Serdica J. Computing **8** (2014), No 1, 47–70

**Serdica**
Journal of Computing

Bulgarian Academy of Sciences
Institute of Mathematics and Informatics

# CONSTRUCTING 7-CLUSTERS

Sascha Kurz, Landon Curt Noll, Randall Rathbun, Chuck Simmons

Abstract. A set of $n$ lattice points in the plane, no three on a line and no four on a circle, such that all pairwise distances and coordinates are integers is called an $n$-cluster (in $\mathbb{R}^2$). We determine the smallest 7-cluster with respect to its diameter. Additionally we provide a toolbox of algorithms which allowed us to computationally locate over 1000 different 7-clusters, some of them having huge integer edge lengths. Along the way, we have exhaustively determined all Heronian triangles with largest edge length up to $6 \cdot 10^6$.

**1. Introduction.** Point sets with pairwise rational or integers distances have been studied for a long time, see e.g. [21]. For brevity we will call those point sets rational or integral. Nevertheless, only a few theoretical results are known; integral point sets seem to be unexpectedly difficult to construct. On the other hand there is the famous open problem, asking for a dense set in the plane such that all pairwise Euclidean distances are rational, posed by Ulam in 1945 [36]. As of now we only know constructions of rational point sets which are dense either on a line or a circle, see e.g. [3, Sec. 5.11] or [1]. In [34] the authors have shown

*ACM Computing Classification System* (1998): G.2, G.4.
*Key words:* Erdős problems, integral point sets, Heron triangles, exhaustive enumeration.

that no irreducible algebraic curve other than a line or a circle contains an infinite rational set. Thus if Ulam's question admits a positive answer the corresponding point set has to be very special.

Almering [12] established that, for a given triangle with rational side lengths, the set of points with rational distances to the three vertices is dense in the plane of the triangle. Berry [15] relaxed the conditions to one rational side length and the other two side lengths being a square root of a rational number. More general considerations can be found in the preprint [13]. So far no such result is known for a quadrilateral with pairwise rational distances. Dubickas states in [16] that every $n \geq 3$ points in $\mathbb{R}^2$ can be slightly perturbed to a set of $n$ points in $\mathbb{Q}^2$ such that at least $3(n-2)$ of the mutual distances are rational. So, for $n = 5$ just 1 distance may be non-rational. Declaring which of the mutual distances have to be rational can be modeled as a graph. Classes of *admissible* graphs have been studied, see e.g. [14, 17].

Given a finite rational point set, we can of course convert it into an integral point set by rescaling its edge lengths with the least common multiple of their respective denominators.[1] Thus, for each finite number $n$ one can easily construct an integral point set consisting of $n$ points where all points are located on a circle. Several constructions of finite integral point sets, where $n-4$ points are located on a line or $n-3$ points are located on a circle, are known, see e.g. [3, Sec. 5.11]. To this end several authors, including Paul Erdős [10, Problem D20], ask for integral point sets in general position, meaning that no three points are on a line and no four points are on a circle. These objects seem to be rather rare or at the very least hard to find. For $n = 6$ points a few general constructions for integral point sets in general position are known [19]. The only two published examples of 7-point integral point sets in general position are given in [20]. Independently and even earlier, in May 2006 Chuck Simmons and Landon Curt Noll found[2] more restricted configurations. Their *smallest* example has integer coordinates:

$$(0,0),\ (327990000,0),\ (238776720,118951040),\ (222246024,-103907232),$$
$$(243360000,21896875),\ (198368352,50379264),\ (176610000,-94192000).$$

Aiming at $n$-point integral point sets in general position, especially $n = 6$, Noll and Bell [30] additionally required that the coordinates also have to be integers. They called those structures $n_2$-clusters, or when the restriction to the dimension[3] is clear from the context, $n$-clusters. Using a computer search, the

[1] As shown in [2, 8] each infinite integral point set is located on a line.
[2] cf. http://www.isthe.com/chongo/tech/math/n-cluster/
[3] The notion of an integral point set can be easily generalized to arbitrary dimensions $m$. The

authors found 91 non-similar 6-clusters, where the respective greatest common divisor of their corresponding edge lengths is one, but no 7-clusters.[4] Using a slightly improved version and lots of computing time, Simmons and Noll found the first 7-clusters in 2006 and extended their list to twenty-five 7-clusters in 2010.

The aim of this paper is to present a set of sophisticated algorithms in order to construct $n$-clusters for $n \geq 5$. Using an exhaustive search, we were able to determine the smallest 7-cluster, with respect to its diameter, and provide heuristic methods to produce more than 1000 non-similar 7-clusters. Unfortunately, no 8-cluster turned up. So the hunt for an integral octagon in general position or even an 8-cluster is still open. In this context we mention the Erdős/Noll "infinite-or-bust" $n_m$-cluster conjecture: For any dimension $m > 1$, and any number $n > 2$ of points, there exists either 0 or an infinite number of primitive $n_m$-clusters.

In Section 2 we summarize the known theory on integral point sets, and in Section 3 we go into the algorithmic details of how to generate large lists of Heronian triangles. Section 4 is devoted to exhaustive searches for $n$-clusters up to a given diameter. Here the idea is to combine $n$-clusters that share a common $n-1$-cluster. Allowing the containment of similar $(n-1)$-clusters, i.e. a scaled version, is the idea behind Section 5. Our most successful algorithmic approach is presented in Section 6. Since the basic operations of our algorithms have to be performed quite often, we present low level details in Section 7. A theoretically interesting algorithm, based on circle inversion, is presented in Section 8. Since almost all of our presented algorithms depend on a selection of Heronian triangles, which may not be too large due to computational limits, we present ways to select Heronian triangles from larger sets in Section 9. Our computational observations are summarized in Section 10. We present our computational results in Section 11 before we draw a conclusion in Section 12.

## 2. Basic results and notation.

**Definition 2.1.** *An integral point set $\mathcal{P}$ is a set of points in the plane, not all on a line, such that the pairwise distances are integers.*

We note that integral point sets can easily be defined in arbitrary dimensions, see e.g. [23, 26]. Here we restrict ourselves to the two-dimensional case.

One of the first questions arising when dealing with integral point sets is how to represent them. Of course, one may use a list of coordinates. One example

---

term *general position* then has the meaning that no $m+1$ points are contained in a hyperplane and no $m+2$ points are contained in a hypersphere, see e.g. [30].

[4] Independently, Randall Rathbun found a few 6-clusters.

of such a representation is given in the introduction. Another way is to provide a table of the pairwise distances – from which a coordinate representation can easily be computed. For the example from the introduction we have the following distance table:

$$\begin{pmatrix}
0 & 327990000 & 266765200 & 245336520 & 244343125 & 204665760 & 200158000 \\
327990000 & 0 & 148688800 & 148251480 & 87416875 & 139067760 & 178292000 \\
266765200 & 148688800 & 0 & 223470520 & 97162325 & 79592240 & 222024000 \\
245336520 & 148251480 & 223470520 & 0 & 127563605 & 156123240 & 46658680 \\
244343125 & 87416875 & 97162325 & 127563605 & 0 & 53249365 & 133911125 \\
204665760 & 139067760 & 79592240 & 156123240 & 53249365 & 0 & 146199440 \\
200158000 & 178292000 & 222024000 & 46658680 & 133911125 & 146199440 & 0
\end{pmatrix}$$

Given a matrix of distances one can decide whether there exists a set of vertices in the $m$-dimensional Euclidean space $\mathbb{R}^m$ attaining those distances, based on a set of inequalities and equations involving the so-called Cayley-Menger determinants [24, 29].

**Definition 2.2.** *If $\mathcal{P}$ is a point set in $\mathbb{R}^m$ with vertices $v_0, v_1, \ldots, v_{n-1}$ and $C = (d_{i,j}^2)$ denotes the $n \times n$ matrix given by $d_{i,j}^2 = \|v_i - v_j\|_2^2$ the Cayley-Menger matrix $\hat{C}$ is obtained from $C$ by bordering $C$ with a top row $(0,1,1,\ldots,1)$ and a left column $(0,1,1,\ldots,1)^T$. With this, the Cayley-Menger determinant $CMD(\{v_{i_0}, v_{i_1}, \ldots, v_{i_{r-1}}\})$ is given by $\det \hat{C}$.*

**Theorem 2.3** (Menger [29])**.** *A set of vertices $\{v_0, v_1, \ldots, v_{n-1}\}$ with pairwise distances $d_{i,j}$ is realizable in the Euclidean space $\mathbb{R}^m$ if and only if*

$$(-1)^r CMD(\{v_{i_0}, v_{i_1}, \ldots, v_{i_{r-1}}\}) \ge 0,$$

*for all subsets $\{i_0, i_1, \ldots, i_{r-1}\} \subset \{0, 1, \ldots, n-1\}$ of cardinality $r \le m+1$, and*

$$(-1)^r CMD(\{v_{i_0}, v_{i_1}, \ldots, v_{i_{r-1}}\}) = 0,$$

*for all subsets of cardinality $m+2 \le r \le n$.*

Thus it is possible to deal with integral point sets by storing their pairwise distances only. Nevertheless it is often computationally cheaper to use coordinate representations which are easy to compute, see Section 7.2. As remarked in the introduction, we are interested in integral point sets in the Euclidean plane $\mathbb{R}^2$ with some additional properties.

**Definition 2.4.** *An integral point set is in general position, if no three points are on a line and no four points are on a circle.*

For the plane it suffices to check the triangle inequality in order to detect three collinear points. Checking the condition of Ptolemy's theorem, one can easily detect when four points lie on a circle. For general dimensions $m \geq 2$ the conditions of general position can be expressed using Cayley-Menger determinants, see e.g. [23, 24].

**Definition 2.5.** *An $n$-cluster is a plane integral point set in general position that consists of $n$ points such that there exists a representation using integer coordinates, i.e., lattice points.*

Fortunately we do not have to deal with the constraint of integral coordinates. For an explanation we have to go far afield: The area $A_\Delta(a,b,c)$ of a triangle with side lengths $a$, $b$, $c$ is given by

$$A_\Delta(a,b,c) = \frac{\sqrt{(a+b+c)(a+b-c)(a-b+c)(-a+b+c)}}{4}$$

due to the Heron formula. If the area is non-zero, we can uniquely write $A_\Delta(a,b,c) = q\sqrt{k}$ with a rational number $q$ and a square-free integer $k$. The number $k$ is called the characteristic of the triangle with side lengths $a$, $b$, $c$. Kemnitz [19] has shown that each non-degenerate triangle of an integral point set has the same characteristic, which was also generalized to arbitrary dimensions in [24]. Since triangles with integral coordinates have a rational area, see e.g. Pick's theorem, the triangles of an $n$-cluster all have to have a characteristic of 1.

We now argue that the opposite is also true. Given an integer sided triangle with characteristic 1, we can easily determine a representation using rational coordinates, see e.g. [24]. Due to Fricke [9], see also [28, 37], each integral point set in the plane which has a representation in rational coordinates has a representation in integral coordinates.

**Lemma 2.6.** *Let $\mathcal{P} \subseteq \mathbb{R}^2$ be a point set with pairwise integral distances. If $\mathcal{P}$ contains a non-degenerated triangle with characteristic* 1*, then $\mathcal{P}$ permits a representation in $\mathbb{Z}^2$.*

Thus, there is no need to explicitly search for integral coordinates for $n$-clusters. One just needs to check that all pairwise distances are integral and that at least one contained non-degenerate triangle has characteristic 1 or, equivalently, that it has a representation in rational coordinates, to ensure the existence of a representation with integral coordinates.

A Heronian triangle is traditionally defined as a triangle with integer side lengths and area.[5] From the formula for $A_\Delta(a,b,c)$ we can deduce that the area

[5]Some authors allow the side lengths and the area of the Heronian triangle to be rational and remark that all quantities can be easily rescaled to be integers.

of an integer sided triangle with characteristic 1 is rational. To conclude that the area is indeed integral one may consider the cases of the side lengths modulo 8 (see [6]). We summarize these findings in:

**Lemma 2.7.** *Given a non-degenerate triangle $T$ with integer side lengths then the following statements are equivalent:*

(a) *$T$ has characteristic* 1

(b) *$T$ has rational area*

(c) *$T$ has integrer area, i.e., $T$ is Heronian.*

Thus Heronian triangles are the basic building blocks of $n$-clusters and we will consider algorithms how to generate them in the next section.

In the introduction we have spoken of *the smallest* cluster. So in order to have a measure of the *size* of an $n$-cluster, or more generally an integral point set, we denote the largest distance between two points as its diameter. If we perform an exhaustive search in the following, we will always impose a limit on the maximum diameter. We note that other metrics are possible too, but most of them can be bounded by constants in terms of the maximum diameter.

Given an $n$-cluster, we can obviously construct an infinite sequence of non-isomorphic $n$-clusters by rescaling the clusters by integers $2, 3, \ldots$. We call those $n$-clusters similar, and we are generally interested in lists of non-similar $n$-clusters. To this end we call a given $n$-cluster primitive if its edge lengths do not have a common factor larger than 1. As argued before, dividing the edge lengths of a given integral point set by the greatest common divisor does not destroy the property of admitting integral coordinates.

Applying this insight to the example given in the introduction, we observe that the greatest common divisor of the edge lengths is 145. Thus dividing all edge lengths by 145 gives the following distance matrix:

$$\begin{pmatrix} 0 & 2262000 & 1839760 & 1691976 & 1685125 & 1411488 & 1380400 \\ 2262000 & 0 & 1025440 & 1022424 & 602875 & 959088 & 1229600 \\ 1839760 & 1025440 & 0 & 1541176 & 670085 & 548912 & 1531200 \\ 1691976 & 1022424 & 1541176 & 0 & 879749 & 1076712 & 321784 \\ 1685125 & 602875 & 670085 & 879749 & 0 & 367237 & 923525 \\ 1411488 & 959088 & 548912 & 1076712 & 367237 & 0 & 1008272 \\ 1380400 & 1229600 & 1531200 & 321784 & 923525 & 1008272 & 0 \end{pmatrix}$$

This 7-cluster has a diameter of 2 262 000, which is the smallest possible as verified in Section 4. A coordinate representation is given by

$$(0,0),\ (374400,-2230800),\ (1081600,-1488240),\ (-453024,-1630200),$$
$$(426725,-1630200),\ (569088,-1291680),\ (-439040,-1308720).$$

**3. Generation of Heronian triangles.** The conceptually simplest algorithm to exhaustively generate all Heronian triangles up to a given diameter is to loop over all non-isomorphic integer triangles and to check whether the area is integral. This leads to time complexity $\Theta(n^3)$. Two $O(n^{2+\varepsilon})$ algorithms, where $\varepsilon > 0$ is arbitrary, have been given in [25]. We give and apply another $O(n^{2+\varepsilon})$ algorithm here.

Complete parameterizations have been known for a long time: the Indian mathematician Brahmagupta (598-668 A.D.) gives the parametric solution

$$a = \frac{p}{q}h(i^2+j^2), \quad b = \frac{p}{q}i(h^2+j^2), \quad \text{and} \quad c = \frac{p}{q}(i+h)(ih-j^2)$$

for positive integers $p$, $q$, $h$, $i$, and $j$ fulfilling $ih > j^2$ and $gcd(p,q) = gcd(h,i,j) = 1$, see e.g. [5, 25].

Due to the presence of the denominators $q$, this parameterization is not strongly compatible with restrictions on the maximum diameter. We can easily generate primitive Heronian triangles by looping over all feasible triples $(h,i,j)$ below a suitable upper bound, setting $p$ to 1 and choosing $q$ such that $gcd(a,b,c) = 1$. Using this approach we can quickly generate a huge number of primitive Heronian triangles. But we may get those with small diameters rather late, compared to the upper bound on $h,i,j$, and have to face the fact that the same primitive Heronian triangle may be generated several times over.

For the purpose of this paper we use a different exhaustive algorithm to generate all primitive Heronian triangles up to a prescribed diameter. Given a triangle with side lengths $a$, $b$, and $c$ we have $\cos\alpha = \frac{b^2+c^2-a^2}{2bc}$ and $\sin\alpha = \frac{2A_\Delta(a,b,c)}{bc}$. For a Heronian triangle $\sin\alpha$ and $\cos\alpha$ are rational numbers so that also $\tan\frac{\alpha}{2} = \frac{\sin\alpha}{1+\cos\alpha} \in \mathbb{Q}$. Thus, there are coprime integers $m,n$ satisfying $\tan\frac{\alpha}{2} = \frac{n}{m}$. With these parameters we obtain

$$\cos\alpha = \frac{1-\tan^2\frac{a}{2}}{1+\tan^2\frac{a}{2}} = \frac{m^2-n^2}{m^2+n^2},$$

where $\gcd(m^2-n^2, m^2+n^2) \in \{1,2\}$. We conclude that $m^2+n^2$ divides $4bc$. So, given two integral side lengths $b$ and $c$ of a Heronian triangle, we can determine all possibilities for $m^2+n^2$, then determine all possibilities for $m$ and $n$, and finally determine all possibilities for the third side $a$:

**Algorithm 3.1** (Find the third side)**.**
*loop over all divisors* $k$ *of* $2bc$
*loop over all solutions* $(m,n)$ *of* $m^2+n^2=k$
*solve* $\dfrac{b^2+c^2-a^2}{2bc} = \dfrac{m^2-n^2}{m^2+n^2}$ *for* $a$
*if* $a \in \mathbb{Q}$ *and the triangle inequalities are strictly satisfied for* $(a,b,c)$
*then output* $a$

So, in order to determine all primitive Heronian triangles up to diameter $N$, we have to loop over all coprime pairs $(b,c)$ with $N \geq b \geq c \geq 1$ and apply the above algorithm to determine $a$. Given $a$, we can check whether $a,b,c$ are coprime, $a \leq N$, and $a \geq b$, $a \in \mathbb{N}$ (to avoid isomorphic duplicates). A similar approach is presented in [18].

In this context the maximum diameter $n$ has to be limited to a few millions so that we can easily determine the prime factorizations of all integers at most $n$ in a precomputation. Given this data, we can quickly determine the prime factorization of $2bc$ and loop over all divisors without any additional testing.

Next, we want to describe the set of solutions of $m^2+n^2=k$ and assume that

$$k = 2^h \cdot q_1^{i_1} \dots q_s^{i_s} \cdot p_1^{j_1} \dots p_t^{j_t},$$

where the $q_l$ are primes congruent to 3 modulo 4 and the $p_l$ are primes congruent to 1 modulo 4. If any of the $i_l$ is odd, then no integer solution of $m^2+n^2=k$ exists. Otherwise each solution can be written as $(m,n) = \lambda \cdot (\tilde{m}, \tilde{n})$, where $\lambda = 2^{\lfloor h/2 \rfloor} \cdot q_1^{i_1/2} \dots q_s^{i_s/2}$ and $\tilde{m}^2 + \tilde{n}^2 = k/\lambda^2 =: \tilde{k}$, i.e.

$$\tilde{k} = 2^{\tilde{h}} \cdot p_1^{j_1} \dots p_t^{j_t},$$

where $\tilde{h} \leq 1$. Due to $(x_1^2+x_2^2)(y_1^2+y_2^2) = (x_1y_1+x_2y_2)^2 + (x_1y_2-x_2y_1)^2$ and the unique factorization of the Gaussian integers $\mathbb{Z}[i]$, it suffices to combine the solutions of the problem, where $\tilde{k}$ is a prime power. Ignoring signs for $\tilde{k}=2$, the unique solution is given by $1^2+1^2=2$. Ignoring signs and order, there is a unique solution for $u^2+v^2=p$, if the prime $p$ is equivalent to 1 modulo 4. Again ignoring signs and order, for prime powers the set of solutions of $x^2+y^2=p^j$ is given by $x+yi = (u+vi)^l(u-vi)^{j-l}$, where $0 \leq l \leq j/2$. Thus, it remains to determine a solution of $u^2+v^2=p$, which can be done by the Hermite-Serret algorithm: First determine an integer $z$ satisfying $z^2 \equiv i \pmod p$, using that $w^{\frac{p-1}{2}} \equiv -1 \pmod p$ for each quadratic nonresidue $w$, and then apply the Euclidean algorithm on $(p, w)$ to determine $(u, v)$. See [11, 32] for the original sources and [4] for an improved algorithm. The just sketched algorithm for the generation of all Heronian triangles up to diameter $n$ runs in $O(n^{2+\varepsilon})$ time, where $\varepsilon > 0$ is arbitrary.

Using this algorithm we have exhaustively generated all primitive Heronian triangles up to diameter $6 \cdot 10^6$. They are available for download at [22]. Having the data at hand we have computed an approximate counting function, which fits best for a given type of functions. Let $count(x)$ denote the number of primitive Heronian triangles with diameter between $(x-1)\cdot 10000+1$ and $x\cdot 10000$. The best least squares fitting function of the form $c_1 + c_2 \log x + c_3 \log^2 x + c_4 x + c_5 x \log x + c_6 x \log^2 x$ is given by

$$\begin{aligned} 160436.33 + 117761.45 \log x + 3191.78 \log^2 x + 12023.76x-& \\ - 2787.79x \log x + 169.14x \log^2 x& \end{aligned}$$

and leads to a $\|\cdot\|_2$-distance of 152331 for the entire data.

We note that, besides the (implicit) $O(n^{1+\varepsilon})$ upper bound from [25], we are not aware of any non-trivial lower and upper bounds for the number of (primitive) Heronian triangles with a given diameter. As shown in [27] one may deduce lower bounds for the minimum diameter of plane integral point sets. However, current knowledge is still incomplete [33]. The number of Heronian triangles with diameter at most $n$ is in $O(n^{\frac{25}{13}+\varepsilon})$, see [18]. Counts with additional restrictions are also given in [35].

**4. Exhaustive generation of $n$-clusters up to a given diameter.** In order to determine the smallest 7-cluster, we have performed an exhaustive search for $n$-clusters up to a given diameter. For the purpose of this paper the chosen maximum diameter is $6 \cdot 10^6$. A starting point is a complete list of all Heronian triangles up to this diameter. More concretely we have chosen the exhaustive algorithm described in Section 3 to generate all primitive Heronian triangles up to diameter $6 \cdot 10^6$ and extended this list by including all rescaled versions such that the resulting diameter is at most $6 \cdot 10^6$.

The underlying basic idea to construct $n$-clusters is to combine two $(n-1)$-clusters sharing a common $(n-2)$ cluster. This way, we can benefit from the fact that the constraints can be partially checked very early. So, starting from a list of 3-clusters, i.e. Heronian triangles, we generate all 4-clusters, then all 5-clusters, then all 6-clusters, and finally all 7-clusters.

For the first combination step, i.e., $n = 4$, “sharing a common $(n-2)$-cluster” means that the two triangles to be combined must both have a side of the same length.

To save time and memory we apply the concept of orderly generation, see [31], which avoids pairwise isomorphism checks when cataloging combinatorial configurations as in our example of integral point sets or $n$-clusters. To this end a canonical form has to be defined, so that during the algorithm only canonical objects are combined. The constructed objects are accepted if and only if they are canonical too. The benefit from such an approach is that no isomorphic copies arise. For the details we refer the reader to [27] with the adaptation of considering triangles of characteristic 1 only.

As a result we have computationally verified that the smallest 7-cluster has diameter 2262000 and that there is no other 7-cluster with diameter less then $4 \cdot 10^6$. Along the way we have also exhaustively constructed all 4-, 5-, 6-, and 7-clusters with diameter at most $6 \cdot 10^6$. Those lists will be beneficial for the construction of additional 7-clusters as will be explained in the following sections.

**5. Combining lists of $n$-clusters.** In the previous section we described an algorithm to exhaustively generate a list of all $n$-clusters up to given diameter $D$. As input we take a complete list of $(n-1)$-clusters up to diameter $D$ so that initially we need a complete list of all Heronian triangles up to diameter $D$. Such an approach is computationally limited to rather small diameters, where only a few 7-clusters exist. So from now on we will leave the approach of exhaustive generation and switch to incomplete construction algorithms.

Our assumption for this section is that we are given a list of $n$-clusters, which we then combine to a list of $n'$-clusters. For our paper, the most general setting is the following: Given a list $L_1$ of $n_1$-clusters and a possibly different list $L_2$ of $n_2$-clusters we consider pairs $(l_1, l_2)$, where $l_1 \in L_1$ and $l_2 \in L_2$, to construct $n'$-clusters. Mostly we assume $n' > \max(n_1, n_2)$.

In Section 4 we have assumed that the $(n-1)$-clusters $l_1$ and $l_2$ share a common $(n-2)$-cluster. Since in the end we are only interested in lists of non-similar $n$-clusters we relax that to the requirement that $l_1$ and $l_2$ contain a common $c$-cluster, where $c$ is an additional parameter.

Having the $c$-cluster $C_1$ of $l_1$ fixed we loop over all $c$-clusters $C_2$ of $l_2$ and check whether $C_1$ and $C_2$ can be rescaled so that they coincide. This check is implemented as follows: Let $diam_1$ be the diameter of $C_1$ and $diam_2$ be the diameter of $C_2$. We define $f_1 = diam_2/gcd(diam_1, diam_2)$ and $f_2 = diam_1/gcd(diam_1, diam_2)$. With this $C_1$ and $C_2$ are similar if and only if $f_1 \cdot C_1$

is isomorphic to $f_2 \cdot C_2$. Comparing the sorted lists of the pairwise distances is a first computationally cheap test for this task. If successful, we compare the canonical forms of $C_1$ and $C_2$.

By rescaling we are in the situation that $l_1$ and $l_2$ contain a common $c$-cluster and we proceed by computing common coordinates: We apply the algorithm from Subsection 7.2 to compute coordinates for $l_1$ and $l_2$ separately.[6] By assuming that the first $c$ points of $l_1$ and $l_2$ coincide, we can obtain a common coordinate system by just scaling the numerators. We note that for $c = 2$ we have two possibilities for the join, otherwise just one. Having the coordinates at hand, we can loop over all $k$-sets of the points and check whether they satisfy the conditions of a $k$-cluster relaxing the condition of integral distances to rational distances. If all (relaxed) conditions are satisfied we store a primitive version of the corresponding, possibly scaled, $k$-cluster.

We have mostly used three instances of this general framework. The first is with the parameters $n_1 = n - 1$, $n_2 = 3$, and $c = 2$, i.e., we try to extend a given list of $(n-1)$-clusters by combining them with a list of primitive Heronian triangles along a common edge. Since we use rescaling, this combination is always possible, although an $n$-cluster might not be formed. Depending on the available computation time and the size of the list of the $(n-1)$-clusters, one may choose all known primitive Heronian triangles for the second list. We have done that to a large extent for the list of known 7-clusters but unfortunately did not locate an 8-cluster.

The second instance has the parameters $n_1 = n_2 = 6$ and $c = 3$, i.e., we combine lists of 6-clusters sharing a common triangle. The resulting point sets consist of nine points. We remark that the second method was able to discover some previously unknown 6- and 7-clusters but turned out to be rather slow. For later reference we call this method the combine-hexagons algorithm.

The third method mimics the exhaustive generation method from Section 4. Starting from $n = 4$ we set $n_1 = n_2 = n - 1$, $c = n - 2$ and increase $n$ by one in each iteration.

## 6. Triangle extensions.

The algorithms in Section 5 have to be applied iteratively in order to end up with $n$-clusters for large $n$. Now we describe an algorithm that directly approaches $n$-clusters without specifying $n$. Let $L$ be a list of primitive Heronian triangles of length $n$.

[6] If $L_2$ is large it is computationally beneficial to store a coordinate representation, given by the algorithm in Subsection 7.2, for each $l_2 \in L_2$.

**Algorithm 6.1** (Triangle extension)**.**

*for* $i$ *from* 1 *to* $n$
  $\mathcal{P} = \emptyset$
  *for* $j$ *from* $i$ *to* $n$
    *combine* $L(i)$ *with* $L(j)$ *in all possible ways*
      *compute coordinates of the fourth point* $p \notin L(i)$
      *if* $L(i) \cup p$ *is a* 4*-cluster then add* $p$ *to* $\mathcal{P}$
  *compute all pairwise distances between the points in* $\mathcal{P}$
  *loop over all* $k$*-sets* $\mathcal{K} = \{p_1, \ldots, p_k\}$ *of* $\mathcal{P}$ *such that* $L(i) \cup \mathcal{K} \backslash \{p_k\}$ *is a cluster*
    *if* $L(i) \cup \mathcal{K}$ *is a cluster then output* $L(i) \cup \mathcal{K}$

The implementation details for the coordinate and distance computations are described in Section 7.

**7. Low-level mathematical and implementation details.** In the previous sections we have described our algorithms omitting implementation details. The application of those algorithms result in many sub-computations, such as coordinate and distance computations. Those sub-routines have to be carefully designed in order to save costly unlimited precision rational computations.

**7.1. Compute rational coordinates of a Heronian triangle.** Suppose we are given three integer side lengths $a$, $b$, and $c$, which form a non-degenerate Heronian triangle. Our aim is to compute rational coordinates for the points $P_1$, $P_2$, and $P_3$ attaining those pairwise distances, i.e., $|P_1P_2| = a$, $|P_1P_3| = b$, and $|P_2P_3| = c$.

W.l.o.g. we can assume that the first point is located in the origin and the second point on the positive part of the $x$-axis, i.e., $P_1 = (0,0) = \left(\frac{0}{2a}, \frac{0}{2a}\right)$ and $P_2 = (0,a) = \left(\frac{0}{2a}, \frac{2a^2}{2a}\right)$. Setting $t_1 := b^2 - c^2 + a^2$ and $t_2 := 4b^2a^2 - (b^2 - c^2 + a^2)^2$ we have $P_3 = \left(\frac{t_1}{2a}, \pm\frac{t_2}{2a}\right)$, where we may use the solution with positive $y$-coordinate.

In some algorithms all permutations of the three edge lengths of a Heronian triangle $(a,b,c)$ have to be considered. To this end we assume that the above auxiliary integer values $t_1$ and $t_2$ have already been computed. Permuting the two latter side lengths, i.e., $(a,c,b)$, is equivalent to swaping the points $P_1$ and $P_2$.

The corresponding coordinates with non-negative $y$-values are given by

$$\left(\frac{0}{2a}, \frac{0}{2a}\right), \left(\frac{2a^2}{2a}, \frac{0}{2a}\right), \left(\frac{2a^2 - t_1}{2a}, \frac{t_2}{2a}\right).$$

By applying a suitable rotation matrix, we obtain the coordinate representation

$$\left(\frac{0}{2b}, \frac{0}{2b}\right), \left(\frac{2b^2}{2b}, \frac{0}{2b}\right), \left(\frac{2b^2 - t_1}{2b}, \frac{t_2}{2b}\right)$$

for the triangle $(b, c, a)$ and

$$\left(\frac{0}{2c}, \frac{0}{2c}\right), \left(\frac{2c^2}{2c}, \frac{0}{2c}\right), \left(\frac{2a^2 - t_1}{2c}, \frac{t_2}{2c}\right)$$

for the triangle $(c, a, b)$.

So there is no need to compute additional square roots. Of course, the common sub-expressions like $a^2$, $b^2$, and $c^2$ should be stored additionally.

**7.2. Compute rational coordinates of an *n*-cluster.** We assume a suitable but fixed ordering of the points and denote the integer distance between the first two points by $a$. According to Subsection 7.1 we set $P_1 = \left(\frac{0}{2a}, \frac{0}{2a}\right)$, $P_2 = \left(\frac{0}{2a}, \frac{2a^2}{2a}\right)$, and $P_3 = \left(\frac{t_1}{2a}, \frac{t_2}{2a}\right)$. For $4 \le i \le n$ we apply the construction of Subsection 7.1 to the triangle given by the points $P_1$, $P_2$, and $P_i$. To decide the sign of the $y$-coordinate of $P_i$ we utilize the distance to $P_3$. Thus all points have coordinates $\left(\frac{x_i}{2d}, \frac{y_i}{2d}\right)$ with integers $x_i, y_i$.

**7.3. Checking for rational distances.** Suppose we are given two points with rational coordinates $\left(\frac{x_1}{a_1}, \frac{y_1}{b_1}\right)$ and $\left(\frac{x_2}{a_2}, \frac{y_2}{b_2}\right)$. The task is to decide whether they are at rational distance and eventually compute the distance. Since during our searches most of the checked distances are irrational, it is important to have a quick check for the decision problem. An exact expression for the distance is given by

$$\frac{\sqrt{(b_1b_2)^2(a_2x_1 - a_1x_2)^2 + (a_1a_2)^2(b_2y_1 + b_1y_2)^2}}{a_1a_2b_1b_2}.$$

Thus the problem is reduced to the question whether a certain integer is a square.

Here we can benefit from modular arithmetic. Suppose that $m$ is an arbitrary integer and compute $(b_1b_2)^2(a_2x_1 - a_1x_2)^2 + (a_1a_2)^2(b_2y_1 + b_1y_2)^2 \mod m$

by performing all intermediate computations modulo $m$. If the result is not a square in $\mathbb{Z}_m$ the distance under study can not be rational. If $m$ is a product of distinct primes then we can check the square property separately for each prime $p$ by simply tabulating a boolean incidence vector for the squares in $\mathbb{Z}_p$. In our implementation we use $m_1 = 493991355 = 3 \cdot 5 \cdot 11 \cdot 13 \cdot 17 \cdot 19 \cdot 23 \cdot 31$ and $m_2 = 622368971 = 7 \cdot 29 \cdot 37 \cdot 41 \cdot 43 \cdot 47$, i.e., we perform two successive modular tests. Since computations modulo 4 are very cheap in most arbitrary precision libraries it pays off to first check whether the integer under study is equivalent to either 0 or 1 modulo 4; otherwise its square cannot be rational.

If we can assume a common denominator of the coordinates, as e.g. implied by the algorithm in Subsection 7.2, the computations can be simplified since the distance between the points $\left(\frac{x_1}{d}, \frac{y_1}{d}\right)$ and $\left(\frac{x_2}{d}, \frac{y_2}{d}\right)$ is given by $\frac{\sqrt{(x_1-x_2)^2+(y_1+y_2)^2}}{d}$.

**7.4. Canonical forms.** In order to be able to check $n$-clusters for similarity, we define a canonical form in such a way that two $n$-clusters are similar if and only if their canonical forms coincide. Given a matrix of the pairwise rational distances we first normalize by multiplying with the unique rational number such that all distances are coprime integers. Since distances are symmetric, it suffices to consider the upper right triangular submatrix without the diagonal of zeros. We append the columns of this matrix to a distance vector $v$. With this we define the canonical form to be the lexicographically maximal distance vector over all permutations of the points.

Clearly, this canonical form is unique and can be determined by comparing all $n!$ possible permutations. For our purposes this was fast enough even for $n = 7$, but we note that one can easily design an $O(n^2)$ algorithm.

## 8. Circle inversion.

As observed in [34], the rationality of distances in $\mathbb{R}^2$ is preserved by translations, rotations, scaling with rational numbers, and by some kind of circle inversion. Here we go into the details of the latter transform. Assume that our point set has a point at the origin. A circle inversion through the origin with radius one sends each point with coordinates $(x, y)$ except the origin to $\left(\frac{x}{x^2+y^2}, \frac{y}{x^2+y^2}\right)$[7].

Using this transform we can construct $(n-1)$-clusters from $n$-clusters by moving each of their points to the origin and applying the described circle in-

[7]Using complex notation this is (ignoring a reflection) equivalent to the map $z \mapsto \frac{1}{z}$.

version. Doing this for the set of all known 7-clusters gives no new 6-clusters. Strangely enough, the set of the contained subtriangles, i.e. the set of the (normalized) subtriangles from the resulting 6-clusters, coincides with the set of the subtriangles contained in the 7-clusters.

Discarding one point is, on the one hand, disadvantageous. On the other hand we obtain some freedom in the initial point set, i.e. it does not have to be an $n$-cluster. To be more precise, we need a rational point set $\mathcal{P}$ with characteristic 1, where no four points are on a line and no four points are on a circle. Circle inversion at a vertex of $\mathcal{P}$ automatically destroys collinear triples. We were able to extend some of the 7-clusters to an 8-point rational set. Unfortunately, in each of these cases the 8th point also was part of a circle containing four points of the point set. A promising configuration might be the so-called Pappus configuration consisting of nine points and nine lines, with three points per line and three lines through each point. Unfortunately we were not able to find a representation of the Pappus configuration with pairwise rational distances.

So while circle inversion might be theoretically interesting, we were not able to draw any computational advantages.

**9. Choosing promising Heronian triangles.** The algorithms presented in the previous sections can in principle deal with large lists of $n$-clusters, but of course the computation time limits such searches. In order to find many non-similar 7-clusters we have tried to restrict ourselves to promising search spaces.

Both the exhaustive-like algorithm from Section 5 and the triangle extension algorithm from Section 6 are based on a list of Heronian triangles. Unfortunately we do not have the computational capacity to run those algorithms with all Heronian triangles known to us, but have to select a subset of them. Of course, this subset should be selected in a way so that it is small but generates many 7-clusters. Satisfying the latter aim is essential but, of course, harder. To formalize this idea, we ask for a method that is able to compute a score for a given Heronian triangle, and then choose a given number of Heronian triangles with the largest scores.

A very easy but effective scoring function is the negative diameter of all Heronian triangles. In order to verify our claim we used the triangle extension algorithm with subsets of 1000 Heronian triangles. Using the first 1000 smallest Heronian triangles produces 237 6-clusters and four 7-clusters (having diameters 5 348 064, 15 772 770, 47 570 250, and 662 026 750). The second smallest 1000 Heronian triangles produces *only* nine 6-clusters and no 7-cluster.

A promising idea might be to use the number of divisors or prime divisors of the side lengths normalized by magnitude, i.e. prime side lengths should get the lowest possible score while highly composite numbers get large scores. As an example, we report the results of two explicit scoring functions based on this idea. For

$$score_1(a,b,c) := \frac{\#\text{prime divisors } a}{\log\log a} + \frac{\#\text{prime divisors } b}{\log\log b} + \frac{\#\text{prime divisors } c}{\log\log c}$$

we have chosen the 1000 Heronian triangles with maximal score among all Heronian triangles with diameter at most 10 000. Applying the triangle-extension algorithm results in three 6-clusters and no 7-cluster. The similar function

$$score_2(a,b,c) := \frac{\#\text{prime divisors } a}{\log a} + \frac{\#\text{prime divisors } b}{\log b} + \frac{\#\text{prime divisors } c}{\log c}$$

increases the number of found 6-clusters to 40 with the same setting. But of course $score_2$ tends to prefer triangles with smaller diameter. We note that using the number of divisors instead of the number of prime divisors yields similar results.

The most successful approach in our computational study was to use the known lists of $n$-clusters as selectors. To be more precise, given a list of $n$-clusters we can determine the contained sub-triangles, which then, after rescaling, gives a list of primitive Heronian triangles. If the resulting list of Heronian triangles is too large for our purposes we take the $m$ smallest ones according to their diameter or we take frequency into account, i.e. we consider only those primitive Heronian triangles which appear at least $k$ times, where $k$ is suitably chosen, as sub-triangles within the list of $n$-clusters.

As an example, we report the following experiments performed near the end of our computational study, when we already knew lots of 6- and 7-clusters. For $n = 6$ and $n = 7$ we choose the 1000 Heronian triangles having the smallest diameter, respectively. In the first case triangle extension yields 247 6-clusters and four 7-clusters. For the latter case we obtain 912 6-clusters and 100 7-clusters. So a higher initial value of $n$ results in more clusters, but of course those examples are harder to find.

A completely different idea is to associate Heronian triangles $(a, b, c)$ with ellipses represented by $\frac{a+b}{c}$. As an experiment we took the 3 000 000 smallest Heronian triangles and computed the three associated ellipses in each case. The most frequent ellipse representation occurs 10 277 times. Taking the smallest 1000 triangles results in 603 5-clusters applying the triangles extension algorithm.

Taking triangles from ellipse representations that occur exactly once result in just six 5-clusters.

We did not come to a satisfactory solution and propose the design of a good scoring function as an open problem.

**10. Computational observations.** In this section we collect some computational observations that help us to design our searches for 7-clusters.

**Observation 10.1.** *The triangle-extension algorithm is more effective than the combine-hexagons algorithm.*

Using the 412 triangles contained in the original twenty-five 7-clusters found by Simmons and Noll in 2010 as an input for the triangle-extension algorithm yields 84 non-similar 7-clusters in less than two minutes of computation time. If we instead take the sub-hexagons of the original twenty-five 7-clusters plus an additional list of 1736 hexagons and apply the combine-hexagons algorithm we end up with 33 non-similar 7-clusters. We remark that all but one of these heptagons are contained in the list of the 84 heptagons from the triangle extension algorithm. Additionally the computation time of the combine-hexagons algorithm is usually much larger than the computation time of the triangle-extension algorithm.

**Observation 10.2.** *Stripping isosceles triangles from the input set of Heronian triangles only mildly reduces the number of* 6- *and* 7-*clusters found in the search of the triangle-extension algorithm.*

Because any pair of isosceles Heronian triangles, after scaling, forms a 4-cluster, there are numerous 4-clusters formed from pairs of isosceles triangles. When three isosceles Heronian triangles are joined together along their base, then the resulting pentagon has pairwise rational distances, but three points are on a line. This situation happens when combining two such 4-clusters with a common isosceles triangle.

As expected the runtime increases while including isosceles Heronian triangles, where the precise factor strongly depends on the chosen subset of Heronian triangles. For comparison we chose the 1000 smallest non-isosceles Heronian triangles and applied the triangle-extension algorithm, which resulted in 172 6-clusters and four 7-clusters. So we have missed 65 6-clusters but no 7-cluster. Here the computation time was decreased by a factor of two. In a larger experiment we have chosen 1383799 Heronian triangles and obtained 424593 6-clusters and 1110 7-clusters. Stripping all 24583 isosceles triangles we have obtained 424543

6-clusters and 1110 7-clusters, while the computation time decreases by a factor larger than 10.

**Observation 10.3.** *Partitioning the set of triangles can speed up the search of the triangle-extension algorithm.*

Given a list of $m$ $n$-clusters containing the same $(n-1)$-cluster the ordinary combination would need $m^2$ tests. Since integral point sets with many points on a line or a circle are quite common it makes sense to take this fact into account. Partitioning 4-clusters by a line through 2 of the points or by a circle through 3 of the points avoids many spurious comparisons and speeds up the search. The important thing is that a pair of items in a partition cannot form an $n+1$-cluster because it would violate a con-circularity or co-linearity constraint. In our programs we can either turn on or off the partitioning algorithm, but mostly use it to increase the computation speed. The typical performance boost is around 10%.

**Observation 10.4.** *Large Heronian triangles tend to not form 4-clusters.*

That is, given two random small Heronian triangles, the probability they form a 4-cluster is relatively high compared to the probability that two large Heronian triangles will form a 4-cluster, i.e. we have to perform many unsuccessful combinations of Heronian triangles per found 4-cluster. To justify this theoretically, one might appeal to Ceva's theorem. As we allow the size of a Heronian triangle to increase the prime factors present in the numerators of the sines of the Heronian angles increase making it more difficult to find sets of angles where the numerators cancel each other out.

**Observation 10.5.** *Iterating the triangle-extension algorithm can find new triangles and $n$-clusters.*

As described in Section 9 combining the triangles contained in the twenty-five 7-clusters found by Simmons and Noll in 2010 yields 84 non-similar 7-clusters. Those 7-clusters contain 602 triangles which combine to 86 non-similar 7-clusters using the triangle extension algorithm. Then the iteration gets stuck since those 7-clusters contain exactly 602 non-similar triangles again.

Similarly we have used the 237 6-clusters which arose from combining the 1000 smallest Heronian triangles, see Section 9. Those 6-clusters contain 1808 non-similar triangles which can be combined to 1644 non-similar 6-clusters and 22 non-similar 7-clusters.

**Observation 10.6.** *The rational distance test rules out most of the combinations of Heronian triangles.*

To make this observation plausible we report the statistics of a large scale experiment. We chose the 3 000 000 smallest primitive Heronian triangles along with those contained in the 6-clusters known to us. Using 25 000 cores during 4.5 days $3.0 \cdot 10^{14}$ pairs of 3-clusters were tried. In 99.71% the missing sixth distance was not rational. The concircular test ruled out 10 414 450 261 possibilities (0.00%) and the collinearity test 20 129 596 307 possibilities (0.01%), while we found 835 620 202 676 (possibly similar) successful combinations (0.28%). The longest list of 4-clusters containing a common 3-cluster had length 396 442. In Table 1 we have summarized the corresponding statistics for the combinations of the resulting $k$-clusters for $3 \leq k \leq 7$.

Table 1. Failure of different checks for $k+1$-clusters combining two $k$-clusters

| $k$ | comb. | distance | concircularity | collinearity | successful | intersectable |
|---|---|---|---|---|---|---|
| 3 | $3.0 \cdot 10^{14}$ | 99.71% | 0.00% | 0.01% | 0.28% | 396442 |
| 4 | $2.1 \cdot 10^{15}$ | 41.87% | 58.13% | 0.00% | 0.00% | 91 |
| 5 | $1.6 \cdot 10^{8}$ | 49.93% | 33.17% | 14.01% | 2.89% | 16 |
| 6 | $1.5 \cdot 10^{5}$ | 60.89% | 18.93% | 8.86% | 11.32% | 2 |
| 7 | 82 | 100% | 0.00% | 0.00% | 0.00% | 0 |

**11. Computational results.** We constructed 1154 non-similar 7-clusters and 443711 non-similar 6-clusters[8]. The 5- and 4-clusters are so numerous that we did not collect them. The total number of stored Heronian triangles is 807 677 361. The smallest diameter of a primitive 7-cluster is 2262000 while the largest found primitive 7-cluster has a diameter of

$$929860180385152289136849449373130154565 \approx 10^{38}.$$

The 1154 7-clusters contain in total $\binom{7}{3} \cdot 1154 = 40390$ sub-triangles, while only 9264 of them are non-similar, i.e., on average each (normalized) triangle is used more than four times. The smallest contained triangle is $(5, 4, 3)$, which is indeed the smallest possible Heronian triangle, and the largest has diameter $121990813408205791 \approx 10^{18}$. Some counts of 7-clusters are given in Table 2. We note that the Heronian triangles $(6, 5, 5)$, $(8, 5, 5)$, and $(13, 12, 5)$ are not contained in any of the known 7-clusters. The 6-clusters contain more than 1 400 000 non-similar Heronian triangles. The smallest Heronian triangle that is not contained in one of the known 6-clusters is $(149, 148, 3)$.

[8]The list of the primitive 6- and 7-clusters currently known to us can be obtained at [22].

Table 2. Number of (known) non-similar 7-clusters up to a given diameter

| diameter | # 7-clusters | diameter | # 7-clusters | diameter | # 7-clusters |
|---|---|---|---|---|---|
| $\leq 10^{7}$ | 4 | $\leq 10^{19}$ | 688 | $\leq 10^{31}$ | 1130 |
| $\leq 10^{8}$ | 11 | $\leq 10^{20}$ | 752 | $\leq 10^{32}$ | 1137 |
| $\leq 10^{9}$ | 26 | $\leq 10^{21}$ | 819 | $\leq 10^{33}$ | 1145 |
| $\leq 10^{10}$ | 52 | $\leq 10^{22}$ | 877 | $\leq 10^{34}$ | 1147 |
| $\leq 10^{11}$ | 89 | $\leq 10^{23}$ | 927 | $\leq 10^{35}$ | 1150 |
| $\leq 10^{12}$ | 139 | $\leq 10^{24}$ | 974 | $\leq 10^{36}$ | 1153 |
| $\leq 10^{13}$ | 198 | $\leq 10^{25}$ | 1024 | $\leq 10^{37}$ | 1153 |
| $\leq 10^{14}$ | 270 | $\leq 10^{26}$ | 1050 | $\leq 10^{38}$ | 1154 |
| $\leq 10^{15}$ | 347 | $\leq 10^{27}$ | 1067 | | |
| $\leq 10^{16}$ | 431 | $\leq 10^{28}$ | 1087 | | |
| $\leq 10^{17}$ | 516 | $\leq 10^{29}$ | 1111 | | |
| $\leq 10^{18}$ | 609 | $\leq 10^{30}$ | 1124 | | |

As hardware we used 25 000 cores at Google Inc. and the Linux computing cluster of the University of Bayreuth, which consists of 201 2xIntel E5520 2.26 GHz and 52 2xIntel E5620 2.4GHz processors (100-300 jobs are done in parallel). The computations for the triangle-extension algorithm using the triangles in the known 7-clusters were done on a customary laptop computer in less than one day of computation time per iteration. We used the GNU MP Bignum library[9] and class library of numbers (CLN)[10] libraries to provide arbitrary precision integers and rationals.

Although we have invested a large amount of processing power we have not found an 8-cluster.

## 12. Conclusion.

The techniques of finding $n$-clusters have dramatically improved since the discovery of the first 6-clusters in $\mathbb{R}^2$. Before that some researchers had even incorrectly conjectured that 6-clusters in $\mathbb{R}^2$ did not exist. At the current state it is still a significant computational challenge to find new 7-clusters, but we have shown that many examples exist. A toolbox of algorithms to generate $n$-clusters is provided. Using the triangle-extension algorithm one may eventually extend a small list of $n$-clusters to a larger list of $n$-clusters by just combining their contained subtriangles. Compared with its running time and its output in terms of newly found $n$-clusters this is certainly the most effective

[9] http://gmplib.org/

[10] http://www.ginac.de/CLN/

algorithm that is currently known. For a given $n$-cluster the knowledge of only $n-2$ of its sub-triangles may suffice to recover all distances and so all $\binom{n}{3}$ sub-triangles. Moreover we have some kind of scale invariance, i.e. only the angles but not the side lengths have to be known in advance. Considering all possible scalings comes at constant cost.

However this algorithm is at the mercy of a good list of Heronian triangles, or indirectly a list of starting $n$-clusters. To some extent the algorithm itself produces some new Heronian triangles so that it can be applied iteratively. But admittedly the number of successful iterations is observed to be rather small in practice. So different algorithms are needed to populate the set of *promising* triangles. Choosing them directly from the list of Heronian triangles, based on a scoring function, still has no satisfactory solution and is left as an open problem. So still the discovery of new 7-clusters depends on extensive computer calculations so that highly optimized low level routines are essential to check a large number of cases.

Along the way we have exhaustively constructed all primitive Heronian triangles with diameter up to $6 \cdot 10^6$. This database may serve as a starting point to check various conjectures.

The question of whether there exists an infinite number of non-similar 7-clusters is still open. At this point one may of course speculate on the existence of 8-clusters in $\mathbb{R}^2$.

## REFERENCES

[1] Anderson W., W. Simons, J. G. Mauldon, J. C. Smith. Elementary Problems and Solutions: A dense subset of the unit circle. *Amer. Math. Mon.*, **86** (1979), No. 3.

[2] Anning N. H., P. Erdős. Integral distances. *Bull. Am. Math. Soc.*, **51** (1945), 598–600.

[3] Brass P., W. Moser, J. Pach. Research problems in discrete geometry. Springer, New York, 2005.

[4] Brillhart J. Note on representing a prime as a sum of two squares. *Mathematics of Computation*, **26** (1972), No. 120, 1011–1013.

[5] Carmichael R. D. Diophantine analysis. John Wiley and Sons, New York, 1915.

[6] Chisholm C., J. A. MacDougall. Rational and Heron tetrahedra. *Journal of Number Theory*, **121** (2006), No. 1, 153–185.

[7] Dickson L. E. Rational triangles and quadrilaterals. *Amer. Math. Monthly*, **28** (1921), 244–250.

[8] Erdős, P. Integral distances. *Bull. Am. Math. Soc.*, **51** (1945), 996.

[9] Fricke J. On Heron Simplices and Integer Embedding. Preprint, 2001. `http://arxiv.org/abs/math/0112239`

[10] Guy R. K. Unsolved problems in number theory. 3rd ed., Springer, 2005.

[11] Hermite C. Note au sujet de l'article précédent. *J. Math. Pures Appl.*, **13** (1848).

[12] Almering J. H. J. Rational quadrilaterals. *Indag. Math.*, **25** (1963), 192–199.

[13] Berry T. G. The Rational Distance Problem for Isosceles Triangles with one rational side. Preprint, 2013. `http://arxiv.org/abs/1301.4931`

[14] Benediktovich V. I. On rational approximation of a geometric graph. *Discrete Mathematics*, **313** (2013), No. 20, 2061–2064.

[15] Berry T. G. Points at rational distance from the vertices of a triangl. *Acta Arith.*, **62** (1992), No. 4, 391–398.

[16] Dubickas, A. On some rational triangles. *Mediterranean Journal of Mathematics*, **9** (2012), No. 1, 95–103.

[17] Geelen J., A. Guo, D. McKinnon. Straight line embeddings of cubic planar graphs with integer edge lengths. *Journal of Graph Theory*, **58** (2008), No. 3, 270–274.

[18] Ionascu E. J., F. Luca, P. Stănică. Heron triangles with two fixed sides. *Journal of Number Theory*, **126** (2007), No. 1, 52–67.

[19] Kemnitz A. Punktmengen mit ganzzahligen Abständen. Habilitationsschrift, TU Braunschweig, 1988.

[20] Kreisel T., S. Kurz. There are integral heptagons, no three points on a line, no four on a circle. *Discrete Comput. Geom.*, **39** (2008), No. 4, 786–790.

[21] Kummer E. E. Über die Vierecke, deren Seiten und Diagonalen rational sind. *J. Reine Angew. Math.*, **37** (1848), 1–20 (in German). `http://resolver.sub.uni-goettingen.de/purl?GDZPPN00214638X`

[22] Kurz S. `http://www.wm.uni-bayreuth.de/index.php?id=554&L=3`

[23] Kurz S. Konstruktion und Eigenschaften ganzzahliger Punktmengen. Ph. D. thesis, Bayreuth. Math. Schr. 76, Universität Bayreuth, 2006 (in German).

[24] Kurz S. On the characteristic of integral point sets in $\mathbb{E}^m$. *Australas. J. Comb.*, **36** (2006), 241–248.

[25] Kurz S. On the generation of Heronian triangles. *Serdica Journal of Computing*, **2** (2008), No. 2, 181–196.

[26] Kurz S., R. Laue. Bounds for the minimum diameter of integral point sets. *Australas. J. Comb.*, **39** (2007), 233–240.

[27] Kurz S., A. Wassermann. On the minimum diameter of plane integral point sets. *Ars Comb.*, **101** (2011), 265–287.

[28] Marshall S. H., A. R. Perlis. Heronian Tetrahedra Are Lattice Tetrahedra. *Am. Math. Mon.*, **120** (2013), No. 2, 140–149.

[29] Menger K. Untersuchungen über allgemeine Metrik. *Math. Ann.*, **100** (1928), 75–163 (in German).

[30] Noll L. C., D. I. Bell. $n$-clusters for $1 < n < 7$. *Math. Comput.*, **53** (1989), No. 187, 439–444.

[31] Read R. C. Every one a winner or how to avoid isomorphism search when cataloguing combinatorial configurations. *Ann. Discrete Math.*, **2** (1978), 107–120.

[32] Serret J.-A. Sur un théorème relatif aux nombres entiers. *J. Math. Pures Appl.*, **27** (1848), 12–14.

[33] Solymosi J. Note on integral distances. *Discrete Comput. Geom.*, **30** (2003), No. 2, 337–342.

[34] Solymosi J., F. de Zeeuw. On a question of Erdös and Ulam. *Discrete Comput. Geom.*, **43** (2010), No. 2, 393–401.

[35] Stănică P., S. Sarkar, S. S. Gupta, S. Maitra, N. Kar. Counting Heron triangles with constraints. *INTEGERS*, **13** (2013), No. A3, 1–13.

[36] Ulam, S. M. A collection of mathematical problems. (Interscience Tracts in Pure and Applied Mathematics. No. 8), Interscience Publishers, New York and London, 1960.

[37] Yiu P. Heronian triangles are lattice triangles. *Am. Math. Mon.*, **108** (2001), No. 3, 261–263.

*Sascha Kurz*
*Department of Mathematics,*
*Physics and Informatics*
*University of Bayreuth*
*Bayreuth, Germany*
*e-mail:* `sascha.kurz@uni-bayreuth.de`

*Landon Curt Noll*
*Cisco Systems*
*San Jose California, USA*
*e-mail:* `ncluster-mail@asthe.com`

*Randall Rathbun*
*Green Energy Technologies, LLC*
*Manning, Oregon, USA*
*e-mail:* `randallrathbun@gmail.com`

*Chuck Simmons*
*Google, Mountain View*
*California, USA*
*e-mail:* `csimmons@google.com`

*Received June 23, 2014*
*Final Accepted October 29, 2014*