# Stage Master 2 Informatique ou fin d'études ingénieur : Classification des packings de l'espace projectif fini PG(3,3)

Nicolas Magaud (magaud@unistra.fr)

## Sujet du stage

La géométrie projective est une approche de la géométrie permettant de capturer les notions de *perspective* et d'*horizon*. En 2D, cela revient à faire l'hypothèse que deux droites quelconques sont toujours concourantes. En 3D, cela revient à dire que deux droites coplanaires se coupent toujours.

La géométrie projective peut être modélisée par un système d'axiomes très simple. Dans ce cas, on peut facilement prouver que certains espaces finis (contenant un nombre fini de points et de droites) vérifient bien les axiomes de la géométrie projective. Dans ce travail, nous nous intéressons aux *petits* espaces projectifs finis de la forme PG(3,q). Nous avons déjà décrit formellement dans l'assistant de preuves Coq [3] les espaces PG(3,2) et PG(3,3) [2]. Nous avons également démontré formellement que les 56 spreads de PG(3,2) sont tous isomorphes alors que les 240 packings de PG(3,2) peuvent être classifiés en deux classes distincts de 120 éléments [4].

| | # points | # lines | # points per line | # lines per spread | # spreads per packing |
|---|---|---|---|---|---|
| $PG(3,2)$ | 15 | 35 | 3 | 5 | 7 |
| $PG(3,3)$ | 40 | 130 | 4 | 10 | 13 |
| $PG(3,4)$ | 85 | 357 | 5 | 17 | 21 |
| $PG(3,q)$ | $(q^2+1)(q+1)$ | $(q^2+q+1)(q^2+1)$ | $q+1$ | $q^2+1$ | $q^2+q+1$ |

Table 1 – Nombres de points, droites et de points par droite pour les espaces projectifs finis PG(3,q).

Dans ce projet, nous considérons l'espace projectif fini PG(3,3), qui contient 40 points et 130 droites. Le but est de calculer et de classifier les spreads et les packings de PG(3,3), en suivant l'approche proposée par Betten [1]. Un *spread* est une partition des points en droites disjointes. Dans PG(3,3), un spread est constitué de 10 droites. Il existe deux familles de spreads non-isomorphes : les spreads Désarguesiens (2 106) et les spreads de Hall (6 318). Un packing de PG(3,3) est une partition des 130 droites en 13 spreads disjoints contenant chacun 10 droites. A isomorphisme près, il a 73 343 packings dans PG(3,3).

Afin de pouvoir classifier les spreads et les packings de PG(3,3), il faut à tout prix éviter de devoir énumérer exhaustivement tous les objets mis en jeu. Pour cela, on étudiera comment les algorithmes de McKay [6] ou Faradžev-Read [7, 5] peuvent s'appliquer et quel impact cela peut avoir sur le travail de formalisation.

A terme, l'objectif est de disposer de tous les spreads et tous les packings de PG(3,3) ainsi que leurs classifications afin de pouvoir prouver formellement en Coq leurs propriétés.

## Informations pratiques

**Lieu du stage :** Laboratoire ICube, sur le campus d'Illkirch de l'Université de Strasbourg

**Equipe de recherche :** équipe IGG `https://igg.icube.unistra.fr/index.php/Accueil`

**Encadrant :** Nicolas Magaud `https://dpt-info.u-strasbg.fr/~magaud/`

Ce stage donne droit à la gratification de stage standard dans un laboratoire de recherche français. Les candidat(e)s intéressé(e)s peuvent me contacter par mail (magaud@unistra.fr) pour plus d'informations. Une poursuite en thèse est envisageable via un contrat de l'école doctorale MSII.

# Références

[1] Anton Betten. The packings of pg(3, 3). *Designs, Codes and Cryptography*, 79(3) :583–595, 2016.

[2] David Braun, Nicolas Magaud, and Pascal Schreck. Formalizing Some "Small" Finite Models of Projective Geometry in Coq. In Jacques Fleuriot, Dongming Wang, and Jacques Calmet, editors, *Proceedings of Artificial Intelligence and Symbolic Computation 2018 (AISC'2018)*, number 11110 in LNAI, pages 54–69, Sept. 2018. https ://hal.archives-ouvertes.fr/hal-01835493.

[3] Coq development team. *The Coq Proof Assistant Reference Manual, Version 8.14.0*, 2021.

[4] Nicolas Magaud. Proof Pearl : Formalizing Spreads and Packings of the Smallest Projective Space PG(3, 2) Using the Coq Proof Assistant. In June Andronick and Leonardo de Moura, editors, *13th International Conference on Interactive Theorem Proving, ITP 2022, August 7-10, 2022, Haifa, Israel*, volume 237 of *LIPIcs*, pages 25 :1–25 :17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022.

[5] Filip Maric. Verifying faradžev-read type isomorph-free exhaustive generation. In Nicolas Peltier and Viorica Sofronie-Stokkermans, editors, *Automated Reasoning - 10th International Joint Conference, IJCAR 2020, Paris, France, July 1-4, 2020, Proceedings, Part II*, volume 12167 of *Lecture Notes in Computer Science*, pages 270–287. Springer, 2020.

[6] Brendan D. McKay. Isomorph-free exhaustive generation. *J. Algorithms*, 26(2) :306–324, 1998.

[7] Ronald C. Read. Every one a winner or how to avoid isomorphism search when cataloguing combinatorial configurations*. In B. Alspach, P. Hell, and D.J. Miller, editors, *Algorithmic Aspects of Combinatorics*, volume 2 of *Annals of Discrete Mathematics*, pages 107–120. Elsevier, 1978.