



- Each line is contained in 3 planes and contains 3 points.
- Each point is contained in 7 lines and 7 planes.
- Each plane contains 7 points and 7 lines.
- Every pair of distinct planes intersect in a line.
- A line and a plane not containing the line intersect in exactly one point.

All 35 lines of PG(3,2) and their points

L0 : 0 1 2	**L7 :** 1 4 6	**L14 :** 2 11 14	**L21 :** 3 9 13	**L28 :** 5 7 13
L1 : 0 3 4	**L8 :** 1 8 10	**L15 :** 2 3 6	**L22 :** 3 7 11	**L29 :** 5 9 11
L2 : 0 5 6	**L9 :** 1 12 14	**L16 :** 2 12 13	**L23 :** 4 9 14	**L30 :** 5 10 12
L3 : 0 7 8	**L10 :** 1 7 9	**L17 :** 2 4 5	**L24 :** 4 8 11	**L31 :** 6 7 14
L4 : 0 9 10	**L11 :** 1 13 11	**L18 :** 2 8 9	**L25 :** 4 10 13	**L32 :** 6 8 13
L5 : 0 11 12	**L12 :** 1 3 5	**L19 :** 3 10 14	**L26 :** 4 7 12	**L33 :** 6 9 12
L6 : 0 13 14	**L13 :** 2 7 10	**L20 :** 3 8 12	**L27 :** 5 8 14	**L34 :** 6 10 11

PG(3,2) features 15 points and 35 lines (with 3 points each).

Each line has exactly 3 points.
Each point exactly belongs to 7 lines.

A spread of PG(3,2) : #17

L0 : 0 1 2	**L7 :** 1 4 6	**L14 :** 2 11 14	**L21 :** 3 9 13	**L28 :** 5 7 13
L1 : 0 3 4	**L8 :** 1 8 10	**L15 :** 2 3 6	**L22 :** 3 7 11	**L29 :** 5 9 11
L2 : 0 5 6	**L9 :** 1 12 14	**L16 :** 2 12 13	**L23 :** 4 9 14	**L30 :** 5 10 12
L3 : 0 7 8	**L10 :** 1 7 9	**L17 :** 2 4 5	**L24 :** 4 8 11	**L31 :** 6 7 14
L4 : 0 9 10	**L11 :** 1 13 11	**L18 :** 2 8 9	**L25 :** 4 10 13	**L32 :** 6 8 13
L5 : 0 11 12	**L12 :** 1 3 5	**L19 :** 3 10 14	**L26 :** 4 7 12	**L33 :** 6 9 12
L6 : 0 13 14	**L13 :** 2 7 10	**L20 :** 3 8 12	**L27 :** 5 8 14	**L34 :** 6 10 11

A spread is a set of 5 lines partitioning the set of all (15) points.

0 (L2) 1 (L8) 2 (L16) 3 (L22) 4 (L23) 5 (L2) 6 (L2) 7 (L22) 8 (L8)
9 (L23) 10 (L8) 11 (L22) 12 (L16) 13 (L16) 14 (L23)

All 56 spreads of PG(3,2)

S0 0 19 24 28 33
S1 0 19 26 29 32
S2 0 20 23 28 34
S3 0 20 25 29 31
S4 0 21 24 30 31
S5 0 21 26 27 34
S6 0 22 23 30 32
S7 0 22 25 27 33
S8 1 8 14 28 33
S9 1 8 16 29 31
S10 1 9 13 29 32
S11 1 9 18 28 34
S12 1 10 14 30 32
S13 1 10 16 27 34
S14 1 11 13 27 33
S15 1 11 18 30 31
S16 2 8 14 21 26
S17 2 8 16 22 23
S18 2 9 13 21 24
S19 2 9 18 22 25
S20 2 10 14 20 25
S21 2 10 16 19 24
S22 2 11 13 20 23
S23 2 11 18 19 26
S24 3 7 14 21 30
S25 3 7 16 19 29
S26 3 9 15 25 29
S27 3 9 17 21 34
S28 3 11 15 23 30
S29 3 11 17 19 33
S30 3 12 14 25 33
S31 3 12 16 23 34
S32 4 7 14 20 28
S33 4 7 16 22 27
S34 4 9 15 24 28
S35 4 9 17 22 32
S36 4 11 15 26 27
S37 4 11 17 20 31
S38 4 12 14 26 32
S39 4 12 16 24 31
S40 5 7 13 21 27
S41 5 7 18 19 28
S42 5 8 15 23 28
S43 5 8 17 21 31
S44 5 10 15 25 27
S45 5 10 17 19 32
S46 5 12 13 23 32
S47 5 12 18 25 31
S48 6 7 13 20 29
S49 6 7 18 22 30
S50 6 8 15 26 29
S51 6 8 17 22 33
S52 6 10 15 24 30
S53 6 10 17 20 34
S54 6 12 13 24 33
S55 6 12 18 26 34

A packing of PG(3,2) : #42

S0 0 19 24 28 33	S14 1 11 13 27 33	S28 3 11 15 23 30	S42 5 8 15 23 28
S1 0 19 26 29 32	S15 1 11 18 30 31	S29 3 11 17 19 33	S43 5 8 17 21 31
S2 0 20 23 28 34	S16 2 8 14 21 26	S30 3 12 14 25 33	S44 5 10 15 25 27
S3 0 20 25 29 31	S17 2 8 16 22 23	S31 3 12 16 23 34	S45 5 10 17 19 32
S4 0 21 24 30 31	S18 2 9 13 21 24	S32 4 7 14 20 28	S46 5 12 13 23 32
S5 0 21 26 27 34	S19 2 9 18 22 25	S33 4 7 16 22 27	S47 5 12 18 25 31
S6 0 22 23 30 32	S20 2 10 14 20 25	S34 4 9 15 24 28	S48 6 7 13 20 29
S7 0 22 25 27 33	S21 2 10 16 19 24	S35 4 9 17 22 32	S49 6 7 18 22 30
S8 1 8 14 28 33	S22 2 11 13 20 23	S36 4 11 15 26 27	S50 6 8 15 26 29
S9 1 8 16 29 31	S23 2 11 18 19 26	S37 4 11 17 20 31	S51 6 8 17 22 33
S10 1 9 13 29 32	S24 3 7 14 21 30	S38 4 12 14 26 32	S52 6 10 15 24 30
S11 1 9 18 28 34	S25 3 7 16 19 29	S39 4 12 16 24 31	S53 6 10 17 20 34
S12 1 10 14 30 32	S26 3 9 15 25 29	S40 5 7 13 21 27	S54 6 12 13 24 33
S13 1 10 16 27 34	S27 3 9 17 21 34	S41 5 7 18 19 28	S55 6 12 18 26 34

Focus on the packing #42

packing #42 = {

S1 :	0 19 26 29 32
S13 :	1 10 16 27 34
S18 :	2 9 13 21 24
S28 :	3 11 15 23 30
S32 :	4 7 14 20 28
S47 :	5 12 18 25 31
S51 :	6 8 17 22 33

Is this actually a partition of the set of lines of PG(3,2) ?

0 (S1) 1 (S13) 2 (S18) 3 (S28) 4 (S32) 5 (S47) 6 (S51) 7 (S32)
8 (S51) 9 (S18) 10 (S13) 11 (S28) 12 (S47) 13 (S18) 14 (S32)
15 (S28) 16 (S13) 17 (S51) 18 (S47) 19 (S1) 20 (S32) 21 (S18)
22 (S51) 23 (S28) 24 (S18) 25 (S47) 26 (S1) 27 (S13) 28 (S32)
29 (S1) 30 (S28) 31 (S47) 32 (S1) 33 (S51) 34 (S13)

Coq Specifications

- Point and Line are implemented as simple inductive types.
  - Case analysis is easy and fast.
  - Writing the specification is a bit boring.

    ```
    Inductive Point := P0 | P1 | P2 | ... | P14.
    ```
- Automation : we use an external program which
  - generates the specification (points, lines, incidence relation)
  - computes all the spreads, all the packings and the collineations relating them
  - generates the witnesses for existential proofs
  - generates all the lemmas and their proofs
- Implementation choices
  - incidence relation as a boolean predicate
  - decidable equality
  - *ad-hoc* order relation on points, lines, etc.
  - witnesses (for existentials) are computed in advance

Implementation in Coq

```
Inductive Point :=
| P0 | P1 | P2 | P3 | P4 | P5 | P6 | P7 | P8 | P9 | P10 | P11 | P12 | P13 | P14 .

Inductive Line :=
| L0 | L1 | L2 | L3 | L4 | L5 | L6 | L7 | L8 | L9
| L10 | L11 | L12 | L13 | L14 | L15 | L16 | L17 | L18 | L19
| L20 | L21 | L22 | L23 | L24 | L25 | L26 | L27 | L28 | L29
| L30 | L31 | L32 | L33 | L34 .

Definition incid_lp (p:Point) (l:Line) : bool :=
match l with
| L0 => match p with P0 | P1 | P2 => true | _ => false end
| L1 => match p with P0 | P3 | P4 => true | _ => false end
| L2 => match p with P0 | P5 | P6 => true | _ => false end
| L3 => match p with P0 | P7 | P8 => true | _ => false end
| L4 => match p with P0 | P10 | P9 => true | _ => false end
| [...]
end.

Definition f_a3_3 (l1:Line) (l2:Line) (l3:Line) :=
match l3 with
 | L0 => match l2 with
        | L0 => match l1 with
                  | L0 => (L0,(P0,P0,P0))
                 | _ => (L0,(P0,P0,P0))
                end
       | _ => (L0, (P0,P0,P0))
        end
 | L1 => [...]
end.
```

Spreads and Packings of PG(3,q)

- A spread of PG(3,q) is a set of $q^2 + 1$ lines which are pairwise disjoint and thus partitions the set of points.
  - In PG(3,2), it corresponds to some sets of 5 lines.
- A packing of PG(3,q) is a set of $q^2 + q + 1$ spreads which are pairwise disjoint and thus partitions the set of lines.
  - In PG(3,2), it corresponds to some sets of 7 spreads.
- In PG(3,2)
  - There are 56 (isomorphic) spreads in PG(3,2).
  - There are 240 packings in PG(3,2), divided into 2 distinct equivalence classes (120 packings each). [1]

1. Finite Projective Spaces of Three Dimensions (Hirschfeld,1985)

Spreads

- The 56 spreads are computed externally into a list spreads.
- Formal definition of a spread in Coq

```
Definition is_spread5 (l1 l2 l3 l4 l5:Line) : bool :=
  disj_5l l1 l2 l3 l4 l5  && is_partition5 l1 l2 l3 l4 l5.
```

- This list exactly contains all the spreads of PG(3,2).

```
forall l1 l2 l3 l4 l5, leL l1 l2 && leL l2 l3 && leL l3 l4 && leL l4 l5 ->
 (is_spread5 l1 l2 l3 l4 l5) <-> In [l1;l2;l3;l4;l5] spreads.
```

  - Proof by induction on the 5 variables l_1, l_2, l_3, l_4, l_5
  - 35^5 = 52 521 875 cases
- All these 56 spreads are isomorphic.
  - One can switch from one to another using a collineation i.e. an automorphism of PG(3,2) which respects incidence.
  - Proof achieved using a circular argument $S_0 \to S_1 \to S_2 \to \ldots \to S_{55} \to S_0$

Packings

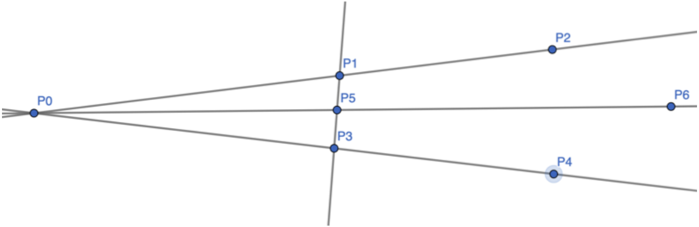
- We build the 240 packings of PG(3,2).
- We show that they are no other packings of PG(3,2).

```
Lemma is_packing_descr : forall s1 s2 s3 s4 s5 s6 s7 : list Line,
 ltS s1 s2 && ... && ltS s6 s7 ->
 In s1 spreads -> ... -> In s7 spreads ->
  (is_packing7 s1 s2 s3 s4 s5 s6 s7) <-> In [s1;s2;s3;s4;s5;s6;s7] packings.
```

- We build two classes of isomorphim (120 packings each).
- Last step : show that they are exactly two **distinct** classes.

All Collineations

- There are 20 160 collineations in PG(3,2).
- We build them all.
- The images of `P0`, `P1`, `P3` and `P7` define a collination.



- Consider two packings (expected to belong to two different classes) and try each collineation to relate them.
- Conclusion : there are exactly two distinct classes of packings.

Proof Engineering

- Issues
  - Requires efficient representations of objects for **both** computation and formal (automated) reasoning
  - Dealing with a large development in Coq : 50+ files, 317 345 lines of code, incl. 290 000 lines of proofs, 13 hours to check it using task parallelism.
- Solutions
  - Small-scale reflection : using `bool` instead of `Prop`
  - Optimizing the proofs
    - Efficient tactics : `reflexivity` vs `apply erefl`
    - Solving goals at first encounter.
    - Without loss of generality (`wlog` tactic)
  - Circumventing the limitations
    - Splitting Proof statements (e.g. into 15 sub-statements)
    - Avoiding large files (even for automatically generated ones)

Conclusions and Future Work

- Achievements
  - Some (Big) Formal Proofs in PG(3,2)
  - Pushing Coq to its limits
- Next steps (examples of state-of-the-art results)
  - Anton Betten. *The packings of PG(3,3)* . 2015
    8 424 distinct spreads instead of 56,
    73 343 classes of packings instead of 2,
    12 130 560 collineations instead of 20 160.
  - Svetlana Topalova and Stela Zhelezova.
    On transitive parallelisms of PG(3,4). 2017

Thanks ! Questions ?

`https://github.com/magaud/PG3q`