SOLVING RAMANUJAN'S SQUARE EQUATION COMPUTATIONALLY

CURTIS BRIGHT

Srinivasa Ramanujan asked [7] in 1913 if the Diophantine equation

$$x^2 + 7 = 2^n \tag{1}$$

had any positive solutions (x, n) other than $(1, 3)$, $(3, 4)$, $(5, 5)$, $(11, 7)$ and $(181, 15)$. It was first proved by Tryve Nagell [5] in 1948 that these are in fact the only solutions; see [6] for an English translation. Accordingly, (1) is often referred to as the Ramanujan-Nagell equation. A summary of its history and related problems is provided by Edward Cohen [1].

The purpose of this article is to show how the equation may be solved using simple congruence techniques with the benefit of a computer. The principle underlying theory required is in the solving of the equation $x^2 - Dy^2 = N$. The method is similar to one presented by Maurice Mignotte [2] although he does not apply it to (1) and uses a another method [3] in its resolution.

The case where n is even is easily solved, since writing $n = 2k$ leads to the difference of squares

$$(x + 2^k)(x - 2^k) = -7.$$

Examining the divisors of -7 we conclude that $x + 2^k = 7$ and $x - 2^k = -1$, i.e., $x = 3$ and $2^k = 4$, which yields the only solution with n even, $(x, n) = (3, 4)$.

The case where n is odd requires more careful analysis. Writing $n = 2k + 1$ and making the substitution $y = 2^k$ leads to the equation

$$x^2 - 2y^2 = -7, \tag{2}$$

so we would like to find all solutions (x, y) to (2) such that y is a power of 2.

The set of solutions (x, y) to equations of the form

$$x^2 - Dy^2 = N \tag{3}$$

(where $D > 0$ is not a square) have a well-known structure. These equations are generalizations of the so-called Pell equation

$$x^2 - Dy^2 = 1, \tag{4}$$

which in fact plays an important role in solving the generalized case. Note that if $(\tilde{x}, \tilde{y})$ is a solution of (4) and (x, y) is a solution to (3) then $(x\tilde{x} + y\tilde{y}D, x\tilde{y} + y\tilde{x})$ is also a solution to (3). Using this fact, we may partition solutions to (3) into equivalence classes: we say that $(x, y) \sim (x', y')$ if there is some solution $(\tilde{x}, \tilde{y})$ to (4) such that $(x', y') = (x\tilde{x} + y\tilde{y}D, x\tilde{y} + y\tilde{x})$. It may be shown [4] that an equivalent condition is if $xx' \equiv yy'D \pmod{|N|}$ and $xy' \equiv x'y \pmod{|N|}$. Thus the pigeonhole principle gives a (generally weak) upper bound of N^2 classes of solutions to (3), since if two solutions are congruent modulo N then they belong to the same class. In particular, we have that every solution to (4) belongs to the same class.

Define the *minimal positive solution* of a class of solutions to be the unique solution (x, y) with the smallest $x, y > 0$. All solutions to (4) may be generated from its minimal positive solution, so to determine all solutions to (3) we need only find the minimal positive solution to (4) and a single solution from each class of (3). This is exposited in the following theorem, which is noted in [8].

Theorem 1. *Let (x, y) be a solution of $x^2 - Dy^2 = N$ and $(\tilde{x}, \tilde{y})$ be the minimal positive solution of $x^2 - Dy^2 = 1$. Define the pair of linear recurrence relations:*

$$\begin{aligned} X_i &= 2\tilde{x}\, X_{i-1} - X_{i-2} \\ Y_i &= 2\tilde{x}\, Y_{i-1} - Y_{i-2} \end{aligned} \tag{5}$$

with initial conditions $(X_0, Y_0) = (x, y)$ and $(X_1, Y_1) = (x\tilde{x} + y\tilde{y}D, x\tilde{y} + y\tilde{x})$. Then all solutions to $x^2 - Dy^2 = N$ in the class of (x, y) are given by $\pm(X_i, Y_i)$ for $i \in \mathbb{Z}$.

Note that (X_i, Y_i) is well-defined for $i < 0$ since rearranging (5) yields

$$\begin{aligned} X_i &= 2\tilde{x}\, X_{i+1} - X_{i+2} \\ Y_i &= 2\tilde{x}\, Y_{i+1} - Y_{i+2}. \end{aligned} \tag{6}$$

Define the *fundamental solution* of a class of solutions to be the solution (x, y) with the smallest $y \geq 0$, along with $x \geq 0$ if $(x, y) \sim (-x, y)$. We will be able to use Theorem 1 if we can compute all fundamental solutions of (3) and the minimal positive solution of (4); methods for doing this are described in [4, 8] and code for Maple implementations is included at the end of this article. The minimal positive solution of (4) may be computed by the "PQa" algorithm; this method uses the convergents to the continued fraction expansion of $\sqrt{D}$. The fundamental solutions of (3) may often be computed by a brute-force search since general bounds on these solutions are known; the following were specifically stated in [8].

Theorem 2. *Let (x, y) be a fundamental solution of $x^2 - Dy^2 = N$ and $(\tilde{x}, \tilde{y})$ be the minimal positive solution of $x^2 - Dy^2 = 1$. Then*

$$\begin{aligned} 0 \leq y &\leq \sqrt{\frac{N(\tilde{x}-1)}{2D}} && \text{if } N > 0; \\ \sqrt{\frac{|N|}{D}} \leq y &\leq \sqrt{\frac{|N|(\tilde{x}+1)}{2D}} && \text{if } N < 0. \end{aligned}$$

Armed with these theorems, we can now find all solutions to (2), i.e., (3) with $D = 2$, $N = -7$. We calculate that the minimal positive solution to $x^2 - 2y^2 = 1$ is $(3, 2)$ and that the fundamental solutions to $x^2 - 2y^2 = -7$ are $(x, y) = (1, 2)$ and $(u, v) = (-1, 2)$. Using Theorem 1 we can construct the sequence of solutions (X_i, Y_i) and (U_i, V_i). Table 1 shows the small solutions; all solutions to $x^2 - 2y^2 = -7$ are given by $\pm(X_i, Y_i)$ and $\pm(U_i, V_i)$ for $i \in \mathbb{Z}$.

Note that $(X_i, Y_i) = (-U_{-i}, V_{-i})$, and since we only want to find solutions (x, y) to (2) where y is a power of 2, it sufficies to just find when Y_i is a power of 2. Examining Table 1, we see that $Y_i = 2^k$ for $i \in \{-3, -1, 0, 1\}$, with $k \in \{7, 2, 1, 3\}$, leading to the remaining solutions (x, n) of (1): $(181, 15)$, $(5, 5)$, $(1, 3)$ and $(11, 7)$.

Next, we will show that these are in fact the only instances when Y_i is a power of 2, and thus completely solve (1). We do this by examining the following sequences:

$$\begin{aligned} z_1(m) &= \{2^i \bmod m\}_{i=0}^{\infty} \\ z_2(m) &= \{Y_i \bmod m\}_{i=0}^{\infty} \end{aligned}$$

i	X_i	Y_i	U_i	V_i
-5	-6149	4348	-12875	9104
-4	-1055	746	-2209	1562
-3	-181	128	-379	268
-2	-31	22	-65	46
-1	-5	4	-11	8
0	1	2	-1	2
1	11	8	5	4
2	65	46	31	22
3	379	268	181	128
4	2209	1562	1055	746
5	12875	9104	6149	4348

TABLE 1. Small solutions to $x^2 - 2y^2 = -7$.

for some suitable m. It is clear from their definition that both $z_1(m)$ and $z_2(m)$ are periodic for all m. Given some m, define λ_i to be the *period* of $z_i(m)$ and μ_i to be the *pre-period* of $z_i(m)$. Note that $\mu_2 = 0$ since the periodic portion of $z_2(m)$ will extend backwards by (6). If we can show that

$$\left\{2^i \bmod m\right\}_{i=\mu_1}^{\mu_1+\lambda_1-1} \cap \left\{Y_i \bmod m\right\}_{i=0}^{\lambda_2-1} = \emptyset \tag{7}$$

then $Y_i \neq 2^k$ for all $i \in \mathbb{Z}$ unless $k < \mu_1$.

Now all that remains is to find an m which satisfies (7); this is best accomplished by a computer search. Although I will not go into detail here, rather than checking each $m > 1$ individually there are conditions which simplify the search considerably. In our case, with $m = 1966336 = 2^8 \cdot 7681$ we find that

$$\mu_1 = 8, \quad \lambda_1 = 3840, \qquad\qquad \mu_2 = 0, \quad \lambda_2 = 256.$$

There are 3840 residues in $\{2^i \bmod m\}_{i=8}^{3847}$ and 256 residues in $\{Y_i \bmod m\}_{i=0}^{255}$, but (7) is satisfied! Since we have already noted all k such that $Y_i = 2^k$ for $k < 8$, we have proved that no other solutions to Ramanujan's square equation exist.

As a final remark, we note that alterative possibilities for m include $16777472 = 2^8 \cdot 65537$ and $25167872 = 2^{11} \cdot 12289$.

REFERENCES

[1] E. Cohen, On the Ramanujan-Nagell Equation and Its Generalizations, *Number Theory: Proceedings of the First Conference of the Canadian Number Theory Association* (1990), 81–92.
[2] M. Mignotte, On the Automatic Resolution of Certain Diophantine Equations, *EUROSAM 84 Proceedings*, *Lecture Notes In Computer Science* **174** (1984), 378–385.
[3] M. Mignotte, Une nouvelle résolution de l'équation $x^2 + 7 = 2^n$, *Rendiconti del Seminario della Facoltà di Scienze dell'Università di Cagliari* **54** (1984), 41–43.
[4] R. Mollin, *Fundamental Number Theory with Applications* (1998), 249, 298–301, 338–339.
[5] T. Nagell, Løsning til oppgave nr 2, 1943, s. 29, *Norsk Mathematisk Tidsskrift* **30** (1948), 62–64.
[6] T. Nagell, The Diophantine Equation $x^2 + 7 = 2^n$, *Arkiv för Matematik* **4** (1961), 185–187.
[7] S. Ramanujan, Question 464, *Journal of the Indian Mathematical Society* **5** (1913), 120.
[8] J. Robertson, Solving the generalized Pell equation $x^2 - Dy^2 = N$, online article (2004), `http://hometown.aol.com/jpr2718/pell.pdf`.
[9] E. Weisstein, Ramanujan's Square Equation, from *MathWorld*–A Wolfram Web Resource, `http://mathworld.wolfram.com/RamanujansSquareEquation.html`.

Maple Code 1 Returns the minimal positive solution (x, y) to the Pell equation $x^2 - Dy^2 = 1$ (where $D > 0$ is not a perfect square) using the PQa algorithm.

```
pellsolve := proc(D::posint)
  local P, Q, a, A, B, i;
  if type(sqrt(D), integer) then
    error("D must be a nonsquare integer");
  end if;
  P := 0;
  Q := 1;
  a := floor(sqrt(D));
  A := 1, a;
  B := 0, 1;
  for i from 1 do
    P := a*Q - P;
    Q := (D - P^2)/Q;
    a := floor((P+sqrt(D))/Q);
    A := A[2], a*A[2]+A[1];
    B := B[2], a*B[2]+B[1];
    if Q = 1 and i mod 2 = 0 then
      break;
    end if;
  end do;
  return A[1], B[1];
end;
```

Maple Code 2 Returns a set containing all fundmental solutions (x, y) to the generalized Pell equation $x^2 - Dy^2 = N$ (where $D > 0$ is not a perfect square) using brute-force search between bounds on y.

```
genpellsolve := proc(D::posint, N::integer)
  local t, u, L1, L2, sols, x, y;
  if type(sqrt(D), integer) then
    error("D must be a nonsquare integer");
  end if;
  t, u := pellsolve(D);
  if N > 0 then
    L1 := 0;
    L2 := floor(sqrt(N*(t-1)/(2*D)));
  elif N < 0 then
    L1 := ceil(sqrt(-N/D));
    L2 := floor(sqrt((-N)*(t+1)/(2*D)));
  else
    return {[0, 0]};
  end if;
  sols := {};
  for y from L1 to L2 do
    x := sqrt(N+D*y^2);
    if type(x, integer) then
      sols := sols union {[x, y]};
      if x^2+y^2*D mod N <> 0 or 2*x*y mod N <> 0 then
        sols := sols union {[-x, y]};
      end if;
    end if;
  end do;
  return sols;
end;
```