Classification of the Hyperovals in $\mathrm{PG}(2, 64)$

Peter Vandendriessche[*]

Department of Mathematics: (WE16)
Krijgslaan 281, gebouw S8
9000 Gent, Belgium

peter.vandendriessche@ugent.be

Submitted: Jan 4, 2018; Accepted: Mar 10, 2019; Published: May 31, 2019

© The author.

Abstract

In this paper, we present a full classification of the hyperovals in the finite projective plane $\mathrm{PG}(2, 64)$, showing that there are exactly 4 isomorphism classes. The techniques developed to obtain this result can be applied more generally to classify point sets with 0 or 2 points on every line, in a broad range of highly symmetric incidence structures.

Mathematics Subject Classifications: 51E22

1 Background

Definition 1. The *projective plane* over a field $\mathbb{F}$ is the pair $(\mathcal{P}, \mathcal{L})$ where $\mathcal{P}$ is the set of 1-dimensional subspaces of $\mathbb{F}^3$ (called the *points* of the plane) and $\mathcal{L}$ is the set of 2-dimensional subspaces of $\mathbb{F}^3$ (called the *lines* of the plane).

This structure has the property that every two points are contained in exactly one line, and every two lines intersect in exactly one common point.

The projective plane over a finite field $\mathbb{F}_q$ is also called the *Desarguesian projective plane of order* q, denoted by $\mathrm{PG}(2, q)$. It is one of the most studied structures in finite combinatorics. The $\frac{q^3-1}{q-1}$ points of the plane are typically represented by the set

$$\{(0, 0, 1)\} \cup \{(0, 1, t) | t \in \mathbb{F}_q\} \cup \{(1, x, y) | x, y \in \mathbb{F}_q\},$$

where the vector (a, b, c) is used to represent the 1-dimensional subspace $\langle(a, b, c)\rangle$.

Definition 2. An *arc* in a projective plane Π is a set of points, no three of which lie on a line. A k*-arc* is an arc of size k.

[*] The research of the author is supported by a postdoctoral grant of the FWO-Flanders.

For any k-arc with $k \geqslant 2$, we can choose the coordinates in such a way that the first two points are $(0,0,1)$ and $(0,1,0)$. Then, all remaining points are of the form $(1,x,y)$ where no value of x appears twice. This implies that $k \leqslant q+2$.

When q is odd, an additional counting argument shows that one even has $k \leqslant q+1$. One well-known example of a large arc is a *conic*, i.e. a point set isomorphic to $\{(1,t,t^2)|t \in \mathbb{F}_q\} \cup \{(0,0,1)\}$.

For q odd, Segre [20] showed that every $(q+1)$-arc is a conic. For q even, this is not necessarily true, but it can be shown that every $(q+1)$-arc is a subset of a $(q+2)$-arc, which we call a *hyperoval.* Hence, it is of interest to study these hyperovals. Using the coordinates above, we can express every hyperoval as $\{(0,0,1),(0,1,0)\} \cup \{(1,x,f(x))|x \in \mathbb{F}_q\}$ for some permutation $f : \mathbb{F}_q \to \mathbb{F}_q$. This f is called the *o-polynomial* of the hyperoval.

Definition 3. A *hyperoval* is a $(q+2)$-arc in $\mathrm{PG}(2,q)$; i.e. a set S of points for which every line contains 0 or 2 points of S.

Conics and hyperovals are the most extensively studied objects in Desarguesian finite projective planes; they are connected to a large variety of combinatorial structures and practical applications. To name a few: hyperovals give rise to generalized quadrangles of non-classical parameters [1]; hyperovals are the smallest code words in the projective plane code (until today the only finite geometry code used in engineering applications [6, 7]); hyperovals give rise to bent functions (a cryptographic function with desirable properties) and linear codes with desirable parameters [10]; hyperovals play a crucial role in the study of ovoids in $\mathrm{PG}(3,q)$ [13]; hyperovals are related to α-flocks [5]; etc.

The two main areas of research on hyperovals are construction and classification. Concerning construction, 11 infinite classes are known: the regular hyperoval [2], the translation hyperovals [21], the Segre hyperovals [22], two classes of Glynn hyperovals [8], the Payne hyperovals [14], the Cherowitzo hyperovals [3], three types of Subiaco hyperovals [5, 15, 16] and the Adelaide hyperovals [4]. Next to these, O'Keefe and Penttila [12] found one sporadic example in $\mathrm{PG}(2,32)$ that has not yet been embedded in an infinite class.

For the sake of completeness, a representative o-polynomial for each of the 11 classes is given in Table 1, where $\sigma = 2^{(h+1)/2}$,

$$P(x) = x^{1/6} + x^{3/6} + x^{5/6}, C(x) = x^{\sigma} + x^{\sigma+2} + x^{3\sigma+4}, S_1(x) = \frac{\omega^2(x^4+x)}{x^4+\omega^2x^2+1} + x^{1/2},$$

$$S_2(x) = \frac{\delta^2x^4+\delta^5x^3+\delta^2x^2+\delta^3x}{x^4+\omega^2x^2+1} + \left(\frac{x}{\delta}\right)^{1/2}, S_3(x) = \frac{(\delta^4+\delta^2)x^4+\delta^3x^3+\delta^2x}{x^4+\omega^2x^2+1} + \left(\frac{x}{\delta}\right)^{1/2},$$

$$A(x) = \frac{T(\beta^m)(X+1)}{T(\beta)} + \frac{T((\beta X+\beta^q)^m)}{T(\beta)(X+T(\beta)X^{1/2}+1)^{m-1}} + X^{1/2} \text{ with } T(y) = y^q + y,\ \forall y \in \mathbb{F}_{q^2}.$$

It should be noted that for q even, since fields have prime power order, it follows that $q = 2^h$ for some positive integer h. The conditions on h restrict in which planes the contruction can exist. For $q = 64$, $h = 6$, and the regular and translation hyperovals coincide. Hence, there are four known hyperovals up to isomorphism in this plane: the regular hyperoval, the Adelaide hyperoval, and the first two types of Subiaco hyperovals.

Name	**f(x)**	**conditions**
Regular	x^2	-
Translation	x^{2^i}	$\gcd(h,i)=1$
Segre	x^6	h odd
Glynn I	$x^{3\sigma+4}$	h odd
Glynn II	$x^{\sigma+\lambda}$	h odd, $\lambda = \begin{cases} 2^m \text{ if } h=4m-1 \\ 2^{3m+1} \text{ if } h=4m+1 \end{cases}$
Payne	$P(x)$	h odd
Cherowitzo	$C(x)$	h odd
Subiaco I	$S_1(x)$	$h \equiv 2 \pmod 4$, $\omega^2+\omega+1=0$
Subiaco I	$S_2(x)$	$h \equiv 2 \pmod 4$, $\delta = \zeta^{1-q}+\zeta^{q-1}$ ζ primitive element in $\mathbb{F}_{q^2}$
Subiaco I	$S_3(x)$	$h \not\equiv 2 \pmod 4$, $\mathrm{Tr}_{\mathbb{F}_2}(\delta^{-1})=1$
Adelaide	$A(x)$	$h \geqslant 4$ even, $\beta \in \mathbb{F}_{q^2} \setminus \{1\}$, $\beta^{q+1}=1$ $m = \pm\frac{q-1}{3} \pmod{q+1}$

Table 1: Representative polynomials of the known families of hyperovals in $\mathrm{PG}(2,2^h)$.

Concerning classification, a full classification of the hyperovals is known only for small q. For $q \leqslant 8$, Segre [21] showed in 1957 that all hyperovals consist of a conic plus its nucleus. In 1975, Hall [9] showed that there are exactly two classes of hyperovals in $\mathrm{PG}(2,16)$, which was later confirmed independently by O'Keefe and Penttila [11] without the use of a computer. In 1994, Penttila and Royle [18] showed that there are exactly six classes of hyperovals in $\mathrm{PG}(2,32)$: five constructions listed in Table 1 and one sporadic construction [12]. For $\mathrm{PG}(2,64)$ the number of classes has been a long standing open problem. Using the same technique as for $\mathrm{PG}(2,32)$, the computer search would take thousands of years to complete even on today's hardware.

Attempts so far have however been able to classify specific cases. Penttila and Pinneri [17] classified all hyperovals in $\mathrm{PG}(2,64)$ admitting a collineation of prime order 5 or higher. Later, Penttila and Royle [19] classified all hyperovals in $\mathrm{PG}(2,64)$ admitting a collineation of order 2 or 3. This leaves only the largest case open: the existence of a hyperoval in $\mathrm{PG}(2,64)$ without any nontrivial automorphisms.

That such effort was put into finding even partial results on the classification of hyperovals in $\mathrm{PG}(2,64)$, illustrates the importance of finding a full classification for $\mathrm{PG}(2,64)$, despite it being a result for a single small projective plane. In this paper, we describe an exhaustive search that leads to a negative answer to this question, summarized in the following theorem.

Theorem 4 (Main Theorem)**.** *There are four isomorphism classes of hyperovals in* $\mathrm{PG}(2,64)$*: the regular hyperoval, the Adelaide hyperoval, the Subiaco I hyperoval and the Subiaco II hyperoval.*

In Section 2, we introduce the necessary concepts and machinery to formally describe the search. In Section 3 and 4, we will discuss the first and second part of the search, respectively. In Section 5, we analyze the results and discuss some additional verification mechanisms to ensure the correctness of our results.

2 Outline of the Search

Notation 5. By H_4 we denote the lexicographically smallest 4-arc up to equivalence in $\mathrm{P\Gamma L}(3,64)$, i.e. $H_4 = \{(0,0,1),(0,1,0),(1,0,0),(1,1,1)\}$.

Notation 6. The collineation group of $\mathrm{PG}(2,q)$ is denoted by $\mathrm{P\Gamma L}(3,q)$; it is the group of semilinear projective mappings. Each element in this group is of the form $x \mapsto Ax^\theta$, where $A \in \mathrm{PGL}(3,q)$ and θ is a Frobenius automorphism of $\mathbb{F}_q$, i.e. $x^\theta = x^{2^i}$ for some i, since $q = 2^h$.

Remark 7. It is well known that there are exactly h elements of $\mathrm{P\Gamma L}(3,2^h)$ to map any ordered 4-arc to a given ordering of H_4, and those elements can easily be constructed.

Definition 8. Let S, H be point sets in $\mathrm{PG}(2,q)$ with $S \subseteq H$. The S-orbit of H is the subset of $H^{\mathrm{P\Gamma L}}$ consisting of all images that contain S; we denote this by $R_{H,S}$. The *standard orbit* R_H of H is the H_4-orbit of H, i.e. $R_H = R_{H,H_4}$. The full $\mathrm{P\Gamma L}$-orbit of H is then $R_{H,\emptyset}$.

Note that, in general, $R_{H,S}$ is not an orbit under the group action.

Definition 9. Let E, S be points sets in $\mathrm{PG}(2,q)$. A set $H \supseteq S$ is *S-disjoint* from E if all elements in the S-orbit of H are disjoint from E. A set $H \supseteq H_4$ is *strongly disjoint* from E if it is H_4-disjoint from E.

Since R_H is not a group orbit, classical group techniques will not work here. We will define an equivalence relationship $\equiv_H$ as follows, to mimic the effect of group orbits.

Notation 10. Let S be an arc in $\mathrm{PG}(2,q)$. By $\mathcal{A}(S)$ we denote the set of points that can be added to S without violating the arc constraint.

Notation 11. Let S be an arc in $\mathrm{PG}(2,q)$. Then S defines an equivalence relation $\equiv_S$ on $\mathcal{A}(S)$ as follows: two elements $a, b \in \mathcal{A}(S)$ have $a \equiv_S b$ if and only if $S \cup \{a\}$ and $S \cup \{b\}$ are projectively equivalent. For any $S' \subseteq S$, this property is equivalent to $R_{S\cup\{a\},S'} = R_{S\cup\{b\},S'}$.

Definition 12. A *node* is a pair $(S, \mathcal{C})$ where S is a set of points in $\mathrm{PG}(2,q)$ and $\mathcal{C}$ is a set of pairs (S_i, C_i) with $S_i \subseteq S$ and $C_i \cap S = \emptyset$.

Definition 13. The *solution set* of a node $(S, \mathcal{C})$, denoted by $\psi(S, \mathcal{C})$, is the set of all $\mathrm{P\Gamma L}$-orbits $R_{H,\emptyset}$ for which

- for all $(S_i, C_i) \in \mathcal{C}$ one has $C_i \cap H' = \emptyset$ for all $H' \in R_{H,S_i}$, and
- at least one representative H' contains S.

Notation 14. We will write $C_i \cap R_{H,S_i} = \emptyset$ to denote $C_i \cap H' = \emptyset$ for all $H' \in R_{H,S_i}$. While this is clearly an abuse of the notation, it greatly improves the readability of statements involving this condition for all C_i with $(S_i, C_i) \in \mathcal{C}$.

Now we are ready to describe the search. We will iterate over the vertices of a particular rooted tree, of which we will first define its essential properties.

Definition 15. Let $\mathcal{T}$ be a rooted tree with nodes as its vertices. A non-leaf node is called *good* if its solution set is contained in the union of the solution sets of its children. A leaf node $(S, \mathcal{C})$ is called *good* if either S is a hyperoval meeting the disjointness conditions in $\mathcal{C}$, or the solution set of the node is empty. The tree $\mathcal{T}$ is good if all of its vertices are good.

The aim of this paper is to find $\psi(\emptyset, \emptyset)$. Given a good tree $\mathcal{T}$ with $(\emptyset, \emptyset)$ as the root, this can be done as follows. We first compute the solution sets of all leaf nodes; this is easily done as the leaves of a good tree either have empty solution set, or are nodes of the form $(S, \mathcal{C})$ with S a hyperoval already meeting the disjointness conditions $\mathcal{C}$, hence the solution set is always $\emptyset$ or $\{R_{S,\emptyset}\}$. For non-leaf nodes, repeatedly applying the property that the solution set of a node is contained in the union of the solution sets of its children, one obtains a superset for the solution set of the root. Verifying if all solutions are effectively valid and distinct solutions for the root node, yields the exact value of $\psi(\emptyset, \emptyset)$. Note that the union computation is not trivial, as we represent orbits by a single representative, and expensive calculations need to be performed to determine whether or not two given hyperovals have the same orbit (i.e. are projectively equivalent).

Remark 16. Determining the S_i-orbit of an arc S becomes computationally expensive when S gets larger. Therefore, for larger S, it is beneficial to not explicitly verify that $C_i \cap R_{H,S_i} = \emptyset$ but instead only verify that $C_i \cap S_i = \emptyset$ and only perform the full verification at the resulting leaf nodes. This way, we travel through more nodes than strictly necessary for the tree, but since only $\emptyset$ can appear as extra set in the union, this will not harm the correctness of the algorithm. Skipping this check speeds up the computations significantly and therefore makes up well for the few extra nodes to check.

We have now introduced all the machinery needed to describe the search tree $\mathcal{T}$. The tree will have depth $q + 2$, with depth 0 (the root) being the node $(\emptyset, \emptyset)$, and depth $q + 2$ being the leaf nodes with the desired hyperovals. Computing the union of all solution sets of all leaves at depth $q+2$ (leaves at lower depth are guaranteed to have empty solution sets) yields the desired classification. In Sections 3 and 4 we will describe the exact construction of the tree.

3 The First Part

For arcs S with $|S| \leqslant 3$, the stabilizer of S acts transitively on $\mathcal{A}(S)$, hence there is trivially only one node at this level. The smallest arc S not acting transitively on $\mathcal{A}(S)$ has size 4 and is projectively equivalent to H_4. There are 3782 points in $\mathcal{A}(H_4)$, and $\equiv_{H_4}$ partitions them in 11 classes of size 720, 720, 720, 360, 360, 360, 240, 180, 90, 30, 2 respectively.

We could use this fact to create 11 nodes at depth 5 of the tree, all having $(H_4, \emptyset)$ as their parent node. While this would make $(H_4, \emptyset)$ good, we can obtain the same result in a much better way. To do this, we first consider the unique $\mathbb{F}_4$-subplane spanned by H_4, which we denote by $\mathrm{PG}(2,4)$. Three of these orbits are associated with $\mathrm{PG}(2,4)$ in the following way.

Notation 17. The orbit of size 2 consists of the only two other points in $\mathrm{PG}(2,4)$ that can still be added to the arc. We denote this orbit by $\mathcal{O}_3$ and we denote these two points in $\mathrm{PG}(2,4)$ by p_r and p_s respectively.

Notation 18. The orbit of size 240 and one of the orbits of size 360 are associated to $\mathrm{PG}(2,4)$ in the following way: their union has 600 points and consists of the following 10 disjoint substructures of size 60 each:

(a) the 8 lines spanned by a point of H_4 and a point of $\{p_r, p_s\}$, each time minus the intersection of that line with $\mathrm{PG}(2,4)$;

(b) the unique conic through $H_4 \cup \{p_r\}$ in $\mathrm{PG}(2,64)$ minus its intersection with $\mathrm{PG}(2,4)$; and

(c) the unique conic through $H_4 \cup \{p_s\}$ in $\mathrm{PG}(2,64)$ minus its intersection with $\mathrm{PG}(2,4)$.

We denote this orbit of size 360 by $\mathcal{O}_2$ and the orbit of size 240 by $\mathcal{O}_1$.

Lemma 19. *Let H be any hyperoval in $\mathrm{PG}(2,64)$ containing H_4, let o_1 be any element of $\mathcal{O}_1$, o_2 be any element of $\mathcal{O}_2$ and o_3 be any element of $\mathcal{O}_3$. Then exactly one of the following statements is true.*

- *Some element in R_H contains H_4 and o_1.*
- *Some element in R_H contains H_4 and o_2, and H is strongly disjoint from $\mathcal{O}_1$.*
- *Some element in R_H contains H_4 and o_3, and H is strongly disjoint from $\mathcal{O}_1 \cup \mathcal{O}_2$.*

Proof. The statements are clearly disjoint, so it is sufficient to prove that at least one of them holds. We distinguish the following cases.

- If H is not strongly disjoint from $\mathcal{O}_1$, then by definition of strongly disjoint there exists some $g \in \mathrm{P\Gamma L}(3,64)$ such that H^g contains H_4 and some element $o \in \mathcal{O}_1$. Now, consider the mapping h that maps $H_4 \cup \{o\}$ to $H_4 \cup \{o_1\}$; such h exists by definition of $\mathcal{O}_1$. Then H^{gh} contains o_1 and H_4, which means the first statement is true.

- If H is strongly disjoint from $\mathcal{O}_1$ but not strongly disjoint from $\mathcal{O}_2$, then by definition of strongly disjoint there exists some $g \in \mathrm{P\Gamma L}(3,64)$ such that H^g contains some element $o \in \mathcal{O}_2$. Since being strongly disjoint is a property of the standard orbit and H^g still contains H_4, it follows that H^g is still strongly disjoint from $\mathcal{O}_1$. Now, again consider the mapping h that maps $H_4 \cup \{o\}$ to $H_4 \cup \{o_2\}$; such h exists by definition of $\mathcal{O}_2$. Then H^{gh} contains o_2 and H_4. Moreover, for the same reason as before, H^{gh} is still strongly disjoint from $\mathcal{O}_1$. Hence, the second statement is true.

- If H is strongly disjoint from both $\mathcal{O}_1$ and $\mathcal{O}_2$, then it is strongly disjoint from $\mathcal{O}_1 \cup \mathcal{O}_2$. Since $\langle (0,0,1), p_r \rangle$ contains at least one point of H, it must contain a second point of H. Since all points other than p_r are either collinear with two points of H_4 or are element of $\mathcal{O}_1 \cup \mathcal{O}_2$, it follows that H cannot be strongly disjoint from $\mathcal{O}_3$. Hence, we can repeat the argument from the second bullet, replacing $\mathcal{O}_2$ by $\mathcal{O}_3$ and replacing $\mathcal{O}_1$ by $\mathcal{O}_1 \cup \mathcal{O}_2$, to obtain the conclusion that the third statement is true. □

Corollary 20. *In terms of node solution sets, Lemma 19 is equivalent to*

$$\psi(H_4, \emptyset) = \psi(H_4 \cup \{o_1\}, \emptyset) \cup \psi(H_4 \cup \{o_2\}, \{(H_4, \mathcal{O}_1)\}) \cup \psi(H_4 \cup \{o_3\}, \{(H_4, \mathcal{O}_1 \cup \mathcal{O}_2)\}).$$

The above theorem gives us a very strong starting seed for the search (with only three 5-arcs to start from, two of which already exclude hundreds of points). We can formally generalize the idea behind Lemma 19 in the following way.

Lemma 21. *Let $(S, \mathcal{C})$ be a node, and let L be a projective line tangent to S. Partition the set $\mathcal{A}(S) \setminus \cup_{(S_i, C_i) \in \mathcal{C}} C_i$ into its $\equiv_S$-equivalence classes and let $W_1, \ldots, W_m$ be the classes that have nonempty intersection with L and simultaneously have $R_{S \cup \{w\}, S_i} \cap C_i = \emptyset$ for all $(S_i, C_i) \in \mathcal{C}$. Pick arbitrary $w_i \in W_i \cap L$ for $i = 1, \ldots, m$ and let $W = \{w_1, \ldots, w_m\}$. Then, regardless of the choice of the w_i and regardless of the ordering of $W_1, \ldots, W_m$,*

$$\psi(S, \mathcal{C}) = \bigcup_{i=1}^{m} \psi\left(S \cup \{w_i\}, \mathcal{C} \cup \{(S, W_1 \cup \cdots \cup W_{i-1})\}\right).$$

Proof. Let $R \in \psi(S, \mathcal{C})$. For any representative $H \supset S$ of R, the unique point $w \in L \cap H' \setminus S$ belongs to exactly one of $W_1, \ldots, W_m$, we denote $i(H)$ the unique value for which $w \in W_{i(H)}$. Now, among all $H' \in R$ with $H' \supset S$, select one for which i is minimal. Then clearly, $\cup_{H \in R} R \cap (W_1 \cup \cdots \cup W_{i-1}) = \emptyset$ and $w \subseteq W_i$. By definition of $\equiv_S$, we can find $g \in \mathrm{P\Gamma L}(3, q)$ mapping $S \cup \{w\}$ to $S \cup \{w_i\}$, and hence H'^g is a representative of R containing S and meeting all disjointness requirements, i.e. $R \in \psi\left(S \cup \{w_i\}, \mathcal{C} \cup \{(S, W_1 \cup \cdots \cup W_{i-1})\}\right)$. □

Remark 22. If $S \neq \emptyset$ has no tangent lines, it is a hyperoval. If S has a tangent line where $W = \emptyset$, then it cannot be completed to a hyperoval and $\psi(S, \mathcal{C}) = \emptyset$ regardless of $\mathcal{C}$.

Remark 23. Lemma 19 is a direct corollary of Lemma 21, using $L = \langle (0,0,1), p_r \rangle$ and $(W_1, W_2, W_3) = (\mathcal{O}_1, \mathcal{O}_2, \mathcal{O}_3)$.

Lemma 21 is the core branching technique in the first part of the search. It can be very potent, but its power depends entirely on the right choice of the line L and the right order of the sets $W_1, \ldots, W_m$. While it is impossible to provably select the optimal choice for these, it is also unnecessary: the lemma is mathematically correct for every choice, so it is sufficient to find a 'rather good' choice.

We tried different heuristics to select these values, and selected the best among our attempts (i.e. the one having the smallest number of nodes at early levels). The experimentally best performing heuristic was the following one. For depth 4, manually pick the choices from Remark 23. For depth 5 and above, compute $\mathcal{A}(S)$, remove those the points would give an arc violating $\mathcal{C}$, compute the $\equiv_S$-classes $\mathcal{W}' = \{W_1', \ldots, W_M'\}$ on what remains of $\mathcal{A}(S)$, and then select the line that minimizes $\sum_{W_i' \in \mathcal{W}'} \frac{|W_i'|}{|W_i' \cap L|}$ (as a compromise between least children on one hand, and children with as large $\equiv$-classes as possible on the other hand). Now, let $\mathcal{W} \subseteq \mathcal{W}'$ be the subset of classes that intersect non-trivially with L. Within $\mathcal{W}$, we ordered the W_i in non-increasing order of $|W_i| - |W_i \cap L|$, in order to exclude as many points as possible in as many nodes as possible.

Clearly, computing L is a very computationally expensive operation. The main part of the computational effort goes into computing the $\equiv_S$-classes and into verifying which $\mathcal{A}(S)$-elements violate $\mathcal{C}$ when added to S. The following observations are crucial in making this computation feasible.

- For $|S| \geqslant 7$, the vast majority of the cases will have $\equiv_S$ trivial, i.e. every point is its own $\equiv_S$-class. This means we need to either compute $\binom{|\mathcal{A}(S)|}{2}$ projective equivalence tests, or $|\mathcal{A}(S)|$ different S-orbits, while the result is that they are all different. This is however relatively easy to detect using an invariant as follows. Associate integer values to all points in $\mathrm{PG}(2,64)$, all starting at 0. For each of the $\binom{|S|}{4}$ 4-subsets of S, compute its unique skew line in the Fano plane spanned by these points, and for each such line, increase all points on it by one. Then, replace all values by a hashed value of them, using any (fixed) hash function one desires. Then sum up the hashed values for all points in $\mathrm{PG}(2,q)$. This hash value is a projective invariant, and computing this hash value $|\mathcal{A}(S)|$ times is a very small cost. Any point $a \in \mathcal{A}(S)$ for which $S \cup \{a\}$ has a unique hash value (i.e. one not appearing for any $S \cup \{b\}$ with $b \in \mathcal{A}(S)$) is guaranteed to have a $\equiv_S$-orbit containing only this element.

- The required time to verify if the conditions in $\mathcal{C}$ are met, does not increase linearly with $|\mathcal{C}|$. If $\mathcal{C} = \{(S_1, C_1), \ldots, (S_n, C_n)\}$, it is guaranteed that $S_1 \subset \cdots \subset S_n$. Hence, when computing the full standard orbit of S and filtering which elements contain S_i, once can first filter for S_1 and check disjointness from C_1, then filter that set for S_1 and check disjointness from C_2, etc.

Despite the above efforts, this method quickly becomes unfeasible, especially in verifying the conditions in $\mathcal{C}$. The cost per arc is $\mathcal{O}(|S|^4)$ and the number of nodes per level grows exponentially in the beginning, so this approach quickly becomes unfeasible. Still, we managed to pursue this approach to generate an optimal set of nodes at 9 with the aid of a computing cluster, yielding 19 nodes at depth 6, 622 nodes at depth 7, 24230 nodes at depth 8, and 777418 nodes at depth 9. Beyond depth 9, the computations become infeasible and would take up a significant portion of the total computation time, far beyond what it would save compared to less optimized approaches.

For this reason, we split the search in two parts. The first part is the method that we just described, and generates a set of 997418 nodes with 9-arcs which are guaranteed to be sufficient to generate all isomorphism classes of hyperovals when completed. The second part will focus on completing each of them as quickly as possible. As an illustration of the power of Lemma 19, out of these 997418 nodes, 997386 are child nodes of $(H \cup \{o_1\}, \emptyset)$, one is a child node of $(H \cup \{o_2\}, \mathcal{O}_1)$, and 31 are child nodes of $(H \cup \{o_3\}, \mathcal{O}_1 \cup \mathcal{O}_2)$. Hence, the latter two subtrees are almost eliminated at this point, showing that Lemma 19 gave us one extra arc point almost for free.

4 The Second Part

Now, we need to complete each of our 997418 distinct 9-arcs to hyperovals. At this point, group-based tricks like the $\equiv_S$-orbits provide no substantial benefits anymore. The disjointness conditions in $\mathcal{C}$ only become stronger as $|S|$ grows… but the computational cost grows even faster. Hence, at this point we decided to go for a more straightforward approach, greatly reducing the cost per node rather than wanting the absolute minimum number of nodes. For this purpose, we introduce the following weakening of Lemma 21.

Lemma 24. *Let $(S, \mathcal{C})$ be a node, and let L be a projective line tangent to S. Let $W = \{w \in (L \cap \mathcal{A}(S)) \setminus \cup_{(S_i,C_i)\in\mathcal{C}} C_i\}$ and remove from W all points for which there is some i for which $S \cup \{w\} \cap C_i \neq \emptyset$. Then $\psi(S, \mathcal{C}) \subseteq \bigcup_{w\in W} \psi(S \cup \{w\}, \mathcal{C})$.*

Proof. Let $R_{H,\emptyset} \in \psi(S, \mathcal{C})$, and let H be a representative containing S. Since H is a hyperoval and L is a tangent to S, H must contain exactly one other point $w \in (L \cap \mathcal{A}(S)) \setminus \cup_{(S_i,C_i)\in\mathcal{C}} C_i$. Hence, $R_{H,\emptyset} \in \psi(S \cup \{w\}, \mathcal{C})$. □

Equality in the solution sets is not guaranteed, because we have not explicitly checked the disjointness requirements. However, as argued early, it is sufficient to find a superset of $\psi(\emptyset, \emptyset)$ and filter out unacceptable or duplicate solutions.

The basics of the search are still the same. We find the best tangent to branch from, and for every possible second point on the secant, we recursively call the construction routine. When no secants exist, a hyperoval is found. When a tangent line with no possible second points is found, the arc can never be completed to a hyperoval and hence the solution set of this node is empty. The best line, in this case, will be the line that gives the least number of child nodes, i.e. the secant line with the least number of points that could not already be excluded.

Practically, we will assign a number to each point, representing the number of conditions that prevent that point from being added to the arc. Whenever a point is added, all points on secant (previously tangent) lines through that point get this number increased by one; when a point is removed, all points on tangent (previously secant) lines through it get their number decreased by one. We start by assigning 0 to all points, raising the number by 1 for all points in $\bigcup_{(S_i,C_i)\in\mathcal{C}} C_i$, and adding the points of S one by one according to the above procedure. The points that can be added to S at any time, are then the points for which this value is still 0.

To find the best line, one needs several nested loops. Therefore, it is advisable to keep track of some intermediary values, such as whether a line is skew, tangent or secant; and how many points with associated value 0 are left on each line. This way, the branching is kept to a minimum, but the cost per node is still very low. For the same reason, it is also advisable to precompute some properties of the incidence geometry, such as the points on every line, the lines through every point, and the line spanned by two points. These can be generated one time at the start, so they do not impact the total computing time.

In total, each 9-arc (i.e. each node at depth 9) has a subtree of about $4 \cdot 10^{10}$ nodes, where depth 14 has 0.5% of the nodes, depth 15 has 3.6%, depth 16 has 17.3%, depth 17 has 41.8%, depth 18 has 35.8% and depth 19 has 1.0%. Depths below 14 and depths above 19 represent a negligible fraction of the total number of nodes in the subtree. We managed to save some work by not selecting the best line at depths 17 and above, as here the tree will complete soon anyway, but instead selecting a 'rather good' line that we could find quickly. From these arc sizes onwards, selecting the best line becomes less and less important, while the computational cost for finding the best line is at its largest here. This way, we managed to get our implementation sufficiently fast to complete a single 9-arc in only a few hours. This brings the total computation workload at a few million hours of CPU time, which is within reach of today's High Performance Computing clusters.

5 Results

In total, our search yielded 42530 hyperovals. We now need to find the equivalence class of each hyperoval, and determine whether or not any new, previously unknown hyperovals occur. This can be done very efficiently in the following way. For each of the known types H, we explicitly compute the H_4-orbit of H, using Remark 7. This takes a few hours, but it is a one-time effort. Now, it is sufficient to test for each hyperoval (which always contains H_4 because of they way our search works) whether or not it belongs to one of these four H_4-orbits. This takes less than 0.01 seconds per hyperoval, and hence does not present a computational challenge even on a single computer. We find 2 regular hyperovals, 21262 Adelaide hyperovals, 17040 Subiaco I hyperovals, 4226 Subiaco II hyperovals, and 0 new hyperovals. Given that the automorphism group sizes are 1572480, 12, 15 and 60, the number of hyperovals found is roughly inversely proportional to the size of the automorphism group, as one would indeed expect. When filtering out the hyperovals not meeting the disjointness conditions of their nodes, we are left with 2 regular hyperovals, 862 Adelaide hyperovals, 695 Subiaco I hyperovals and 217 Subiaco II hyperovals.

As a final step, since this is a computer search with negative outcome, it is good practice to perform extensive verifications that the results are correct. Since the main part of our search consists of completing a set of 9-arcs to a set of hyperovals containing the 9-arc, and since this is the most error prone part (happening on remote computer clusters with little checks or additional output, and too large to keep a human eye on), we want some verification that we have done this correctly.

Fortunately, a strong verification mechanism exists in this case: it is possible to compute how many hyperovals of known types we *should* find for a given 9-arc. To determine these expected numbers, we compute again the standard orbit R_H for each of the known types H. Then, for each 9-arc S, we consider all elements of R_H that contain it, i.e. we compute $R_{H,S}$; this is relatively cheap in practice as R_H has already been computed and one usually detects very early that a hyperoval of R_H is not in $R_{H,S}$. This way, we found exactly the same output numbers for each 9-arc and for each hyperoval type as in the actual search (both with and without $\mathcal{C}$-filtering), providing strong evidence that the presented results are indeed correct.

We conclude by again stating our main theorem and a short evaluation for the next case.

Theorem 25 (Main Theorem)**.** *There are four isomorphism classes of hyperovals in* $\mathrm{PG}(2,64)$*: the regular hyperoval, the Adelaide hyperoval, the Subiaco I hyperoval and the Subiaco II hyperoval.*

A natural next question is if the improved method in this paper is sufficient to handle the next open case as well. Using the state of the art computer hardware, and by completing the search for a large number of randomly selected subtrees, we estimate that the classification of the hyperovals in $\mathrm{PG}(2,128)$ would take over 10^{25} years of CPU time to complete. It is therefore highly unlikely that we will see this classification anytime soon.

Remark 26. One of the motivations for classifying the hyperovals in $\mathrm{PG}(2,32)$ was that it implies a classification of the ovoids (sets of q^2+1 points, no three collinear) in $\mathrm{PG}(3,32)$. The same could now be done for $\mathrm{PG}(3,64)$, revealing whether or not any examples exist other than the elliptic quadric $Q^+(3,q)$. T. Penttila already solved this problem prior to our research, under the assumption that no new hyperovals would be found, and will publish this result soon.

Acknowledgment.

The author wants to thank T. Penttila for the interesting conversations on this topic.

References

[1] R.W. Ahrens and G. Szekeres, ‘On a combinatorial generalization of 27 lines associated with a cubic surface’, *J. Austral. Math. Soc.* 10 (1969), 485–492.

[2] R.C. Bose, ‘Mathematical theory of the symmetrical factorial design’, *Sankhya* 8 (1947), 107–166.

[3] W.E. Cherowitzo, ‘α-flocks and hyperovals’, *Geom. Dedicata* 72 (1998), 221–246

[4] W.E. Cherowitzo, C.M. O’Keefe, and T. Penttila, ‘A unified construction of finite geometries in characteristic 2’, *Adv. Geom.* 3 (2003), 1–21.

[5] W.E. Cherowitzo, T. Penttila, I. Pinneri and G.F. Royle, ‘Flocks and ovals’, *Geom. Dedicata* 60 (1996), 17–37.,

[6] I.B. Djordjevic and B.V. Vasic, ‘Projective geometry LDPC codes for ultralong-haul WDM high-speed transmission’, *IEEE Photonics Technology Letters* 15 (2003), 784–786.

[7] I.B. Djordjevic, S. Sankaranarayanan and B.V. Vasic, 'Projective-Plane Iteratively Decodable Block Codes for WDM High-Speed Long-Haul Transmission Systems', *J. Lightwave Technol.* 22 (2004), 695–702.

[8] D.G. Glynn, 'Two new sequences of ovals in finite Desarguesian planes of even order', Combinatorial Mathematics X (ed. L.R.A. Casse), Lecture Notes in Mathematics 1036, Springer, 1983, 217–229.

[9] M. Hall, 'Ovals in the Desarguesian plane of order 16', *Ann. Mat. Pura Appl.* 102 (1975), 159–176.

[10] S. Mesnager, 'Bent vectorial functions and linear codes from o-polynomials', *Des. Codes Cryptogr.* 77 (2015), 99–116.

[11] C.M. O'Keefe and T. Penttila, 'Hyperovals in PG(2, 16)', *European J. Combin.* 12 (1991), 51–59.

[12] C.M. O'Keefe and T. Penttila, 'A new hyperoval in PG(2,32)', *J. Geom.* 44 (1992), 117–139.

[13] C.M. O'Keefe, T. Penttila, G.F. Royle, 'Classification of ovoids in PG(3,32)', *J. Geom.* 50 (1994), no. 1-2, 143–150.

[14] S.E. Payne, 'A new infinite family of generalized quadrangles', *Congr. Numer.* 49 (1985), 115–128.

[15] S.E. Payne, 'A tensor product action on q-clan generalized quadrangles with $q = 2^e$', *Linear Algebra Appl.* 226-228 (1995), 115–137.

[16] S.E. Payne, T. Penttila, and I. Pinneri, 'Isomorphisms between Subiaco q-clan geometries', *Bull. Belg. Math. Soc.* 2 (1995), 197–222.

[17] T. Penttila and I. Pinneri, 'Irregular hyperovals in PG(2,64)', *J. Geom.* 51 (1994), 89–100.

[18] T. Penttila and G.F. Royle, 'Classification of hyperovals in PG(2,32)', *J. Geom.* 50 (1994), 151–158.

[19] T. Penttila and G.F. Royle, 'On hyperovals in small projective planes', *J. Geom.* 54 (1995), 91–104.

[20] B. Segre, 'Ovals in a finite projective plane', *Canad. J. Math.* 7 (1955), 414–416.

[21] B. Segre, 'Sui k-archi nei piani finiti di caratteristica due', *Rev. Math. Pures Appl.* 2 (1957), 289–300.

[22] B. Segre, 'Ovali e curve σ nei piani di Galois di caratteristica due', *Atti dell' Accad. Naz. Lincei Rend.* (8) 32 (1962), 785–790.