The Search for Maximal Values of \( \frac{\min(A, B, C)}{\gcd(A, B, C)} \) for \( A^x + B^y = C^z \)

Arthur R. Townsend
mail@arthurtownsend.com

April 6, 2010

Abstract
This paper answers a question asked by Ed Pegg Jr. in 2001: “What is the maximal value of \( \frac{\min(A, B, C)}{\gcd(A, B, C)} \) for \( A^x + B^y = C^z \) with \( A, B, C \geq 1; x, y, z \geq 3 \)?” Equations of this form are analyzed, showing how they map to exponential Diophantine equations with coprime bases. A search algorithm is provided to find the largest \( \frac{\min}{\gcd} \) value within a given equation range. The algorithm pre-calculates a multi-gigabyte lookup table of power residue information that is used to eliminate over 99% of inputs with a single array lookup and without any further calculations. On inputs that pass this test, the algorithm then performs further power residue tests, avoiding modular powering by using lookups into precalculated tables, and avoiding division by using multiplicative inverses. This algorithm is used to show the largest \( \frac{\min}{\gcd} \) value for all equations with \( C^z \leq 2^{100} \).

The Tijdeman-Zagier Conjecture (T-Z Conjecture) [1, 2, 3, 4], also known as the Beal Prize Problem[5], is the claim that the following equation has no solutions:

Tijdeman-Zagier Conjecture Equation / Beal Prize Problem Equation
\[ A^x + B^y = C^z \]  \( x, y, z \in \mathbb{Z} \geq 3 \)  \( A, B, C \in \mathbb{Z}^+ \)  \( \gcd(A, B, C) = 1 \)

Regarding this conjecture, in 2001, Ed Pegg Jr. asked the following question [6]: Without the restriction that \( A, B, C \) be coprime, what is the maximal value of \( \frac{\min(A, B, C)}{\gcd(A, B, C)} \)? We define the Pegg Value to be \( \frac{\min(A, B, C)}{\gcd(A, B, C)} \) of a T-Z Conjecture Equation or Resultant Pegg Equation.

Resultant Pegg Equation
\[ A^x + B^y = C^z \]  \( x, y, z \in \mathbb{Z} \geq 3 \)  \( A, B, C \in \mathbb{Z}^+ \)  \( \gcd(A, B, C) > 1 \)

Notice that in the T-Z Conjecture, the \( \gcd(A, B, C) = 1 \) condition avoids an infinite number of trivial and uninteresting solutions. For example, adding any two integers \( A \) and \( B \) gives an equation, \( A + B = C \) which multiplied by \( A^{20}B^{15}C^{24} \) yields \( (A^1B^1C^8)^3 + (A^5B^3C^6)^4 = (A^1B^3C^5)^9 \). The Pegg Value for this construction is 1, even if the original terms are all coprime. In contrast to this construction, any T-Z Conjecture counterexample will have a Pegg Value equal to \( \min(A, B, C) \). And \( \min(A, B, C) \) will be > 1, as Preda Mihăilescu showed[7] that the only perfect powers that differ by 1 are \( 2^3 \) and \( 3^2 \). So in some sense, equations with a Pegg Value > 1 are “closer” to being T-Z Conjecture counterexamples. Let us examine precisely what types of equations have a Pegg Value > 1 that are not T-Z Conjecture counterexamples.

1 Original Equations and Conversion to Resultant Equations

Starting with a Resultant Pegg Equation with a Pegg Value > 1, then expanding and dividing out the common factors (so the terms to be added are coprime), one is left with: \( da^e + eb^f = fe^g \); \( x, y, z \geq 3 \); \( a, b, c, d, e, f \geq 1 \); \( \gcd(da, eb, fe) = 1 \); and
there exists a number $N$ such that $Nd$ is an $x$-th power, $Ne$ is a $y$-th power, and $Nf$ is a $z$-th power. If the smallest $N = 1$, then the resultant Pegg Equation is simply a T-Z Conjecture Counterexample that has been multiplied by a number that is simultaneously an $x$-th power, $y$-th power, and $z$-th power. Otherwise at least one of $d, e, f > 1$, and we call the equation an Original Pegg Equation. Multiplying this original equation $da^x + eb^y = fc^z$ by $N$ yields its resultant equation: $A^x + B^y = C^z$ with $A = (Nd)^{1/x} a; B = (Ne)^{1/y} b; C = (Nf)^{1/z} c$.

For ease of reference throughout this paper, $a, b, c$ will refer to the bases to the $x, y, z$ exponents, respectively, of the original equation. And $A, B, C$ will refer to the bases to the $x, y, z$ exponents, respectively, of the resultant equation. Similarly, $d, e, f$ refer to the original coefficients associated with the $a, b, c$ bases, respectively. And $D, E, F$ will refer to the resultant coefficients associated with the resultant $A, B, C$ bases, respectively. $D, E, F$ are the portions of the resultant bases that result from “spreading” the original coefficients across the equation to make each term a perfect power: $D = (Nd)^{1/x} = A/a; E = (Ne)^{1/y} = B/b; F = (Nf)^{1/z} = C/c$. So the original equation $da^x + eb^y = fc^z$ after multiplying by $N$, converts to $(Da)^x + (Eb)^y = (Fc)^z$ or equivalently $A^x + B^y = C^z$. If $a, b, c, d, e, f$ are all pairwise coprime we see that the Pegg Value cannot be less than $\min(a, b, c)$, because one of $a, b, c$ will be a factor of $\min(A, B, C)$ and $gcd(A, B, C)$ will not contain any factors of $a, b, c$.

Most of our original equations needing conversion to resultant form will only have a single coefficient $> 1$ and the other two coefficients will $= 1$. So for ease of reference, when only a single coefficient $> 1$, that coefficient will be mapped to coefficient $f$ (itself associated with base $c$ to the $z$-th power) and coefficients $d$ and $e$ will $= 1$. Each of $a, b, c$ will always have a coefficient $\geq 1$, so when a coefficient of an original equation is referred to as the coefficient, it specifically means the sole coefficient $> 1$.

Regarding the multiplier $N$, consider the case that has only a single coefficient: $\pm a^x \pm b^y \pm f c^z = 0$. Using the symmetrical form of the equation allows us to avoid repeating some arguments and also allows us to consider only positive integers throughout this paper. In order to convert this original equation into its resultant form, the equation must be multiplied by a number $N$ such that: $N$ is an $x$-th power, $N$ is a $y$-th power and $Nf$ is a $z$-th power. One method of generating such an $N$ is let $N = f^n$ such that $f^y = N$ is an $x$-th power, $f^y$ is a $y$-th power, and $f^{y+1}$ is a $z$-th power. We see that $q \equiv 0 \pmod{x}; q \equiv 0 \pmod{y}; q + 1 \equiv 0 \pmod{z}$. The Chinese Remainder Theorem (CRT) guarantees a solution for $q$ with this system of congruences when $gcd(x, y, z) = 1$. As an example, consider the original equation $\pm a^x \pm b^y \pm f c^z = 0$ with $x = y = z = 3$. Let $N = f^9$. Solving the congruences $q \equiv 0 \pmod{x}; q \equiv 0 \pmod{y}; q \equiv -1 \pmod{z}$ per the CRT gives $q = 0$, and multiplying the equation by $N = f^9$ yields a conversion from the original to the resultant form: $(f a^3)^x + (f b^3)^y = (f c^3)^z$.

However, an $N$ produced in this fashion is not necessarily the smallest number possible that meets those conditions. Consider the situation when the coefficient happens to be a perfect square (such as: $1369 \times 39^5 + 22505^3 = 225863^2$) with $g^2 = f$, $\pm a^x \pm b^y \pm g^2 c^z = 0$. Instead of multiplying by the entire coefficient to the ninth power $(g^2)^9$, we can instead multiply by $g^3$, giving $\pm (ga)^3 \pm (gb)^3 \pm (gc)^3 = 0$. This multiplier reduction applies not only when a coefficient is a perfect power, but also to any perfect power that divides the coefficient. To produce the smallest $N$ such that $N$ is a perfect $x$-th power, $N$ is a perfect $y$-th power, and $Nf$ is a perfect $z$-th power, we need to consider the highest power of each distinct prime factor of $f$.

Let $p > 1$ be a prime number. For an integer $n \geq 1$, the $p$-adic valuation $v_p(n)$ is defined to be the largest integer $r \geq 0$ such that $p^r \mid n$. Consider an equation with $d \geq 1; e \geq 1; f > 1$. To convert from its original form to the resultant form, for every distinct prime $p \mid f$, the entire equation must be multiplied by number $p^t$ such that $q \equiv -v_p(d) \pmod{x}; q \equiv -v_p(e) \pmod{y}; q \equiv -v_p(f) \pmod{z}$. But $d, e, f$ are pairwise coprime, so any prime dividing $f$ will not divide $d$ or $e$, so the congruence reduces to: $q \equiv 0 \pmod{x}; q \equiv 0 \pmod{y}; q \equiv -v_p(f) \pmod{z}$. When $gcd(x, y, z) = 1$, the
Table 1: Minimum prime power multiplier for each prime dividing the coefficient. Given the exponent set \{x, y, z\} shown in column 1, for the original equation \( \pm da^x \pm eb^y \pm fc^z = 0 \), for each \( p \) dividing the coefficient shown in column 2, this table shows the minimum \( q_p \) for \( p^q \) that the equation needs to be multiplied by to convert the original equation into a resultant equation.

| Exponent set | Coeff associated with base to this exponent | \( v_p \) (coefficient) | 1 | 2 | 3 | 4 |
|--------------|--------------------------------------------|--------------------------|---|---|---|---|
| {4,4,3}     | 3                                         | 8                        | 4 |   |   |   |
| {5,5,3}     | 3                                         | 5                        | 10|   |   |   |
| {3,3,4}     | 4                                         | 3                        | 6 | 9 |   |   |
| {5,5,4}     | 4                                         | 15                       | 10| 5 |   |   |
| {3,3,5}     | 5                                         | 9                        | 3 | 12|   | 6 |
| {4,4,5}     | 5                                         | 4                        | 8 | 12| 16|   |
| {3,4,5}     | 3                                         | 20                       | 40|   |   |   |
| {3,3,5}     | 4                                         | 15                       | 30| 45|   |   |
| {3,4,5}     | 5                                         | 24                       | 48| 12| 36|   |

CRT guarantees a minimal unique solution. When \( \gcd(xy, z) \neq 1 \), and \( \gcd(d, e, f) = 1 \), the CRT guarantees that no solution exists.

To complete the conversion of the equation from its original form to the resultant form, the coefficients \( d \) and \( e \) will also need to be “spread” across the resultant equation in a similar manner in order to make \( A, B, C \) perfect powers.

To convert an equation from its original form \( da^x + eb^y = fc^z \) to its resultant form, the entire equation must be multiplied by a number \( N \), the smallest of which is:

\[
N = \prod_{p|d} p^{\text{smallest}_{qd}(p)} \prod_{p|e} p^{\text{smallest}_{qe}(p)} \prod_{p|f} p^{\text{smallest}_{qf}(p)}
\]

with the functions:

- \( \text{smallest}_{qd}(p) = \text{smallest } q \text{ where } q \equiv 0 \pmod{\text{lcm}(y, z)}, q \equiv -v_p(d) \pmod{x} \)
- \( \text{smallest}_{qe}(p) = \text{smallest } q \text{ where } q \equiv 0 \pmod{\text{lcm}(x, z)}, q \equiv -v_p(e) \pmod{y} \)
- \( \text{smallest}_{qf}(p) = \text{smallest } q \text{ where } q \equiv 0 \pmod{\text{lcm}(x, y)}, q \equiv -v_p(f) \pmod{z} \)

If these systems of congruences each have a solution, the CRT can guarantee a minimum product \( N \). Table 1 shows the minimum multiplier information obtained by the CRT for all exponent sets with each exponent \( \leq 5 \).

### 1.1 Non-coprime Exponents

The CRT guarantees that an \( N \) exists to convert a given original equation when the moduli are pairwise coprime (in our case, when the exponents are pairwise coprime). But when exponents are not pairwise coprime, \( N \) can only exist when the coefficients of the bases of the non-pairwise coprime exponents have factors in common. But being that \( \gcd(d, e, f) = 1 \), we know that:

- If an original equation has two or more bases with pairwise non-coprime exponents, the equation can only be converted into a resultant form when the coefficients of each of those bases = 1.
- Correspondingly, the original equation cannot be converted if all three exponents have a factor in common (because then all coefficients = 1, and we know that at least one of \( d, e, f \) must be > 1).
1.2 $x$-th, $y$-th, $z$-th Power Free Coefficients

An integer $n$ is said to be $k$-free ($k \geq 2$) if for every prime $p$ the $p$-adic valuation $v_p(n)$ \(< k$ (that is, $p^k \nmid n$). We can specify that original equation coefficients are minimized and their associated bases are maximized such that: $d$ is $x$-th power free, $e$ is $y$-th power free, $f$ is $z$-th power free. Having an equation with $v_p(d) \geq x$, $v_p(e) \geq y$, or $v_p(f) \geq z$ and converting it to a resultant equation yields a final equation that is identical to that produced by first moving powers from the coefficients to their associated bases such that $v_p(d) < x$, $v_p(e) < y$, $v_p(f) < z$, then multiplying that equation by a number that is a perfect $x$-th power, perfect $y$-th power, and perfect $z$-th power. Consider this example: Exponent set $\{x, y, z\}$ with gcd($x, z$) = 1; $d = e = 1$; and coefficient $f$ is the product of non-powered prime: $p_1$, and contains a $z$-th powered prime: $p_2$. As part of the process of converting this original equation into its resultant form, the entire equation must be multiplied by $p_2^z$ such that $p_2^z$ is an $x$-th power and $p_2^{z+x}$ is a $z$-th power. So: $q \equiv 0 \pmod{x}$; $q \equiv -z \pmod{z}$. This is equivalent to looking for: $q \equiv 0 \pmod{x}$; $q \equiv 0 \pmod{z}$. With $x$ and $z$ coprime, the smallest $q$ is $xz$. With the allowance that the entire resultant equation can always be further multiplied by a number, there is no reason to allow coefficients to have a $v_p$($\text{coefficient}$) $\geq$ the exponent of the associated base.

Putting all this information together, in order for an Original Pegg Equation to be converted to a Resultant Pegg Equation with a Pegg Value $> 1$, it must be of the form:

\[
d a^x + e b^y = f c^z \quad \text{where } x, y, z \geq 3 \quad \text{and at least one of } d, e, f > 1 \quad \gcd(x, y, z) = 1
\]

$d$ is $x$-th power free; $e$ is $y$-th power free; $f$ is $z$-th power free; and any two bases that have non coprime exponents must each have an associated coefficient $= 1$.

Continuing, we can also determine the types of original equations that could be converted to resultant equations with a minimum desired Pegg Value.

1.3 Smallest Possible $\gcd(A, B, C)$

$\gcd(A, B, C)$ can be no smaller than the resultant coefficient of the base(s) to the highest exponent. Consider an equation with a coefficient $f$ associated with the base to the $z$-th power. For every prime $p \mid f$, the equation must be multiplied by $p^d$ such that $p^d$ is an $x$-th power, $p^e$ is an $y$-th power, and $p^{d+y+f}$ is a $z$-th power. The $p^d$ multiplication will present itself as part of $D$ (the resultant coefficient as part of the $A$ base) as $p^{d+y}$, of $E$ (the resultant coefficient as part of the $B$ base) as $p^{d+y}$, and of $F$ (the resultant coefficient as part of the $C$ base) as $p^{(d+y)+f}/z$. Per [3,4] with $f > 1$, we know that $\gcd(x, x) = 1$, and therefore $z \neq x$. Consider the two possibilities:

- With $z < x$: Because $z < x$, $q/z$ always $> q/x$, so when $q/x$ is an integer, $q/z + v_p(f)/z$ must be a greater integer.
- With $z > x$: Because $z > x$, $q/z$ always $< q/x$, so when $q/x$ is an integer, $q/z$ must be less than that integer. The sum $q/z + v_p(f)/z$ may $= q/x$, but it cannot $> q/x$, because in order for the sum to equal a higher integer, $v_p(f)/z$ must be $> 1$, and that is impossible (as $f$ is $z$-th power free, $v_p(f) < z$, and $v_p(f)/z < 1$).

The above logic applies to all combinations of bases and exponent sets and primes dividing each coefficient. An original equation must be one of the following exponent sets:

- a single base has the highest exponent, and the base to the highest exponent must have a coefficient (such as \{3,3,4\}).
- a single base has the highest exponent, and it may or may not have an associated coefficient (such as \{3,4,5\}).
• two bases have the highest exponent, and each of those bases are precluded from having a coefficient (such as \{5,5,3\}). Regardless of the situation, the resultant coefficient of the base(s) associated with the highest power will divide the resultant coefficients of the remaining base(s).

1.4 Re-associating $\min(A, B, C)$

If $\min(A, B, C)$ is not associated with a base to the highest exponent, the entire equation can be multiplied by a number that will make $\min(A, B, C)$ associated with a base to the highest exponent. For example, consider the example Original Pegg Equation $5^3 + 427^3 = 60073 \cdot 6^4$.

It converts to the smallest resultant equation $(60073 \cdot 5)^3 + (60073 \cdot 427)^3 = (60073 \cdot 6)^4$ which does not have $\min(A, B, C)$ associated with the base to the highest power. This equation has a Pegg Value $a = 5$. Further multiplying the resultant equation by $N = 2^{\gcd(x,y,z)} = 2^{12}$ yields $(2^{12} \cdot 60073 \cdot 5)^3 + (2^{12} \cdot 60073 \cdot 427)^3 = (2^{12} \cdot 60073 \cdot 6)^4$ which moves $\min(A, B, C)$ to $C$, but the Pegg Value $< c$ as the gcd “steals” a factor of 2 from the $c$ base, resulting in a Pegg Value $= 3$. So we do want to multiply by a number that will not “steal” a factor from the original base. If the resultant equation was instead multiplied by $N = 5^{12}$, the final equation would be $(5^{12} \cdot 60073 \cdot 5)^3 + (5^{12} \cdot 60073 \cdot 427)^3 = (5^{12} \cdot 60073 \cdot 6)^4$ and the Pegg Value $= c = 6$ (at the cost of a much larger resultant equation).

1.5 Reduced Pegg Value When Converting from Original to Resultant Form

This “stealing” can also result when the equation is multiplied by the $N$ to convert from its original form to resultant form. It can occur any time the $p^a$ as part of the resultant coefficient of the base to the highest power is of a smaller power than the $p^b$ as part of the resultant coefficient of both the other bases. As an example, Table 2 shows that this can occur with the \{3,3,5\} exponent set with $\nu_p(f)$ either 1 or 3. Consider the original \{3,3,5\} equation: $(5 \cdot 23)^3 + (2^7)^3 = f(3)^3$, with $f = 3 \cdot 7 \cdot 709$. After converting it to its resultant form: $(f^3 \cdot 5 \cdot 23)^3 + (f^3 \cdot 2^7)^3 = (f^2 \cdot 3)^5$ the resulting $\min(A, B, C) = C$, but instead of $\gcd(A, B, C)$ being $f^2$, the gcd is $f^2 \cdot 3$, because it “steals” a factor of 3 from $c$. So the Pegg Value of the resultant equation is 1, less than the smallest original base to the highest exponent. And there is no way around this “stealing” during the original to resultant equation conversion, because we must multiply the equation by the primes that divide the coefficient.

1.6 Minimum Original Equation Bases for a Given Pegg Value

Per [13], $\gcd(A, B, C)$ can be no smaller than the resultant coefficient associated with the base(s) to the highest power. And per [14] $\min(A, B, C)$ can always be associated with the smallest base to the highest power. Therefore the smallest base criteria for an original equation to convert to a resultant equation of a given Pegg Value $V$ is only that each of the base(s) to the highest exponent must be $\geq V$. Under the condition that the equation not be multiplied by a number to re-associate $\min(A, B, C)$ to a different base, $\min(A, B, C)/\gcd(A, B, C)$ must immediately be $\geq V$. Consider the original \{3,4,5\} equation with $x = 3, y = 4, z = 5$ and a coefficient of 2 associated with the base to the third power: $\pm 2a^3 \pm b^4 \pm c^5 = 0$. It converts to its resultant form: $\pm (2^2a)^3 \pm (2^b)^4 \pm (2^5c)^5 = 0$. For this equation to be able to be converted to a
Table 2: Power of prime in resultant coefficient after conversion from original equation to resultant equation. Given the exponent set \{x, y, z\} shown in column 1, for the original equation \pm da^x \pm eb^y \pm fc^z = 0, for each \( p \) dividing the coefficient shown in column 2, this table shows how the multiplier \( p^q \) (with \( q \) determined from Table 1) is represented in the resultant coefficients using the format \( \log_p(D), \log_p(E), \log_p(F) \).

| Exponent set | Coeff associated with base to this exponent | \( v_p(\text{coefficient}) \) |
|--------------|-------------------------------------------|-------------------|
| \{4,4,3\}   | 3                                         | [2, 2, 3] [1, 1, 2] |
| \{5,5,3\}   | 3                                         | [1, 1, 2] [2, 2, 4] |
| \{3,3,4\}   | 4                                         | [1, 1, 1] [2, 2, 2] [3, 3, 3] |
| \{5,5,4\}   | 4                                         | [3, 3, 4] [2, 2, 3] [1, 1, 2] |
| \{3,3,5\}   | 5                                         | [3, 3, 2] [1, 1, 1] [4, 4, 3] [2, 2, 2] |
| \{4,4,5\}   | 5                                         | [1, 1, 1] [2, 2, 2] [3, 3, 3] [4, 4, 4] |
| \{3,4,5\}   | 3                                         | [7, 5, 4] [14, 10, 8] |
| \{3,5,5\}   | 4                                         | [5, 4, 3] [10, 8, 6] [15, 12, 9] |
| \{3,4,5\}   | 5                                         | [8, 6, 5] [16, 12, 10] [4, 3, 3] [12, 9, 8] |

The resultant equation with a Pegg Value \( \geq V \), without further multiplying the equation to re-associate \( \min(A, B, C) \), the following conditions must be met:

\[
\begin{align*}
Da/F \geq V & \quad 2^a/2^4 \geq V \quad a \geq V/8 \\
Eb/F \geq V & \quad 2^b/2^4 \geq V \quad b \geq V/2 \\
Fc/F \geq V & \quad 2^c/2^4 \geq V \quad c \geq V 
\end{align*}
\]

The Pegg Value may still be less than \( V \) due to gcd “stealing” when \( \gcd(2, c) \neq 1 \), but the above minimums must be met for the equation to have a Pegg Value \( \geq V \).

2 Generating Equations with a Desired Pegg Value

Darmon and Granville [8] showed that with fixed coefficients and exponents, there can be, at most, finitely many Original Pegg Equations with unknown integers \( a, b, c \). Oesterl´e and Masser’s ABC-conjecture implies that for fixed \( d, e, f \) coefficients, even allowing \( x, y, z \) to vary, the total number of solutions is limited, thereby implying that the total number of T-Z Conjecture counterexamples is finite.

But for our Pegg Value searching, the sizes of the coefficients on an original equation do not have an impact on its Pegg Value - only its resultant size. As \( d, e, f \) are not fixed in Original Pegg Equations, the limitation proven by Darmon and Granville does not constrain the number of original equations.

With \( W = V^{x+2} - 1 \), consider the identity \( (W^{x+2})^x + (W^{x+1})^{x+1} = (W^x V)^{x+2} \). As \( W \) and \( V \) are necessarily coprime, \( \gcd(A, B, C) = W^x \), and the Pegg Value \( \equiv V \) whenever \( W^x V < W^{x+1} \). With \( x \geq 3 \), this is true whenever \( V > 1 \). So with \( V \geq 2, x \geq 3 \), the equation yields a Resultant Pegg Equation with exponents \( \{x, x+1, x+2\} \) and a Pegg Value of \( V \). Therefore the answer to Pegg’s question: “What is the maximal value of \( \min(A, B, C) / \gcd(A, B, C) \)?” is: there is no maximal value; we can construct an equation with any desired Pegg Value. Using the above method with \( x = 3 \), we can generate a resultant equation with a Pegg Value of 60000 and a size \( \approx 2^{1270} \). This is far from proving that 60000 is the highest Pegg Value for equations \( \leq 2^{1270} \). Indeed, we can use a different identity on the \( \{3,3,5\} \) exponent set to generate smaller equations of a given Pegg Value. The obvious question arises: What is the highest Pegg Value for equations with \( C^z \leq \) a certain size?

1 Note that with \( J + K = L \), any two terms being coprime forces all three to be pairwise coprime. \( W \) and \( V \) are coprime, because they are both coprime to 1, the third term in the equation \( W + 1 = V^{x+2} \).
3 The Highest Pegg Value in All Equations \( \leq 2^{100} \)

The strategy is to guess the exponent set that will generate the highest Pegg Value within a given equation range. Then search the range for equations with higher and higher Pegg Values. Each time a new solution is found, it reduces the search space. For example, after finding an equation with a Pegg Value of 1000, we can exclude from the search any original equations that could not possibly convert to a resultant equation with a Pegg Value > 1000.

After the maximum Pegg Value is found for the given exponent set within the resultant equation size range, this Pegg Value can be used to reduce (or eliminate) the search space of other exponent sets. A quick guess as to the type of exponent sets of original equations that will yield the highest Pegg Value:

- Will have small exponents. In addition to the ABC-conjecture’s expectation that possible solutions thin out as exponents become higher, within a given equation range higher exponents reduce the size of their respective base, thereby limiting the equation’s Pegg Value.
- Will have two identical exponents. We know that the exponent set \{3, 4, 5\} yields an infinite number of Pegg Value solutions, but any coefficient on this set will necessarily need a large multiplier to make the coefficient become a perfect 3rd, 4th, and 5th power. This large multiplier significantly reduces the maximum Pegg Value within a given range.
- Will have an exponent set where the multiplier needed to convert between the original equation and its resultant equation will be as small as possible. For example, exponent set \{3, 3, 4\} requires a square-free coefficient to be cubed to convert an original equation to resultant form. But exponent set \{4, 4, 3\} requires a square-free coefficient to be 8th powered to convert an original equation to resultant form.

These conditions indicate that exponent set \{3, 3, 4\} is the best candidate exponent set. To gain some confidence in this assessment, we searched the abc@home database of 7.5 million ABC-Hits looking for hits that could be formed into Original Pegg Equations (with each exponent \( \leq 5 \)) that had a large Pegg Value in relationship to their resultant equation size.

Recall that if non-zero positive integers \(A + B = C, A < B < C, \) and \( \text{gcd}(A, B, C) = 1\), then the three are called an ABC-Triple. The radical of \(N\) is the product of the distinct primes dividing \(N\); i.e., the largest square-free factor of \(N\). The ABC-Power of an ABC-Triple is defined as \(\log(C) / \log(\text{rad}(ABC))\). If the ABC-Power of an ABC-Triple is \(> 1\), then the triple is called an ABC-Hit.

Original Pegg Equations are not necessarily ABC-Hits. For example, \(61^3 + 67^3 = 4123 \times 2^7\) converts to a resultant equation with a Pegg Value of 2, yet has an ABC-Power of only 0.7602. But ABC-Hits are excellent candidates to have a large Pegg Value in relationship to their resultant equation size (because original equations with small radicals will tend to be those that have small coefficients).

We define the Pegg Power to be \(\log(\text{Pegg Value of equation}) / \log(\text{resultant equation size})\). Similar in concept to the ABC-Power, the Pegg Power provides an easy guide as to the Pegg Value “quality” of the equation. The highest Pegg Power found was for the \{3,3,4\} Original Pegg Equation \(14 \times 111^4 + 3595^3 = 3649^3\) which converts to a resultant equation with a Pegg Value of 111 and a Pegg Power = 0.1448. This resultant equation had the highest Pegg Power of all ABC-Hits in the database and is shown in Table II as it has the highest Pegg Value of all \{3,3,4\} equations < 3558^3. Forty-nine of the top fifty Pegg Powers in the database were \{3,3,4\} equations. The highest non-\{3,3,4\} Pegg Power in the database was thirteenth place 0.10677 for a \{3,3,5\} equation.

This information lends support to the plan to find the highest Pegg Value in \{3,3,4\} equations \(\leq 2^{100}\), and use that result to reduce or eliminate the search space in other exponent sets.
3.1 Perfect Power Testing

In order to search for the highest Pegg Value equation \( \leq \ 2^{100} \), we will need a perfect power tester for cubes, fourth powers, and fifth powers. But each power testing algorithm can be distinct. Having three different perfect power testers reduces the runtime inefficiency of having a general-purpose perfect power tester that checks an input for being a perfect \( y \)-th power against multiple \( y \).

Our perfect power testers work along the lines used by GMP for testing perfect squares [10], namely, before performing a rigorous but expensive \( y \)-th power test on an input, first verify that the input is compatible with being a \( y \)-th power modulo small integers. Most non-powers will quickly be eliminated, and for those inputs that pass the residue testing, the algorithm will then perform a rigorous perfect power test.

For perfect cube testing, primes \( p \equiv 1 \pmod{3} \) each rule out \( \frac{2}{3}(p - 1)/p \) possible inputs. In addition, powers of an unused prime can assist. For cubic residues, the modulus 9 rules out 6/9 of possible inputs. Most non-powers will quickly be eliminated, and for those inputs that pass the residue testing, the algorithm will then perform a rigorous perfect power test.

For perfect cube testing, primes \( p \equiv 1 \pmod{3} \) each rule out \( \frac{2}{3}(p - 1)/p \) possible inputs. In addition, powers of an unused prime can assist. For cubic residues, the modulus 9 rules out 6/9 of possible inputs. Most non-powers will quickly be eliminated, and for those inputs that pass the residue testing, the algorithm will then perform a rigorous perfect power test.

For perfect cube testing, primes \( p \equiv 1 \pmod{3} \) each rule out \( \frac{2}{3}(p - 1)/p \) possible inputs. In addition, powers of an unused prime can assist. For cubic residues, the modulus 9 rules out 6/9 of possible inputs. Most non-powers will quickly be eliminated, and for those inputs that pass the residue testing, the algorithm will then perform a rigorous perfect power test.

For fourth powers, checking inputs for being compatible with being a cube modulo 9 and the 34 primes \( p \equiv 1 \pmod{3} \) with \( p \leq 367 \) rules out 99.9999999999999946% of all inputs.

For fifth powers, checking inputs for being compatible with being a fifth power modulo 25, and the 23 primes \( p \equiv 1 \pmod{5} \leq 521 \) rules out 99.99999999999999571% of all inputs.

Residue testing in such a manner offers another benefit. Because all our perfect power testing will ultimately be done on inputs of the form \( a^x - fc^z \) and \( fc^z - a^x \), we do not need to compute the actual difference before performing residue tests. We can use modular powering, and only compute the true total when we need the rigorous perfect power check. For more runtime efficiency, all the modular powering operations can be replaced with precalculated table lookups so no actual modular powerings are needed.

Our computer algorithm would instinctively seem to require expensive division instructions to perform residue calculations, but divisions needed for the modulus operation (\% in C) can be avoided entirely by structuring the code to allow the compiler to optimize away actual divisions and instead use multiplicative inverses [11] [12]. In GCC, this requires using a hard-coded literal as a modulus (not stored in an array – even a constant array).

The smallest allowable Pegg Equation exponent is 3, so using bases up to \( 2^{64} - 1 \) allows us to handle all inputs through \( (2^{64} - 1)^3 \) while still using the natively supported 64-bit unsigned integer size of the host computer. This also contributes to impressively minimal testing time.

As we can perform the residue tests with multiplicative inverses instead of divisions, and can perform memory lookups instead of modular powering operations, the time to perform a perfect \( y \)-th power test for non-\( y \)-th power inputs is effectively independent of the size of the inputs \( a, x, f, c, z, \) and \( y \).
3.2 Algorithm to Find Resultant Pegg Equations with a Minimum Desired Pegg Value

This section will present an algorithm that will solve the simple case when two exponents are identical, and then conclude with a discussion of how it also works for the more complicated case when all exponents are different (but still have only a single coefficient).

When two exponents are the same we will map them to the \( x \) and \( y \) exponents, allowing us to follow our convention that exponent \( z \) is associated with the base with the coefficient. The equation \( \pm a^x \pm b^y \pm f c^z = 0 \) with \( x, y, z \geq 3; a, b, c \geq 1 \) has three permutations when the signs are all positive. A convenient naming system is to title these permutations according to what \( b^y \) equals:

| Equation Permutation | Permutation Name |
|----------------------|------------------|
| \( a^x + f c^z = b^y \) | ax-minus-\( cz \) |
| \( f c^z - a^x = b^y \) | cz-minus-\( ax \) |
| \( a^x - f c^z = b^y \) | ax-plus-\( cz \) |

When two exponents are the same, the first and last permutations listed above are identical, and the set of permutations to check can be reduced to \( ax_{-\text{minus}\_\text{cz}} \) and \( cz_{-\text{minus}\_\text{ax}} \).

Our algorithmic strategy is to loop through the valid \( f \) coefficients, loop through the valid \( c \) bases, then loop through the valid \( a \) bases then, depending on the permutation that is being checked, see if \( f c^z - a^x \) or \( a^x - f c^z \) yields a perfect \( y \)-th power. Consider the exponent set \( \{3,3,4\} \) with \( x = 3, y = 3, z = 4 \). There are fewer fourth powers within a given range than there are perfect cubes. And our perfect cube test takes about the same amount of time to perform regardless of the offset of the input. So regarding the two bases with identical exponents, it makes sense for us to loop through the range of cubes that has a smaller number of cubes than the other range.

Let \( t = \) an offset \( > 0 \)

Let \( r_{\text{min}} = \) range minimum \( > 0 \) to search for a perfect power

Let \( r_{\text{max}} = \) range maximum \( > r_{\text{min}} \) to search for a perfect power

There are fewer cubes between \( t+r_{\text{min}} \) and \( t+r_{\text{max}} \) than between \( r_{\text{min}} \) and \( r_{\text{max}} \). For efficiency, we establish the convention when \( x = y \), the \( a \) base is larger than the \( b \) base.

With permutation type \( ax_{-\text{minus}\_\text{cz}} \), we have \( a^x - c^z = b^y \) equivalently \( a^x = c^z + b^y \). As applied to our application, there are fewer cubes between \( c^z + b^y \) and \( c^z + b^y_{\text{max}} \) than there are between \( b^y_{\text{min}} \) and \( b^y_{\text{max}} \). So for the permutation type \( ax_{-\text{minus}\_\text{cz}} \), in comparison to looping through the \( b \) base and then checking if the sum \( f c^z + b^y \) is a perfect cube, it is more efficient to loop through the \( a \) base, then check if the difference \( a^x - f c^z \) is a perfect cube. With type \( cz_{-\text{minus}\_\text{ax}} \), we have \( f c^z - a^x = b^y \) equivalently \( a^x = f c^z - b^y \). Similarly, it will be more efficient to loop through the \( a \) base, and check the difference \( f c^z - a^x \) and see if the result is a perfect cube.

We can extend the use of perfect power residues even further. Consider permutation type \( cz_{-\text{minus}\_\text{ax}} \). For each \( c \) base, instead of looping through the entire range of possible \( a \) bases and seeing if \( f c^z - a^x \) is a perfect \( y \)-th power, we can instead loop through only the \( a \) bases for a given \( c \) base that are compatible with \( f c^z - a^x \) being congruent to a perfect \( y \)-th power modulo a few small integers.

Consider again the example exponent set \( \{3,3,4\} \), with \( x = y = 3; z = 4 \). and permutation type \( cz_{-\text{minus}\_\text{ax}} \) with \( f = 8; c = 103; a = 21 \). We know \( f c^z - a^x \) cannot be a cube because \( 8 \times 103^3 - 21^3 \equiv 2 \pmod{7} \), which is incompatible with being a perfect cube (a perfect cube is \( \equiv 0, 1, \text{or} 6 \pmod{7} \)).
Since we know beforehand that with a \( f c^z \) residue of 2 (mod 7) that an \( a \) base of 21 is incompatible with \( f c^z - a^z \) being a perfect cube, there is no sense checking an \( a \) base of 21 in the first place. We can precalculate a large lookup array that, for a given \( f c^z \), shows the offsets between \( a \) bases that are compatible with the difference being a perfect \( y \)-th power (modulo the product of several small integers). These moduli will be smallest moduli from the list generated for our perfect power tester. Increasing the number of small moduli in this \( \text{a\_base\_skipahead} \) modulus product increases the size of the table (and table precalculation time), but decreases the number of possible \( a \) bases that must be checked. This table gets large quickly, because it must list the deltas between acceptable \( a \) bases (per the \( \text{a\_base\_skipahead} \) modulus) for every possible \( f c^z \) residue. Basically, the strategy is to multiply together as many of the smallest moduli together to produce a single \( \text{a\_base\_skipahead} \) modulus product such that the lookup table will still fit within computer RAM (such as within a typical 4 GiB limit). Then any \( a \) bases that are selected for a given \( f c^z \) residue will automatically be a perfect \( y \)-th power modulo the \( \text{a\_base\_skipahead} \) modulus product. So the perfect power testing routine will then use remaining moduli that were not used in the \( \text{a\_base\_skipahead} \) modulus product.

In addition, sometimes we will know immediately upon checking the current \( f c^z \) that there are no \( a \) bases that could generate a \( f c^z - a^z \) congruence compatible with the result being a perfect \( y \)-th power. For example, if \( f c^z \equiv 4 \pmod{7} \), there is no \( a \) base such that cubing it results in \( f c^z - a^z \) being congruent to a cube \( \pmod{7} \). In this case the entire \( \text{a\_base\_loop} \) can be avoided. This lookup table is quite small because it is simply a list of \( f c^z \) residues that can be skipped.

The specifics of generating the \( \text{a\_base\_elimination} \) and \( \text{a\_base\_skipahead} \) tables are detailed in Appendix A. The key point is that once the tables are precalculated, they continually eliminate over 99% of possible inputs without any further calculations. A single increment through the \( \text{a\_base\_skipahead} \) table eliminates a large swath of possible inputs.

The specifics of how to calculate the range of \( c \) bases, \( f \) coefficients, and \( a \) bases are detailed in Appendix B. Of note is a significant reduction in search space for the permutation type \( cz\_minus\_ax \): \( f c^z = a^x + b^y \). When \( x = y \), either \( a^x \) or \( b^y \) must be \( \geq f c^z/2 \), and the other must be \( \leq f c^z/2 \). This reduces the range to check from the standard range: \( a^x = a_{\min} \cdot f c^z - b^y_{\min} \) [when \( x \neq y \)] to \( a^x = f c^z/2 \cdot f c^z - b^y_{\min} \) [when \( x = y \)]. With this reduction, it takes much less time to check the \( cz\_minus\_ax \) permutation than the \( ax\_minus\_cz \) permutation.

Notice that the gcd condition on line [7] is checked after first verifying that the appropriate difference is a perfect power. If the gcd check was performed before checking for a perfect power, it would rule out \( \approx 20\% \) of possibilities (approximately \( 1 - (6/\pi^3) \)) but even though gcd is an efficient algorithm, we can test for perfect cubes even more quickly than calculating gcd when most inputs are not perfect cubes. And being that the difference is rarely a perfect cube, it is more efficient to test the gcd condition after first verifying the perfect \( y \)-th power condition.

A note on line 8: “Calculate maximum Pegg Value with this original equation within given resultant equation range”

The first step is to multiply the entire equation by smallest \( N \) to produce a valid Resultant Pegg Equation. At this point, the Pegg Value will be \( \geq \) the desired Pegg Value unless either:

- \( \text{min}(A, B, C) \) is not associated with the base to the highest exponent. In this case, the Pegg Value may be able to be increased by multiplying the entire equation by a number that moves \( \text{min}(A, B, C) \) to a different base. Of course, the new resulting equation still must fit within the desired equation range. See [11.4]
- \( \text{min}(A, B, C) \) is associated with a base to the highest exponent, and the gcd “steals” a factor from the base associated with the original coefficient. In this case, nothing can increase the Pegg Value. See [11.5]
Algorithm 1 Within the specified equation range, find an equation with a minimum desired Pegg Value that is based on an Original Pegg Equation with a single coefficient

1: for permutation in $ax_{\text{minus}cz}$ to $cz_{\text{minus}ax}$ do
2: for $c$ base in all possible inside resultant eq range with this permutation do
3: for $f$ coeff in all possible inside resultant eq range with this permutation and $c$ base do
4: if $a_{\text{base}}$ elimination table does not rule out all $a$ bases for this $f$ and $c^z$ then
5: for $a$ base in all possible inside resultant eq range with this permutation and $c$ base and $f$ coeff [use the $a_{\text{base}}$ skipahead lookup table to only check the $a$ bases that are compatible with the result being congruent to a $y$-th power modulo $a_{\text{base}}$ skipahead modulus] do
6: if (permutation = $ax_{\text{minus}cz}$ and $a^x - fc^z = \text{perfect } y\text{-th power}$) or (permutation = $cz_{\text{minus}ax}$ and $fc^z - a^x = \text{perfect } y\text{-th power}$) then
7: if gcd($a, c$) = 1 then
8: if the maximum Pegg Value $\geq$ minimum desired Pegg Value then
9: return Pegg Value, Resultant Pegg Equation
10: end if
11: end if
12: end if
13: end if
14: end for
15: end for
16: end for
17: end for
18: return entire range searched - none exist

This algorithm can also be used for exponent sets where all exponents are different by reordering the input parameters. The standard input parameters are $\{x, y, z\}$ where $x$ and $y$ are exponents associated with bases with no coefficients.

- $z =$ exponent associated with the base with coefficient (the algorithm will loop through these $c$ bases)
- $x =$ exponent associated with the base with the smaller number of elements (the algorithm will loop through these $a$ bases)
- $y =$ exponent associated with the base with the larger number of elements (these are detected by performing perfect $y$-th power tests on $f c^z - a^x$ or $a^x - f c^z$)

When $x = y$, the permutation $ax_{\text{plus}cz}$ was ignored, as it was identical to $az_{\text{minus}cz}$. But when $x \neq y$, the unprogrammed permutation $ax_{\text{plus}cz}$ can be checked by reordering the inputs to the algorithm. Consider the exponent set $\{3,4,5\}$ where the coefficient is associated with the base to the third power. We follow our convention that exponent $z$ is associated with the base with the coefficient, so $z = 3$.

The following table shows how we check all three permutation for $z = 3$ with our current algorithm which handles the two permutation types $ax_{\text{minus}cz}$ and $cz_{\text{minus}ax}$:

| Equation Permutation | Handled with input parameter ordering $\{x, y, z\}$ |
|----------------------|-----------------------------------------------|
| $a^x - fc^z = b^y$   | $ax_{\text{minus}cz}$ with exponent set $\{5,4,3\}$ |
| $fc^z - a^x = b^y$   | $cz_{\text{minus}ax}$ with exponent set $\{4,5,3\}$ or $\{5,4,3\}$ |
| $a^x + fc^z = b^y$   | $ax_{\text{minus}cz}$ with exponent set $\{4,5,3\}$ |

11
Table 3: All combinations of T-Z Conjecture counterexample exponent sets with each exponent \(\leq 5\), along with a possible reference to a proof showing that none exist

| exponent set          | reference to proof that exponent set has no T-Z Conjecture counterexamples |
|-----------------------|------------------------------------------------------------------------------|
| \{n,n,n\}            | Wiles and Taylor. Fermat’s last Theorem. [18][19] although the specific cases with \(3 \leq n \leq 5\) were all solved earlier |
| \{3,3,3\}            | Euler (18th century)                                                        |
| \{4,4,4\}            | Fermat (17th century)                                                       |
| \{5,5,5\}            | Dirichlet and Legendre (19th century)                                       |
| \{3,3,4\}            | Bruin [1]                                                                   |
| \{3,3,5\}            | Bruin [1]                                                                   |
| \{4,4,3\}            | Lucas (19th century)                                                        |
| \{4,4,5\}            | Bruin [20] showed complete list of \{2,4,5\}                               |
| \{5,5,3\}            | Poonen [21]                                                                 |
| \{5,5,4\}            | Poonen [21] ruled out \{5,5,2\}                                             |
| \{3,4,5\}            | has not been ruled out                                                       |

Notice that the second possibility can be checked with exponent set \{4,5,3\} or exponent set \{5,4,3\}. Our ax\textsubscript{minus}cz and cz\textsubscript{minus}ax search algorithms run more efficiently with \(x \geq y\), so it will be most efficient to check exponent set \{5,4,3\}.

For the third possibility, it is less efficient at runtime to check ax\textsubscript{minus}cz with exponent set \{4,5,3\} than using a new ax\textsubscript{plus}cz routine and checking exponent set \{5,4,3\}, but the inefficiency is not great enough to overcome the overhead of writing a new routine. This searching does not take very long because there are far fewer combinations to check when all three exponents are different.

Using the above algorithm, it is easy to determine that the smallest \{3,3,4\} equation with a Pegg Value \(> 1\) is \(207^3 + 126^3 = 639^3\) which has Pegg Value of 14. Searching \{3,3,4\} equations up to \(2^{100}\) for higher and higher Pegg Values yields the information in Table 3 and Figure 1. The last row in Table 3 shows a resultant equation \(\leq 2^{100}\) with a Pegg Value of 63742. The next step is to prove the hunch that this is the highest Pegg Value in all equations \(\leq 2^{100}\).

### Lemma 1

There are no T-Z Conjecture counterexamples \(\leq 2^{100}\) with a Pegg Value \(> 63742\).

Any equation with a higher Pegg Value must have all bases \(> 63742\). This precludes any base having an exponent of 7 or higher, as \(63743^7 \geq 2^{100}\).

A good summary of exponent sets for which it has been proven that no T-Z Conjecture counterexamples exist is provided in both [14] and [15]. Of all exponent sets with each exponent \(\leq 5\), all but one have been ruled out as having any T-Z Conjecture counterexamples. This information is summarized in Table 3.

Note that exponent 6 is covered under exponent 3, as \(a^6 = (a^2)^3\), so for T-Z Conjecture counterexamples we only need to consider exponent sets with each exponent \(\leq 5\).

The only T-Z Conjecture counterexample exponent set listed above that has not been ruled out is \{3,4,5\}. Notice that \{3,4,5\} can be rewritten as a special form of \{2,3,5\} for which the complete parameterization was shown by Johnny Edwards [16] and conveniently available on Dario Alpern’s web-site [17]. Using these parameterizations, checking all solutions for

\[\pm a^2 \pm b^3 \pm c^5 = 0\] with \(a, b, c \geq 1, \gcd(a, b, c) = 1\) shows that for equations \(\leq 2^{800}\), in no case is \(a\), the base to be squared, itself a perfect square. So there are no \{3,4,5\} T-Z Conjecture counterexamples with an equation size \(\leq 2^{800}\), thereby proving that there are no T-Z Conjecture counterexamples of any exponent set under an equation size \(\leq 2^{100}\) that have a Pegg Value \(> 63742\).
Table 4: Smallest \{3,3,4\} resultant equation with a Pegg Value higher than that shown in the preceding row (Row 1 shows the smallest \{3,3,4\} equation with a Pegg Value > 1).

| \log_2 \text{ (resultant equation)} | Pegg Value | Pegg Power | Original Equation |
|-------------------------------------|------------|------------|-------------------|
| 27.96                               | 14         | 0.1362     | \(23^4 + 9 \times 14^4 = 71^4\) |
| 33.81                               | 21         | 0.1299     | \(13 \times 21^4 + 163^4 = 190^4\) |
| 43.80                               | 43         | 0.1239     | \(23 \times 43^4 + 1056^4 = 1079^4\) |
| 46.92                               | 111        | 0.1448     | \(14 \times 111^4 + 3595^4 = 3649^4\) |
| 56.75                               | 133        | 0.1243     | \(1157^4 + 139 \times 133^4 = 3558^4\) |
| 57.82                               | 183        | 0.1300     | \(1966^4 + 121 \times 183^4 = 5233^4\) |
| 60.68                               | 194        | 0.1252     | \(126 \times 194^4 + 9071^4 = 9743^4\) |
| 66.96                               | 201        | 0.1143     | \(5006^4 + 8809^4 = 545 \times 201^4\) |
| 66.98                               | 365        | 0.1271     | \(10973^4 + 15002^4 = 301 \times 365^4\) |
| 69.24                               | 399        | 0.1248     | \(12146^4 + 391 \times 399^4 = 22703^4\) |
| 72.75                               | 455        | 0.1214     | \(513 \times 455^4 + 33247^4 = 38872^4\) |
| 73.74                               | 1482       | 0.1429     | \(1609^4 + 239 \times 1482^4 = 104857^4\) |
| 73.81                               | 1638       | 0.1447     | \(97103^4 + 193 \times 1638^4 = 132095^4\) |
| 74.25                               | 2994       | 0.1555     | \(104 \times 2994^4 + 226199^4 = 271127^4\) |
| 90.12                               | 3858       | 0.1322     | \(25031^4 + 1570 \times 3858^4 = 703271^4\) |
| 90.16                               | 5838       | 0.1388     | \(729217^4 + 971 \times 5838^4 = 1148689^4\) |
| 90.82                               | 11598      | 0.1487     | \(341 \times 11598^4 + 3662591^4 = 3809903^4\) |
| 92.75                               | 49476      | 0.1681     | \(7771657^4 + 8824055^4 = 193 \times 49476^4\) |
| 99.91                               | 63742      | 0.1597     | \(2192137^4 + 20440855^4 = 518 \times 63742^4\) |

Figure 1: Highest \{3,3,4\} Pegg Values for a given resultant equation size (information from Table 4).
Lemma 2 There are no Resultant Pegg equations \( \leq 2^{100} \) with a Pegg Value \( > 63742 \) with an exponent of 6

In an equation \( \leq 2^{100} \), the maximum possible base to a sixth power is \( [2^{100}/6] = 104031 \). The only reason to use exponent 6 instead of 3 on a particular base would be if one wanted to force that particular base to be smaller so that it would be selected as \( \min(A, B, C) \). When \( \min(A, B, C) \) is associated with an exponent of 6, \( \min(A, B, C)/\gcd(A, B, C) \) can be no higher than \( [104031/2] = 52015 \), because \( \gcd(A, B, C) \) must be \( \geq 2 \). As 52015 \( < 63743 \), there is no reason to consider equations with an exponent of 6.

Lemma 3 There are no Resultant Pegg equations \( \leq 2^{100} \) with a Pegg Value \( > 63742 \) with two exponents the same

The Original Pegg Equation condition that \( \gcd(x, y, z) = 1 \) precludes \( z = x \), and as all exponents to consider are \( < 6 \), the following sets must be checked: \( \{4,4,3\}, \{5,5,3\}, \{3,3,4\}, \{5,5,4\}, \{3,3,5\}, \{4,4,5\} \). Regarding the search for a \( \{3,3,4\} \) equation with a coefficient of 2, Henri Cohen \(^{22}\) showed that the equation:

\[ \pm a^3 \pm b^3 \pm 2c^2 = 0, \text{ with } a, b, c \geq 1, \gcd(a, b, c) = 1 \]

can be parameterized by one of four parameterizations. Using these parameterizations, checking all solutions with \( 2c^2 \leq 2^{240} \) shows that in no case is \( c \), the base to be squared, itself a perfect square. So there are no \( \{3,3,4\} \) original equations \( \leq 2^{240} \) with a coefficient of 2, thereby eliminating any need for our algorithm to consider a coefficient of 2 for exponent \( \{3,3,4\} \) in a search for any resultant equations \( \leq 2^{100} \).

The remaining coefficients for \( \{3,3,4\} \) and the other listed exponent sets were checked with Algorithm \( ^{1} \) and in no case was an Original Pegg Equation found that could be converted to a Resultant Pegg Equation \( \leq 2^{100} \) with a Pegg Value \( > 63742 \).

Lemma 4 There are no Resultant Pegg Equations \( \leq 2^{100} \) with a Pegg Value \( > 63742 \) with no two exponents the same

As all exponents to consider are \( < 6 \), the only the set to check is \( \{3,4,5\} \). All the other exponent sets to check had two identical exponents, so we mapped the non-identical exponent to \( z \) according to our convention that the sole base that had a coefficient \( > 1 \) would be mapped to the \( c \) base. For exponent set \( \{3,4,5\} \) all bases could have a coefficient \( > 1 \), so it is not obvious how to map the \( x, y, z \) exponents. To ease confusion, for now we will simply map: \( x = 3 \); \( y = 4 \); \( z = 5 \).

First to note is that there is not room to multiply the entire equation by a number in order to increase its Pegg Value by moving \( \min(A, B, C) \) from being associated with a smaller exponent to a larger exponent: Per \( ^{13} \), the smallest possible \( \gcd(A, B, C) = F \), so the largest Pegg Value for a \( \{3,4,5\} \) equation is \( \min(A, B, C)/F \). The highest value for \( C \) is \( [2^{100}/5] = 1048576 \). So \( \min(A, B, C) \) must be \( \leq 1048576 \). As the Pegg Value is limited to \( \min(A, B, C)/F \), the Pegg Value \( \leq [1048576/F] \). For the Pegg Value to be \( \geq 63743 \), \( F \) must be \( \leq [1048576/63743] \leq 16 \). The smallest number by which the entire equation could be multiplied is \( 2^{\text{lcm}(x,y,z)} = 2^{60} \), which would contribute to a resultant coefficient \( F \) of \( 2^{60}/5 = 4096 \), which is greater than the maximum of 16.

Second to note is that \( \{3,4,5\} \) can have only single coefficient. There must be at least one original coefficient \( > 1 \), therefore the resulting \( D, E, F \) will all be \( > 1 \), and we also know that \( F \leq 16 \). We must answer the question, “Under what conditions can an original \( \{3,4,5\} \) equation result in \( 1 \leq F \leq 16 \)?”

Our search is limited to \( F \leq 16 \), so \( v_p(F) \leq 4 \), for every \( p | F \). Looking at Table 2 for options where a \( \{3,4,5\} \) equation has \( F \leq 4 \)th power, we have:

\( \{3,4,5\} \), with \( v_p(d) = 1, F = p^4 \) \( \{3,4,5\} \), with \( v_p(e) = 1, F = p^3 \)
\( \{3,4,5\} \), with \( v_p(f) = 3, F = p^3 \)

There cannot be coefficients on more than one base (because there is only room for a
maximum resultant coefficient of $2^4$ for $F$ and the smallest combination of two coprime coefficients would yield a resultant coefficient of $F = p_1^3 p_2^4$ which would be $> 2^4$, the maximum value of $F$). And $p$ can only $= 2$, as the smallest possible $F$ is $p^4$ and cubing the next higher base yields $3^3$ which is greater than the maximum $F$ of 16. The search for \{3,4,5\} original equations that convert to resultant equations $\leq 2^{100}$ with a Pegg Value $> 63742$ is limited to:

- $\{3,4,5\}$ with a coefficient of 2 associated with the base to the 3rd power.
- $\{3,4,5\}$ with a coefficient of 2 associated with the base to the 4th power.
- $\{3,4,5\}$ with a coefficient of 8 associated with the base to the 5th power.

As the equation cannot be multiplied by a number to re-associate $\min(A, B, C)$, \{1,6\} allows us to calculate the original base minimums for each base (and shows the first \{3,4,5\} case worked as an example). The three coefficient scenarios were all checked with Algorithm \[\] and in no case was an Original Pegg Equation found that could be converted to a Resultant Pegg Equation $\leq 2^{100}$ with a Pegg Value $> 63742$.

**Conclusion**

The last row of Table 4 shows a \{3,3,4\} Original Pegg Equation that can be converted to the equation $1135526966^3 + 10588362890^3 = 330183564^4$ that has a Pegg Value of 63742 and a $C^2 \leq 2^{100}$. Lemmas 1 through 4 show that there are no equations $\leq 2^{100}$ of any exponent set that have a higher Pegg Value.

Searching and proof verifying took place over several weeks on three AMD 64-bit computers; from two to four cores - 2.5 Ghz to 3.0 Ghz; 4 GiB to 8 GiB of memory; running NetBSD 4.0, 4.0.1 or 5.0 [23] with the operating system included GCC 4.1.2 or 4.1.3 [24]; and using GMP 4.2.3 or 4.3.1 [25].

The highest Pegg Power of the equations in Table 4 is 0.1681. The Pegg Power limit for \{3,3,4\} equations and \{4,4,3\} equations is $\lim_{V \to \infty} \log(V) / \log((2V)^4) = 0.25$. Any T-Z Conjecture counterexample or other Pegg Equation exponent set will have at least one exponent $\geq 5$, which limits the Pegg Power to 0.2. So the highest Pegg Power for all equations lies somewhere within the range 0.1681 and 0.25. Rather than simply looking to increase the lower limit by further searching, a better technique to reduce this range could help shed light on reducing or eliminating possible T-Z Conjecture counterexamples. What is the highest Pegg Power?

**4 Acknowledgements**

Without the availability of GMP, the GNU Multiple Precision Arithmetic Library, the work that resulted in this paper would not have been started. I thank the authors and maintainers for such a great product.

**5 Electronic Availability**

The Pegg Value searching code, written in C with calls to GMP, is released under the GNU General Public License (GPL) and is available at http://www.arthurtownsend.com/peggvalue.htm.

**A Generating the a_base_skipahead and a_base_elimination Lookup Tables**

This appendix details the suggested a_base_elimination moduli and a_base_skipahead moduli for all exponent sets with each exponent $\leq 5$. The use of these tables was
elimination table is a list of \( f^c \) residues for which there are no \( a \) bases that can generate a \( f^c \) and \( a \) combination that is a \( y \)-th power. For these \( f^c \), the entire \( a \)-base loop can be avoided.

The \( a \)-base_skipahead table is a large lookup array showing the offsets between possible \( a \) bases for a given \( f^c \) (offsets between \( a \) bases that are not incompatible with the \( f^c \) and \( a^x \) combination being a perfect \( y \)-th power modulo the \( a \)-base_skipahead modulus). Any \( a \) bases that are selected will automatically generate an \( f^c \) and \( a^x \) combination that is congruent to a perfect \( y \)-th power (modulo \( a \)-base_skipahead modulus), so the perfect power testing routine will then use \( y \)-th power testing moduli that were not used in the \( a \)-base_skiphead modulus product.

The tables based on these moduli were used to generate the information in Table 4 and also used for verifying Lemmas 3 and 4.

The bulk of the time of the algorithm will be spent on the coefficients that produce the largest number of combinations of \( c \) bases and \( a \) bases within a given resultant equation range. For \( \{3,3,4\} \), these will be the smallest coefficients. The \( \{3,3,4\} \) \( a \)-base_skiphead table for all possible coefficients is 2.6 GiB for the modulus 13 * 19 * 31 * 37. The single coefficient “3” provides so many \( c \) base and \( a \) base combinations within a given range that it is worthwhile to generate a \( \{3,3,4\} \) single-coefficient \( a \)-base_skiphead table just for the coefficient 3. For a given \( a \)-base_skiphead modulus, having an \( a \)-base_skiphead table for only a single coefficient produces a much smaller table than a table that can handle all coefficients. And because the resulting table is much smaller, we can use a larger modulus 7 * 13 * 19 * 31 * 37 and keep the table size less than our 4 GiB limit. Using the additional 7 as part of the modulus rules out \( 2/3 \) (7 - 1)/7 \( \approx \) 57% more \( a \) bases than the original \( a \)-base_skiphead table. Note that these savings do not result in 57% fewer \( a \) bases being checked, because some of those now rejected would already have been rejected by the \( a \)-base_elimination modulus which also includes the modulus of 7. But the benefits of a custom individual coefficient \( a \)-base_skiphead modulus are sufficiently great that they can be used for a few coefficients on the \( \{3,3,4\} \) exponent set (as shown in Table 6 column 4). It is not worthwhile using a custom \( a \)-base_skiphead table for every coefficient on the \( \{3,3,4\} \) exponent set, because the higher the coefficient the fewer combinations it produces, and the benefit of the 57% \( a \) base reduction must be balanced against the single-coefficient table precalculation time.

Some notes on the tables:

- Regarding the coefficients tested individually (listed in Table 6 Column 4):
  - Table 4 shows that regardless of the powers of the prime factors that divide \( f \), for exponent set \( \{3,3,4\} \) the smallest \( N \) possible is always \( N = f^3 \). So for \( \{3,3,4\} \), the smaller the coefficient, the more combinations there are to be checked.
  - Table 1 shows that the multiplier to convert a \( \{4,4,3\} \) Original Pegg Equation to its resultant equation is smallest when \( v_p(f) = 2 \), so for \( \{4,4,3\} \), the smaller and the “more square” the coefficient, the more combinations there are to be checked.

Exponent set \( \{3,3,4\} \) shows the coefficient 7 being included in the list of coefficients checked individually. For coefficient 7, the listed individual \( a \)-base_skiphead modulus 7 * 13 * 19 * 31 * 37 offers no reduction in the \( a \) base search space over the smaller 13 * 19 * 31 * 37. So for coefficient 7, either the standard \( a \)-base_skiphead table or single-coefficient \( a \)-base_skiphead table could be used with identical effect, or searching efficiency could be improved by using a modulus of 9 * 13 * 19 * 31 * 37.

Exponent set \( \{3,3,5\} \) does not have coefficients checked individually because after increasing the standard \( a \)-base_skiphead modulus, even for use with just a single coefficient, the resulting \( \{3,3,5\} \) single-coefficient \( a \)-base_skiphead table is 16 GiB (which exceeds the 4 GiB limit). Other exponent sets do not take long to check, so it is not worth the overhead to precalculate single-coefficient \( a \)-base_skiphead tables for them.

For a given exponent set, the \( ax \_minus \_cz \) and \( cz \_minus \_ax \) \( a \)-base_elimination
Table 5: Suggested a\textsubscript{base elimination} moduli and standard a\textsubscript{base skipahead} moduli.

For each exponent set \{x, y, z\} shown in column 1 and permutation type shown in column 2, column 3 shows the a\textsubscript{base elimination} modulus to generate the list of \(fe^z\) residues that result in no possible \(a^x\) being able to produce a \(fe^z\) and \(a^x\) combination that is perfect \(y\)-th power.

Column 4 shows the standard a\textsubscript{base skipahead} modulus. This modulus results in an a\textsubscript{base skipahead} table less than 4 GiB that can handle all coefficients.

For all possible \(c\) bases and for all \(f\) coefficients \(\leq 100000\), columns 5, 6, and 7 show how many \(a\) bases are eliminated by the listed lookup table as being able to produce an \(fe^z\) and \(a^x\) combination that could be perfect \(y\)-th power. Column 5 shows what percentage would be eliminated solely based on the a\textsubscript{base elimination} modulus. Column 6 shows what percentage would be eliminated solely based on the a\textsubscript{base skipahead} modulus. Column 7 shows what percentage were eliminated based on both tables.

| exponent set | permutation type | a\textsubscript{base elimination} modulus | standard a\textsubscript{base skipahead} modulus | \% of inputs eliminated by a\textsubscript{base elimination} modulus | \% of inputs eliminated by a\textsubscript{base skipahead} modulus | \% of inputs eliminated by both tables |
|--------------|-----------------|------------------------------------------|-----------------------------------------------|-------------------------------------------------|-------------------------------------------------|----------------------------------------|
| \{3,3,4\}   | both            | \(7 \times 3^2\)                        | \(13 \times 19 \times 31 \times 37\)          | 47.149                                          | 97.596                                          | 98.729                                 |
| \{3,3,5\}   | both            | \(7 \times 3^2\)                        | \(13 \times 19 \times 31 \times 37\)          | 46.956                                          | 97.596                                          | 98.725                                 |
| \{4,4,3\}   | ax\_minus \(cz\) | \(5 \times 2^4 \times 17\)             | \(3^2 \times 13 \times 29 \times 37\)        | 77.398                                          | 98.063                                          | 99.562                                 |
| \{4,4,3\}   | cz\_minus \(ax\) | \(5 \times 2^4 \times 17\)             | \(3^2 \times 13 \times 29 \times 37\)        | 91.228                                          | 99.160                                          | 99.853                                 |
| \{4,4,5\}   | ax\_minus \(cz\) | \(5 \times 2^4 \times 17\)             | \(3^2 \times 13 \times 29 \times 37\)        | 69.265                                          | 98.022                                          | 99.392                                 |
| \{4,4,5\}   | cz\_minus \(ax\) | \(5 \times 2^4 \times 17\)             | \(3^2 \times 13 \times 29 \times 37\)        | 91.228                                          | 99.160                                          | 99.853                                 |
| \{5,5,3\}   | both            | \(11 \times 5^2 \times 31 \times 41 \times 61\) | \(61 \times 71 \times 101\)                 | 87.392                                          | 98.915                                          | 99.830                                 |
| \{5,5,4\}   | both            | \(11 \times 5^2 \times 31 \times 41 \times 61\) | \(41 \times 61 \times 71\)                 | 87.392                                          | 98.915                                          | 99.830                                 |
| \{4,3,5\}   | both            | \(13 \times 7 \times 3^2 \times 19 \times 31\) |                                         | 7.101                                          | 99.475                                          | 99.512                                 |
| \{5,3,4\}   | both            | \(31 \times 7 \times 3^2 \times 13 \times 19\) |                                         | 49.951                                          | 99.773                                          | 99.886                                 |
| \{3,4,5\}   | both            | \(13 \times 5^2 \times 2^4 \times 17 \times 29\) |                                         | 7.101                                          | 99.140                                          | 99.758                                 |
| \{5,4,3\}   | both            | \(11 \times 41 \times 5^2 \times 13 \times 2^4\) |                                         | 17.002                                          | 99.638                                          | 99.699                                 |
| \{3,5,4\}   | both            | \(31 \times 11 \times 5^2 \times 41 \times 61\) |                                         | 49.951                                          | 99.993                                          | 99.997                                 |
| \{4,5,3\}   | both            | \(11 \times 41 \times 5^2 \times 31 \times 61\) |                                         | 17.002                                          | 99.865                                          | 99.888                                 |
Table 6: Suggested single-coefficient a\textsubscript{base}skipahead moduli. For each exponent set \{x, y, z\} shown in column 1 and permutation type shown in column 2, column 3 shows the single-coefficient a\textsubscript{base}skipahead modulus. This modulus results in an a\textsubscript{base}skipahead table less than 4 GiB that can handle just a single coefficient. For the two exponent sets that take the longest to search, it is worthwhile to check the coefficients that result in the most combinations with their own individual a\textsubscript{base}skipahead table. Column 4 lists the coefficients that were checked individually to generate the information in Table 4 and for verifying Lemmas 3 and 4. The list of coefficients in column 4 is sorted by decreasing number of combinations the coefficient produces.

| exponent set | permutation type | single-coefficient | coefficients |
|--------------|------------------|---------------------|--------------|
| \{3,3,4\}   | both             | \(7 \times 13 \times 19 \times 31 \times 37\) | 3,4,5,6,7,8,9,10,11,12,13 |
| \{4,4,3\}   | both             | \(5 \times 3^2 \times 13 \times 29 \times 37\) | 4,2,9,25,3,36,49,18,100 |

moduli are different only when two exponents are even, which when each exponent \(\leq 5\) means when two exponents = 4, which means \{4,4,3\} and \{4,4,5\}.

When based upon identical a\textsubscript{base}skipahead moduli, the generated a\textsubscript{base}skipahead tables for ax\textsubscript{minus}cz and cz\textsubscript{minus}ax permutation types for each exponent set are identical except when \(y\) is even, which when each exponent \(\leq 5\) means when \(y = 4\), which means \{4,4,3\}, \{4,4,5\}, \{3,4,5\}, \{5,4,3\}.

The a\textsubscript{base}skipahead table is overwhelmingly size-sensitive to the original moduli size, so for non-prime moduli typically the smallest helpful prime power is used (such as 9 for \{3,3,4\} and \{4,4,3\}), even though a higher power (such as 27) eliminates slightly more candidates. But the a\textsubscript{base}elimination table is not under such strict size restrictions, so it can easily use the higher power if the higher power assists. For \{3,3,4\} the a\textsubscript{base}elimination modulus 27 offers no advantage over 9, but for \{4,4,3\} cz\textsubscript{minus}ax, the a\textsubscript{base}elimination modulus 27 does eliminate a few more candidates.

The general plan is to use whatever small moduli as part of the a\textsubscript{base}elimination modulus that help eliminate a bases, then use small moduli coprime to those as part of the a\textsubscript{base}skipahead modulus. This combination of coprime moduli typically eliminates the most candidates. But some exceptions are noted: Exponent set \{4,4,3\} permutation type cz\textsubscript{minus}ax uses the modulus 29 as part of both the a\textsubscript{base}elimination modulus and the a\textsubscript{base}skipahead modulus. Instead of using 29, if the a\textsubscript{base}skipahead modulus was to use the next smallest unused perfect \(y\)-th power modulus (41) the resulting a\textsubscript{base}skipahead table would be > 4 GiB. Keeping the modulus of 29 in the a\textsubscript{base}elimination modulus provides for quicker elimination for those \(f c^2\) residues that do not have any valid a base combinations because the a\textsubscript{base}elimination table is more likely to be stored in cache. And keeping the modulus of 29 in the a\textsubscript{base}elimination modulus provides for quicker elimination for those \(f c^2\) residues that do not have any valid a bases. Similarly, \{4,4,3\} permutation type cz\textsubscript{minus}ax uses \(3^2\) as part of the a\textsubscript{base}skipahead modulus and, as mentioned in the previous item, uses \(3^3\) as part of the a\textsubscript{base}elimination modulus.

The a\textsubscript{base}elimination modulus may not exclude as many possibilities as possible. For example, for \{4,4,3\} permutation type cz\textsubscript{minus}ax, the a\textsubscript{base}elimination modulus contains a modulus of 16, even though 1024 would rule out a few more candidates. The small number of candidates that would be excluded by a modulus of 1024 that are not excluded by the modulus of 16 are not worth the overhead of the increased a\textsubscript{base}elimination table size. Instead, these few candidates that make it past the a\textsubscript{base}elimination check will be immediately rejected by the a\textsubscript{base}skipahead table.

Exponent set \{4,4,5\} permutation type cz\textsubscript{minus}ax could use the same a\textsubscript{base}skipahead...
modulus as \{4,4,5\} permutation type ax, minus cz, but the \{4,4,5\} permutation type cz, minus ax a\_base\_skipahead modulus is modified to be the product of moduli that are coprime to the exponent set’s a\_base\_elimination modulus.

Exponent set \{5,5,4\} could use the same 61 * 71 * 101 a\_base\_skipahead modulus as exponent set \{5,5,3\}, but set \{5,5,4\} takes such a short time to check all possible equations that it is not worth the extra table precalculation time.

All the a\_base\_skipahead table offsets fit within a 16-bit unsigned table entry (all offsets \leq 65535), other than some of the \{4,4,3\} single-coefficient a\_base\_skipahead tables. For some \{4,4,3\} single-coefficient tables (such as for coefficient 3, permutation cz, minus ax), the a\_base\_skipahead modulus eliminates so many a\_base candidates that sometimes the delta between valid a\_bases for a given f c\_z residue is > 65535, requiring the entire table to use 4-byte fields instead of the 2-byte fields required for all the other tables.

As mentioned in \S3.2, the a\_base\_elimination lookup table is quite small because it is simply a list of f c\_z residues that can be skipped. The list could be stored through a variety of methods (such as a hash table or binary search tree), although storing it as a simple binary array provides for quicker access. A binary array requires a number of elements equal to the modulus product even this table can become large, in which case the modulus product can be split into different moduli (into multiple lookup tables), with each table able to reduce possibilities (unlike the a\_base\_skipahead table which must be just a single table). Even though Table 5 shows only a single a\_base\_elimination modulus for each exponent set, for the purposes of code implementation, the moduli for exponent sets \{4,4,3\} and \{4,4,5\} permutation cz, minus ax were each split into three to provide three distinct a\_base\_elimination lookup tables. And the moduli for \{5,5,3\} and \{5,5,4\} were each split into two to provide two distinct a\_base\_elimination lookup tables.

Table 1 shows the percentage of c\_base and f\_coefficients combinations eliminated for all c\_bases and for f\_coefficients ranging from 2 to 100,000. The elimination percentages could have been calculated based on all possible f\_coefficients (by ranging the coefficient residue from 0 to modulus-1), but then it would be impossible to exclude coefficients that were not z\_th power free. The current table is more representative of actual searching scenarios, and the ratios between the two calculation methods differ by only a few thousandths of a percent.

### B Calculation of f\_coefficient, c\_base, and a\_base Ranges

This appendix details the logic to calculate the valid range of values for the loops in Algorithm 1 in \S3.2. Let V = the desired minimum Pegg Value. For exponent sets \{x, x, z\}, the absolute minimum for the three bases is established by \S1.6 which states that the Pegg Value can be no higher than the smallest original base to the highest exponent. For the exponent set \{3,4,5\}, Lemma 4 necessitates the use of only the three coefficient possibilities listed below (and also provides that the resultant equation will not be further multiplied by a number to re-associate min(A, B, C) to a base with a higher exponent). For these \{3,4,5\} scenarios, \S1.6 details how to determine the absolute minimum for the three bases. We summarize this information here:

If exponent set = \{x, x, z\}, \(z > x\) then

\[ a_{\text{min}} \leftarrow 1 \]
\[ b_{\text{min}} \leftarrow 1 \]
\[ c_{\text{min}} \leftarrow V \]

If exponent set = \{x, x, z\}, \(z < x\) then

\[ a_{\text{min}} \leftarrow V \]
\[ b_{\text{min}} \leftarrow V \]
\[ c_{\text{min}} \leftarrow 1 \]
else if exponent set = \{3,4,5\} then
  if coefficient is to be associated with the base to the 3rd power then
    [base to the 3rd power]_{min1} ← V/8
    [base to the 4th power]_{min1} ← V/2
    [base to the 5th power]_{min1} ← V
  else if coefficient is to be associated with the base to the 4th power then
    [base to the 3rd power]_{min1} ← V/4
    [base to the 4th power]_{min1} ← V/2
    [base to the 5th power]_{min1} ← V
  else if coefficient is to be associated with the base to the 5th power then
    [base to the 3rd power]_{min1} ← V/2
    [base to the 4th power]_{min1} ← V
    [base to the 5th power]_{min1} ← V
  end if
end if

The base minimums may be further increased by considering the permutation type and combination of other bases.

B.1 Calculating $c$ base Range

For a given $f$ coefficient, the following logic is used to determine the valid minimum and maximum $c$ bases that allow for the resultant equation to fit inside the desired range.

Let $N = \text{the smallest equation multiplier (based on } f)\text{ to convert the Original Pegg Equation to a resultant equation}

Let $S_{min} = \text{the desired minimum resultant equation size}

Let $S_{max} = \text{the desired maximum resultant equation size}

B.1.1 Permutation Type $ax - fc^z$, $c$ base Minimum

\[
ax - fc^z = b^y \\
fc^z + b^y = a^x \\
N(fc_{min2}^z + b^y) = Na^x = \text{resultant equation size} \\
N(fc_{min2}^z + b^y) \geq S_{min}
\]

This information does not increase the minimum possible value for $c$ because $b$ can always be large enough to make $N(fc_{min2}^z + b^y) \geq S_{min}$. The minimum $c$ base is $c_{min1}$ as set initially.

$$c_{min} = c_{min1}$$
B.1.2 Permutation Type ax, c base Maximum

\[
\begin{align*}
    a^x - fc^z &= b^y \\
    fc^z + b^y &= a^x \\
    N(fc^z + b^y) &= N(a^x) = \text{resultant equation size} \\
    N(fc_{max1}^z + b_{min1}^y) &\leq S_{max} \\
    fc_{max1}^z + b_{min1}^y &\leq S_{max}/N \\
    fc_{max}^z &\leq S_{max}/N - b_{min1}^y \\
    c_{max}^z &\leq (S_{max}/N - b_{min1}^y)/f \\
    c_{max} &= \left\lceil \frac{z}{\sqrt{S_{max}/(Nf)}} \right\rceil
\end{align*}
\]

B.1.3 Permutation Type cz, c base Minimum

For a given \( f \), \( c \) must be large enough so that the resulting equation will be \( \geq S_{min} \):

\[
\begin{align*}
    fc^z - a^x &= b^y \\
    fc^z &= a^x + b^y \\
    Nfc^z &= N(a^x + b^y) = \text{resultant equation size} \\
    Nfc_{min2}^z &\geq S_{min} \\
    c_{min2}^z &\geq S_{min}/(Nf) \\
    c_{min2} &= \left\lceil \frac{z}{\sqrt{S_{min}/(Nf)}} \right\rceil
\end{align*}
\]

and \( c \) must be large enough so that \( fc^z \geq \) the sum of the minimum \( a^x \) and \( b^y \):

\[
\begin{align*}
    fc^z - a^x &= b^y \\
    fc^z &= a^x + b^y \\
    fc^z &\geq a_{min1}^x + b_{min1}^y \\
    c_{min3}^z &\geq (a_{min1}^x + b_{min1}^y)/f \\
    c_{min3} &= \left\lceil \frac{z}{\sqrt{(a_{min1}^x + b_{min1}^y)/f}} \right\rceil
\end{align*}
\]

The minimum \( c \) base is the maximum of the above two minimums and the \( c_{min1} \) as set initially.

\[
    c_{min} = \max(c_{min1}, c_{min2}, c_{min3})
\]

B.1.4 Permutation Type cz, c base Maximum

\[
\begin{align*}
    fc^z - a^x &= b^y \\
    fc^z &= a^x + b^y \\
    Nfc^z &= N(a^x + b^y) = \text{resultant equation size} \\
    Nfc_{max}^z &\leq S_{max} \\
    c_{max}^z &\leq S_{max}/(Nf) \\
    c_{max} &= \left\lceil \frac{z}{\sqrt{S_{max}/(Nf)}} \right\rceil
\end{align*}
\]
B.2 Calculating \( a \) base Range

For a given \( f \) coefficient and \( c \) base, the following logic is used to determine the valid minimum and maximum \( a \) bases that allow for the resultant equation to fit inside the desired range.

Let \( N \) = the smallest equation multiplier (based on \( f \)) to convert the Original Pegg Equation to a resultant equation

Let \( S_{\text{min}} \) = the desired minimum resultant equation size

Let \( S_{\text{max}} \) = the desired maximum resultant equation size

B.2.1 Permutation Type ax_minus_cz, \( a \) base Minimum

For a given \( f \) and \( c \), \( a \) must be large enough so that the resulting equation will be \( \geq S_{\text{min}} \):

\[
\begin{align*}
a^x - fc^z &= b^y \\
a^x &= fc^z + b^y \\
Na^x &= N(fc^z + b^y) = \text{resultant equation size} \\
Na_{\text{min}2}^x &\geq S_{\text{min}} \\
a_{\text{min}2}^x &\geq S_{\text{min}}/N \\
a_{\text{min}2} &= \left\lceil \sqrt[\frac{x}{N}]{S_{\text{min}}/N} \right\rceil
\end{align*}
\]

and \( a \) must be large enough so that \( a^x \geq \) the sum of \( fc^z \) and the minimum \( b^y \):

\[
\begin{align*}
a^x - fc^z &= b^y \\
a^x &= fc^z + b^y \\
a_{\text{min}3}^x &\geq fc^z + b_{\text{min}1}^y \\
a_{\text{min}3} &= \left\lceil \sqrt[\frac{x}{N}]{fc^z + b_{\text{min}1}^y} \right\rceil
\end{align*}
\]

The minimum \( a \) base is the maximum of the above two minimums and the \( a_{\text{min}1} \) as set initially.

\[a_{\text{min}} = \max(a_{\text{min}1}, a_{\text{min}2}, a_{\text{min}3})\]

B.2.2 Permutation Type ax_minus_cz, \( a \) base Maximum

\[
\begin{align*}
a^x - fc^z &= b^y \\
a^x &= fc^z + b^y \\
Na^x &= N(fc^z + b^y) = \text{resultant equation size} \\
Na_{\text{max}}^x &\leq S_{\text{max}} \\
a_{\text{max}}^x &\leq S_{\text{max}}/N \\
a_{\text{max}} &= \left\lfloor \sqrt[\frac{x}{N}]{S_{\text{max}}/N} \right\rfloor
\end{align*}
\]
B.2.3 Permutation Type cz_minus_ax, a base Minimum

For a given \( f \) and \( c \), \( a \) must be large enough so that the resulting equation will be \( \geq S_{\text{min}} \):

\[
f c^z - a^x = b^y \\
a^x + b^y = f c^z \\
N(a^x + b^y) = N(f c^z) = \text{resultant equation size} \\
N(a^x + b^y) \geq S_{\text{min}}
\]

This information does not increase the minimum possible value for \( a \) because \( b \) can always be large enough to make \( N(a^x + b^y) \geq S_{\text{min}} \). If \( x \neq y \), then this information does not increase the minimum possible value for \( a \) because \( b \) can always be large enough to make \( a^x + b^y = f c^z \). In this case:

\[
a_{\text{min}} = a_{\text{min}1}
\]

But when \( x = y \) one of \( a^x \) and \( b^y \) must be \( \geq f c^z/2 \), and the other must be \( \leq f c^z/2 \).

As established in §3.2 for searching efficiency we label the larger value \( a^x \) and the smaller value \( b^y \). When \( x = y \), this increases the minimum \( a \) base to:

\[
a_{\text{min}2} = \lceil \sqrt[3]{\frac{f c^z}{2}} \rceil \\
a_{\text{min}} = \max(a_{\text{min}1}, a_{\text{min}2})
\]

B.2.4 Permutation Type cz_minus_ax, a base Maximum

\[
f c^z - a^x = b^y \\
a^x = f c^z - b^y \\
a_{\text{max}}^x \leq f c^z - b_{\text{min}1}^y \\
a_{\text{max}} = \lceil \sqrt[3]{f c^z - b_{\text{min}1}^y} \rceil
\]

B.3 Calculating Valid \( f \) coefficients

For the exponent set \( \{3,4,5\} \), Lemma 4 details the valid coefficients to be checked.

This section establishes the valid coefficients for exponent sets \( \{x, x, z\} \).

Per §1.6, the Pegg Value of a resultant equation can be no greater than the smallest original base to the highest exponent. We can use this information to determine the largest original coefficient that could result in an equation having a minimum desired Pegg Value. Given exponent set \( \{x, x, z\} \).

Let \( V = \text{the desired minimum Pegg Value} \)

Let \( S_{\text{max}} = \text{the desired maximum resultant equation size} \)

Let \( R = \text{the resultant coefficient as part of a resultant base to the highest exponent} \)

Let \( H = \max(x, z) \)

\[
(R_{\text{max}} V)^H \leq S_{\text{max}} \\
R_{\text{max}} V \leq \sqrt[3]{S_{\text{max}}} \\
R_{\text{max}} \leq \sqrt[3]{S_{\text{max}}/V} \\
R_{\text{max}} = \lceil \sqrt[3]{S_{\text{max}}/V} \rceil
\]

23
If \( R > R_{max} \), it would limit the original base associated with the highest exponent to a value smaller than the desired Pegg Value, thereby precluding the resultant equation from having a Pegg Value \( \geq V \). But our search algorithm does not directly concern itself with resultant coefficients. It loops through a range of original coefficients. We must answer the question “What is the highest original coefficient that would generate an \( R \leq R_{max} \)?”

Consider an Original Pegg Equation \( \{x, x, z\} \) with a single coefficient \( f \) associated with the base to the \( z \)-th power. With \( \{x, x, z\} \), \( z < x \), the two bases with the highest exponent will have a resultant coefficient that is based solely on the multiplier \( N \). With \( \{x, x, z\} \), \( z > x \), the sole base with the highest exponent will have a resultant coefficient that is based on the product of \( f \) and the multiplier \( N \).

Define the function \( \text{cvt}(x, z, v_p(f)) \) for \( x \geq 3; z \geq 3; 0 < v_p(f) < z \)

if \( z < x \), \( \text{cvt} = \text{smallest}_q(v_p(f))/\text{max}(x, z) \)

if \( z > x \), \( \text{cvt} = (\text{smallest}_q(v_p(f)) + v_p(f))/\text{max}(x, z) \)

using the function:

\[ \text{smallest}_q(v_p(f)) = \text{smallest}_q \text{ where } q \equiv 0 (\text{mod } x), q \equiv -v_p(f) (\text{mod } z) \]

For a \( p \mid f \), \( \text{cvt}(x, z, v_p(f)) \) returns the power of \( p \) as represented in the resultant coefficient of the base(s) to the highest power.

Table 7: Power of \( p \) as represented in \( R \) (the resultant coefficient of the base(s) to the highest power). Output of \( \text{cvt} \) function for all exponent sets \( \{x, x, z\} \) with each exponent \( \leq 5 \). A subset of the information in Table 2.

| Exponent set | Original Coefficient associated with base to this exponent | Highest exponent | 1 | 2 | 3 | 4 |
|--------------|------------------------------------------------------------|------------------|---|---|---|---|
| \{4,4,3\}   | 3                                                          | 4                | 2 | 1 |   |   |
| \{5,5,3\}   | 3                                                          | 5                | 1 | 2 |   |   |
| \{3,3,4\}   | 4                                                          | 4                | 1 | 2 | 3 |   |
| \{5,5,4\}   | 4                                                          | 5                | 3 | 2 | 1 |   |
| \{3,3,5\}   | 5                                                          | 5                | 2 | 1 | 3 | 2 |
| \{4,4,5\}   | 5                                                          | 5                | 1 | 2 | 3 | 4 |

With the \( \text{cvt} \) function to show how a prime dividing an original coefficient is represented in the resultant coefficient of the base(s) to the highest exponent, we can answer the question “What is the highest original coefficient that would generate an \( R \leq R_{max} \)?” The highest possible original coefficient must be \( \leq R_{max} \) where \( T = \text{maximum ratio of } v_p(f) \text{ to } \text{cvt}(x, z, v_p(f)) \).

Consider the exponent set \( \{5,5,4\} \). For \( p \mid f \) with \( v_p(f) = 3 \), the entire equation must be multiplied by \( p^3 \) (as shown in Table 1). This is represented as part of the resultant coefficient of the bases to the highest power as \( p^{3/2} \) and the \( \text{cvt} \) function shows this with the return value of 1 (representing \( \log_p(p^{3/2}) \)). So with \( v_p(f) = 3 \), an original coefficient of \( p^3 \) generates an \( R \) of \( p \). This 3:1 power ratio is the highest ratio for the \( \{5,5,4\} \) exponent set. To produce an \( R \leq R_{max} \), the original coefficient of a \( \{5,5,4\} \) equation must be \( \leq R_{max}^3 \).

For \( \{x, x, z\} \) with all exponents \( \leq 5 \), \( \{5,5,4\} \) has a highest ratio of 3, \( \{4,4,3\} \) and \( \{3,3,5\} \) each have a highest ratio of 2, and the remaining exponent sets \( \{5,5,3\}, \{3,3,4\}, \{4,4,5\} \) each have a highest ratio of 1.

At this point, the list of possible original coefficients is all integers \( \geq 2 \) and \( \leq 24 \).
$R_{\text{max}}^T$. The strategy is to create a “valid original coefficient” boolean array of these integers, initially marking all the entries as valid. Then we will perform several steps, each time possibly ruling out candidates.

Step 1: Per §1.2 mark as invalid any coefficients that are not $z$-th power free.

Step 2: Mark as invalid any coefficients that have a smallest multiplier $N$ that is sufficiently large that it precludes an original equation that has both:

- a base to the highest exponent that is $\geq$ the desired Pegg Value
- that converts to a resultant equation $\leq$ the maximum equation size

For example, exponent set \{3,3,5\} requires a coefficient with $v_p(f) = 2$ to have the entire equation multiplied by $p^3$, but a coefficient with $v_p(f) = 1, 3, \text{ or } 4$ requires a larger multiplier. Consider the search for Pegg Values $> 63742$ in \{3,3,5\} equations $\leq 2^{88}$ The highest possible resultant coefficient is 3, as

$$R_{\text{max}} = \left\lfloor \sqrt[3]{S_{\text{max}}/V} \right\rfloor$$
$$R_{\text{max}} = \left\lfloor 2^{288}/63743 \right\rfloor$$
$$R_{\text{max}} = 3$$

Let $T = 2 = \text{the maximum } \{3,3,5\} \text{ ratio of } v_p(f) \text{ to } \text{cvt}(x, z, v_p(f))$. The maximum original coefficient is given by:

$$O_{\text{max}} \leq R_{\text{max}}^T$$
$$O_{\text{max}} \leq 3^2$$
$$O_{\text{max}} \leq 9$$

Examining each possible coefficient $\geq 2$ and $\leq 9$, only the coefficients 4 and 9 can product a resultant equation $\leq 2^{88}$. Step 2 marks as invalid the coefficients 2, 3, 5, 6, 7, 8.

Step 3: for each coefficient still marked as valid, for the given permutation type, calculate the minimum and maximum possible $c$ bases that could result in an equation with a desired Pegg Value and within the resultant equation range. If a coefficient produces a minimum $c$ base $>$ its maximum $c$ base, the coefficient is marked as invalid. For example, consider the search for Pegg Values $> 63742$ in \{5,5,3\} equations $\leq 2^{100}$ of permutation type cz_minus_ax. The highest possible resultant coefficient is 16, as

$$R_{\text{max}} = \left\lfloor \sqrt[3]{S_{\text{max}}/V} \right\rfloor$$
$$R_{\text{max}} = \left\lfloor \sqrt[3]{2^{100}/63743} \right\rfloor$$
$$R_{\text{max}} = 16$$

The coefficient 16 is marked as invalid in Step 1 because it is not $z$-th power free.

For the original coefficient 15, the minimum $c$ base is given by §B.1.3

$$c_{\text{min}} = c_{\text{min3}} = \left\lfloor \sqrt[3]{(a_{\text{min}1}^2 + b_{\text{min}1})/f} \right\rfloor$$
$$c_{\text{min}} = c_{\text{min3}} = \sqrt[3]{(63743^5 + 63743^5)/15}$$
$$c_{\text{min}} = c_{\text{min3}} = 51963742$$

25
For the original coefficient 15, the maximum $c$ base is given by \[c_{\text{max}} = \lfloor \frac{z \sqrt{S_{\text{max}}}}{(Nf)} \rfloor\]

\[c_{\text{max}} = \lfloor \frac{3 \sqrt{2^{100}}}{(3^5 \times 5^5) \times (3 \times 5)} \rfloor \]

\[c_{\text{max}} = 48100619\]

So coefficient 15 is excluded as it has no valid $c$ bases. All the other coefficients that passed the tests in step 1 and step 2 are valid, as each coefficient has a minimum $c$ base $\leq$ its maximum $c$ base.

After step 3, all the coefficients still marked as valid are to be checked.

References

[1] Nils Bruin, On Powers as sums of two cubes, Proceedings of the 4th International Symposium on Algorithmic Number Theory 1838 (2000), 169–184.
[2] Richard E. Crandall, Carl Pomerance, Prime Numbers: A Computational Perspective (2nd ed.), Springer-Verlag, New York, 2005, pp. 417.
[3] Noam D. Elkies, The ABC’s of Number Theory, Harvard College Mathematics Review 1 (Spring 2007), 57–766.
[4] Michel Waldschmidt, Open Diophantine Problems, Moscow Mathematics Journal, 4 (January–March 2004), 245–305.
[5] R. Daniel Mauldin, A Generalization of Fermat’s Last Theorem: The Beal Conjecture and Prize Problem, AMS Notices 44 (December 1997), 1436–1437.
[6] Ed Pegg Jr., material added 12 May 2001, http://www.mathpuzzle.com/Aug52001.htm (29 August 2009)
[7] P. Mihăilescu, Primary Cyclotomic Units and a Proof of Catalan’s Conjecture, J. reine angew. Math. 572 (2004), 167–195.
[8] Henri Darmon, Andrew Granville, On the equations $z^m = F(x, y)$ and $Ax^p + By^q = Cz^r$, Bull. London Math. Soc. 27 (1995), 513–543.
[9] University of Leiden, ABC@Home, http://www.abcathome.com/ (29 August 2009)
[10] Torbjörn Granlund et al., 16.5.3 Perfect Square, The GNU Multiple Precision Arithmetic Library, Edition 4.3.0, http://gmplib.org/manual/Perfect-Square-Algorithm.html (29 August 2009)
[11] Advanced Micro Devices, Inc., Software Optimization Guide for AMD64 Processors, Publication 25112, Rev. 3.06, September 2005, http://support.amd.com/us/Processor_TechDocs/25112.PDF (29 August 2009), 8.1 Replacing Division with Multiplication
[12] Derrick Coetzee, Integer division by constants, Published Monday, December 12, 2005 5:07 PM, http://blogs.msdn.com/devdev/archive/2005/12/12/502980.aspx (29 August 2009).
[13] G.M.Hardy, E.M.Wright, An Introduction to the Theory of Numbers (5th ed), Oxford University Press, New York, 1989 [1938], pp. 269, Theorem 332.
[14] Frits Beukers, The generalized Fermat equation 20 January 2006, http://www.math.uu.nl/people/beukers/Fermatlectures.pdf (29 August 2009)
[15] Bjorn Poonen, Edward F. Schaefer, Michael Stoll, Twists of X(7) and Primitive solutions to $x^2 + y^3 = z^2$, Duke Math. J. 137 (2007), 103–158.
[16] Johnny Edwards, A complete solution to $X^2 + Y^3 + Z^5 = 0$, J. Reine Angew. Math. 571 (2004), 213–236.
[17] Dario Alpern, Sum of powers, http://www.alpertron.com.ar/SUMPOWER.HTM (29 August 2009)
Andrew Wiles, Modular elliptic curves and Fermat’s last theorem, *Ann. of Math.* **141** (1995), 443–551.

Richard Taylor and Andrew Wiles, Ring-theoretic properties of certain Hecke algebras, *Ann. Of Math.* **141** (1995), 553–572.

Nils Bruin, Chabauty methods using elliptic curves, *J. Reine Angew. Math.* **562** (2003), 27–49.

Bjorn Poonen, Some Diophantine equations of the form $x^n + y^n = z^m$, *Acta Arith.* **86** (1998), 193–205.

Henri Cohen, *Number Theory Volume II: Analytic and Modern Tools*, Springer Science Business Media, New York, 2007, pp. 470.

The NETBSD Project, [http://www.netbsd.org](http://www.netbsd.org) (29 August 2009)

Free Software Foundation, [http://gcc.gnu.org](http://gcc.gnu.org) (29 August 2009)

Torbjörn Granlund et al., [http://www.gmplib.org/](http://www.gmplib.org/) (29 August 2009)