# Provably Complete
# Rank 2 Temperament Searches

## Graham Breed

## March 10, 2008

## Contents

In *Primerr*[1] I described ways to calculate the error and complexity of different regular temperament classes. It's more difficult to find temperament classes with a low error and complexity. Here I describe a method guaranteed to find all rank 2 classes within a given error and complexity.

---

[1]"Prime Based Error and Complexity Measures", completed February 7, 2008. Available at `http://x31eq.com/primerr.pdf`. I'll call this *Primerr* throughout for the sake of brevity.

This is a refinement of the method I explained on the *Yahoo! Groups* mailing list `tuning-math` on 20 and 31 January, 2006. Hopefully it'll be clearer in PDF because I can format the equations.

## 1 Problem Definition

The key to this method is that the square error of a rank 2 temperament is defined to be a quadratic function of the generator size, as in Equation 55 on page 14 of *Primerr*.

$$E^2 = \sigma_{M_1}^2 (g - g_{\text{opt}})^2 + E_{\text{opt}}^2 \tag{1}$$

where $\sigma_{M_1}^2$ is the variance of the generator mapping, $g$ is the generator size, $g_{\text{opt}}$ is the optimal tuning of the generator, and $E_{\text{opt}}$ is the optimal error given by

$$E_{\text{opt}}^2 = \frac{\sigma_{M_0}^2 \sigma_{M_1}^2 - \sigma_{M_0 M_1}^2}{M_{00}^2 \, \sigma_{M_1}^2} \tag{2}$$

$M_{00}$ is the number of periods to an octave and $M_1$ is the weighted generator mapping (including a zero).

The corresponding choice for complexity is Equation 69 on page 17 of *Primerr*.

$$k(M) = M_{00}\sigma_{M_1} \tag{3}$$

The goal is to find all rank 2 temperament classes with $E \leq E_{\text{max}}$ and $k \leq k_{\text{max}}$ for given $E_{\text{max}}$ and $k_{\text{max}}$.

## 2 Equal Temperament Constraints

Good equal temperaments are fairly easy to find. A simple method is to look at each number of steps to the octave, and take the nearest approximation to each prime. In practice, this isn't guaranteed to give the best equal temperaments. However, the reliable algorithm doesn't take much longer. I won't explain it today.

Similarly, once you have an equal temperament, finding the best rank 2 temperaments that belong to it is fairly easy. You choose each number of steps to the generator and find the best mapping.[2] In practice, the algorithm for this is similar to the one for finding equal temperaments, and I still won't explain it today.

In Equation 1, $g$ can be any real number. For an equal temperament, it's always a rational number because it's the ratio of the number of steps to a generator to the number of steps to an octave. So the error equation becomes

$$E^2 = \sigma_{M_1}^2 \left(\frac{n}{d} - g_{\text{opt}}\right)^2 + E_{\text{opt}}^2 \qquad (4)$$

Let's choose $d_{\text{max}}$ as the highest number of notes to the octave we want to deal with. Then, we know that the equal temperaments we want to look at obey the following theorem.

**Theorem 1** *All rank 2 temperament classes with complexity $k \leq k_{\text{max}}$ and STD error[3] $E \leq E_{\text{max}}$ have a tuning in the equal temperaments with $d$ notes to the octave, $d \leq d_{\text{max}}$, and STD error $E$ such that*

$$E^2 \leq \frac{k_{\text{max}}^2}{d^2(d_{\text{max}} + 1)^2} + E_{\text{max}}^2 \qquad (5)$$

*provided $d_{\text{max}}$ is sufficiently large.*

To prove Theorem 1, consider that the overall error of a temperament depends on how close $g$ gets to $g_{\text{opt}}$. From number theory, we know that where

the period equals the octave, an $n$ and $d$ must exist where $0 < d \leq d_{\text{max}}$ and[4]

$$\left| g_{\text{opt}} - \frac{n}{d} \right| \leq \frac{1}{d(d_{\text{max}} + 1)} \qquad (6)$$

For a general rank 2 temperament, that describes the ratio of the number of steps to a generator to the number of steps to the period. To make $d$ the number of steps to an octave, we need to adjust by the number of periods to the octave $M_{00}$. To do this, multiply $g_{\text{opt}}$ by $M_{00}$ and divide $d$ and $d_{\text{max}}$ by $M_{00}$.

$$\left| M_{00}g_{\text{opt}} - \frac{nM_{00}}{d} \right| \leq \frac{M_{00}}{d\left(\frac{d_{\text{max}}}{M_{00}} + 1\right)} \qquad (7)$$

Because $M_{00}$ is defined to be positive, that becomes

$$M_{00} \left| g_{\text{opt}} - \frac{n}{d} \right| \leq \frac{M_{00}^2}{d(d_{\text{max}} + M_{00})} \qquad (8)$$

and because $M_{00}$ is never zero,

$$\left| g_{\text{opt}} - \frac{n}{d} \right| \leq \frac{M_{00}}{d(d_{\text{max}} + M_{00})} \qquad (9)$$

This requires $d_{\text{max}}$ to be at least the largest number of periods to the octave we might be interested in.

It follows that

$$\left( g_{\text{opt}} - \frac{n}{d} \right)^2 \leq \frac{M_{00}^2}{d^2(d_{\text{max}} + M_{00})^2} \qquad (10)$$

Substituting into Equation 4 and square rooting everything,

$$\sqrt{E^2 - E_{\text{opt}}^2} \leq \frac{M_{00}\sigma_{M_1}}{d(d_{\text{max}} + M_{00})} \qquad (11)$$

The numerator is the complexity as defined in Equation 3. So,

$$\sqrt{E^2 - E_{\text{opt}}^2} \leq \frac{k(M)}{d(d_{\text{max}} + M_{00})} \qquad (12)$$

For the rank 2 temperaments we're looking for, $k(M) \leq k_{\text{max}}$. That means replacing $k(M)$ with $k_{\text{max}}$ can only make the right hand side larger, and so

$$\sqrt{E^2 - E_{\text{opt}}^2} \leq \frac{k_{\text{max}}}{d(d_{\text{max}} + M_{00})} \qquad (13)$$

---

[2]Gene Ward Smith talked about this general method on `tuning-math` long before I implemented it.

[3]"Optimal Tenney-weighted STD Error" is a real mouthful. As I want to talk about it a lot today I'm shortening it to "STD error".

[4]See Yu. I. Manin & A. A. Panchishkin, *Introduction to Modern Number Theory*, Second Edition. The field is called "Diophantine approximations". The reference doesn't explicitly exclude the case where $n = d = 0$ but as $\frac{n}{d}$ should be in its lowest terms I hope division by zero always fails that test.

Rearrange that a bit:

$$E^2 \le \left[ \frac{k_{\max}}{d(d_{\max} + M_{00})} \right]^2 + E_{\text{opt}}^2 \qquad (14)$$

Here, $E_{\text{opt}}$ is the smallest possible error for the rank 2 temperament we're looking at. If that's one of the temperaments we're looking for, $E_{\text{opt}} \le E_{\max}$. That means replacing $E_{\text{opt}}$ with $E_{\max}$ can only make the right hand side larger, and

$$E^2 \le \frac{k_{\max}^2}{d^2(d_{\max} + M_{00})^2} + E_{\max}^2 \qquad (15)$$

This is the function we're looking for but with the extra factor of $M_{00}$, the number of periods to the octave. To simplify it, note that there's always at least one period to the octave and replacing $M_{00}$ with one makes the denominator smaller and so the right hand side larger. That leaves us with Theorem 1.

## 3 The Largest Equal Temperament

So far, there's an arbitrary term $d_{\max}$ equal to the largest number of steps to the octave in any equal temperament we want to look at. The only constraint is that it has to be at least as large as the largest number of periods to the octave we expect.

Common sense tells us that you need at least two notes to a period for a rank 2 temperament to make any sense. That means the highest number of periods per octave is half the largest number of notes you might want to use. As long as $d_{\max}$ is at least as large is the size of scale you're looking for you shouldn't have to worry about missing temperaments with a high $M_{00}$. However, I'm talking about proofs today, not common sense, so I'll give you a quantitative constraint on $d_{\max}$.

**Theorem 2** *When searching for temperament classes with $n_p$ prime intervals and buoyancies of $b$, Theorem 1 holds for*

$$d_{\max} \ge \text{floor} \left[ \frac{k_{\max} n_p}{\sqrt{n_p - 1}} \frac{\max(b)}{b_0} \right] \qquad (16)$$

The proof is in Appendix B.

That gives us a lower limit for something we didn't expect to get too low anyway. To make the search as efficient as possible it's also nice to know the optimal value for $d_{\max}$. Here's a simple rule to find it:

**Rule of Thumb 1** *It's easiest to find equal temperaments according to Theorem 1 when*

$$d_{\max} = \text{floor} \left[ \sqrt{\frac{k_{\max}}{E_{\max}}} \right] \qquad (17)$$

To demonstrate its validity, turn Theorem 1 into a badness inequality by multiplying both sides by the square of the number of steps to the octave.

$$B^2 \le \frac{k_{\max}^2}{(d_{\max} + 1)^2} + d^2 E_{\max}^2 \qquad (18)$$

For a given badness of this form, it's about as difficult to find an equal temperament below the threshold regardless of the number of steps to the octave. The highest value of the badness is where $d = d_{\max}$. That is,

$$B_{\max} = \frac{k_{\max}^2}{(d_{\max} + 1)^2} + d_{\max}^2 E_{\max}^2 \qquad (19)$$

The $d_{\max} + 1$ term is fiddly, so to get a rough idea of the best value replace it with $d_{\max}$.

$$B_{\max} \simeq \frac{k_{\max}^2}{d_{\max}^2} + d_{\max}^2 E_{\max}^2 \qquad (20)$$

The search will be most efficient when this takes its lowest possible value. To find that, differentiate with respect to $d_{\max}$ and set the result to zero.[5]

$$\frac{\mathrm{d}(B_{\max})}{\mathrm{d}(d_{\max})} \simeq 0$$

$$-2 \frac{k_{\max}^2}{d_{\max}^3} + 2 d_{\max} E_{\max}^2 \simeq 0$$

$$d_{\max}^4 E_{\max}^2 - k_{\max}^2 \simeq 0$$

$$d_{\max} \simeq \sqrt{\frac{k_{\max}}{E_{\max}}} \qquad (21)$$

---

[5] Note the similarity with Equation 89 on page 19 of *Primerr*. Now you know what it means; it indicates the region where you're most likely to find equal tunings of the temperament class with low badness.

As this is a compromise between $d_{\max}$ and $d_{\max} + 1$, it's safe to round it down. Hence Rule of Thumb 1. It's is a good bet for $d_{\max}$ unless it violates Theorem 2. So the best thing is to take the larger of the two.

**Rule of Thumb 2** *When applying Theorem 1, a good and safe value for $d_{\max}$ is*

$$d_{\max} = \text{floor}\left[\max\left(\sqrt{\frac{k_{\max}}{E_{\max}}}, \frac{k_{\max}n_p}{\sqrt{n_p - 1}}\frac{\max(b)}{b_0}\right)\right] \tag{22}$$

Given this rule, you could re-apply Theorem 1 for each equal temperament you find, setting $d_{\max}$ to one less than the number of notes to the octave in the equal temperament you found first. You can then use the formulae in *Primerr* that describe rank 2 temperament classes in terms of a pair of equal temperaments. This works, but only if Theorem 2 still applies. In general, you can't be sure if the equal temperaments that come out the first time will be big enough so you still need to implement the code to find the rank 2 temperament classes from a single equal temperament.

## 4 Examples

### 4.1 Miracle

Miracle temperament caused a stir when it was rediscovered back in 2001. It looked like the optimal rank 2 temperament class for an important range of error and complexity. But could we be sure we weren't missing something better? Now we can prove we weren't.

From *Primerr*, notice that miracle has an 11-limit STD error of $0.484$ cents per octave (cpo) and an STD complexity of $2.788$. The optimal value for $d_{\max}$ is

$$d_{\max} = \text{floor}\left[\sqrt{\frac{2.788 \times 1200}{0.484}}\right] = 83 \tag{23}$$

The equal temperament search gives one result:

$$\langle 72, 114, 167, 202, 249|$$

To find another equal temperament to pair with this, repeat the search with $d_{\max} = 71$. That also gives a single result:

$$\langle 31, 49, 72, 87, 107|$$

That means any linear temperament that does at least as well as miracle must look exactly like miracle. As the largest octave division compatible with the complexity is 24, this is generally true for all rank 2 temperaments.

As it happens, running the rank 2 search returns no results. That's because the figures for error and complexity both happen to be rounded down. Change the error to $2.7881$ and the complexity to $0.4841$ cents per octave and the same equal temperaments are produced, along with a rank 2 mapping of

$$|\langle 1, 1, 3, 3, 2|, \langle 0, 6, -7, -2, 15|\rangle$$

This matches the 11-limit mapping for miracle in *Primerr*.

That shows miracle's better at being miracle than any other temperament is. But maybe there's a temperament that's a bit simpler but has almost the same error. Making the complexity limit $2.788$ and the error limit $1.0$ cpo gives a single result:

$$|\langle 1, -1, 0, 1, -3|, \langle 0, 10, 9, 7, 25|\rangle$$

This is called myna in *Primerr*, where it's given an STD complexity of $2.609$ and an STD error of $0.852$ cpo. Hence a search with complexity of $2.7881$ and error of $0.851$ cpo will still only return miracle. Requiring the error to almost double means miracle really does stand out.

Well, how about more complex temperaments with a smaller error than miracle? The only one listed in *Primerr* is wizard with an STD error of $0.448$ cpo and an STD complexity of $4.063$. The mapping is

$$|\langle 2, 1, 5, 2, 8|, \langle 0, 6, -1, 10, -3|\rangle$$

A search with $k_{\max} = 4.064$ and $E_{\max} = 0.484$ cpo gives only wizard. A search with $k_{\max} = 4.063$ and $E_{\max} = 0.4841$ cpo gives only miracle. Hence wizard is the temperament class with the next lowest complexity to miracle and a smaller optimal error. That means a rank 2 temperament has to get a fair bit more complex than miracle to have a simpler error.

That covers temperaments which are either simpler or more accurate than miracle. How about those which are close to miracle, but not quite as good? Doing a broad search with $k_{\max} = 4.063$ and $E_{\max} = 0.851$ cpo gives seven results. Narrowing it down to $k_{\max} = 3.75$ and $E_{\max} = 0.65$ cpo and only miracle comes out. (I chose these values because the optimal $d_{\max}$ is 83, as for the narrowest miracle search, so the trade-off between complexity and error is about the same.) Miracle has a scalar badness[6] of $2.788 \times 0.484 = 1.3494$. There are no temperaments nearby with a badness of less than $3.75 * 0.65 = 2.4735$ so miracle is nearly twice as good as anything similar. Truly outstanding!

Increase the complexity and error limits to $3.78$ and $0.68$ cpo and two new temperament classes pop out. Their mappings are

$$|\langle 1, 1, -1, 3, 6| , \langle 0, 3, 17, -1, -13|\rangle$$

$$|\langle 1, -5, 0, -3, -7| , \langle 0, 17, 6, 15, 27|\rangle$$

The first is in *Primerr* as rodan. It has an STD complexity of $3.640$ and an STD error of $0.671$ cpo. The other can be called an extension of semisept[7]. Its STD complexity is $3.773$ and its STD error is $0.614$ cpo. Its generator is 12 steps from a 31 note scale. In so far as STD error and complexity mean anything, these are miracle's nearest neighbors.

## 4.2 Ennealimmal

Standard deviations involve subtracting the squares of numbers that are very close together. Because of this they can suffer from rounding errors. Ennealimmal[8] temperament can get very close to 7-limit just intonation, so let's check that it comes up when it's supposed to. The canonical mapping is

$$|\langle 9, 15, 22, 26| , \langle 0, -2, -3, -2|\rangle$$

*Primerr* gives it an STD error of 0.030 cpo and an STD error of 4.724. Sure enough, it comes up in a

search with $k_{\max} = 4.724$ and $E_{\max} = 0.03$ cpo, but not when $E_{\max}$ is reduced to 0.029 cpo.

So, is ennealimmal all it's cracked up to be? Isn't there something simpler that does nearly as well? You can keep $k_{\max}$ at $4.724$ and increase $E_{\max}$ to $0.22$ cpo and it's still the only result you get. That makes the error seven times as large! So yes, ennealimmal is very special indeed.

## 4.3 Mystery

Mystery is the name I give to a temperament class that sits at the top of some of my searches for the odd limits of 13 and 15. It divides the octave into 29 equal parts and the whole 15-limit is contained in the 58 note MOS. The mapping is

$$|\langle 29, 46, 68, 82, 101, 108| , \langle 0, 0, -1, -1, -1, -1|\rangle$$

It's STD error is $0.513$ cpo and its STD complexity is $4.837$.

Sure enough, mystery comes up in a search with $E_{\max} = 0.513$ cpo and $k_{\max} = 4.837$. However, so does a kind of miracle:

$$|\langle 1, 1, 3, 3, 2, 7| , \langle 0, 6, -7, -2, 15, -34|\rangle$$

It has an STD complexity of $4.534$ and an STD error of $0.478$ cpo. So, mystery isn't a stand-out with STD measures.

Generally, mystery comes up better with odd-limit minimax measures than mean-based measures. Its virtue is that none of the 15-limit intervals require more than one generator step, and so they all fit into that 58 note MOS with plenty of modulation by fifths. However, all the intervals that require that generator step have a relatively high complexity (not equally high with weighting). Counting intervals like $25 : 16$ that lie outside the 58 note MOS isn't really to the point, but this is what the STD complexity implies. In these terms mystery is still good but not the best.

Widen the parameters to $k_{\max} = 4.9$ and $E_{\max} = 0.52$ cpo and another temperament class comes out:

$$|\langle 2, 4, 7, 7, 9, 11| , \langle 0, -6, -17, -10, -15, -26|\rangle$$

---

[6] Complexity times optimal error with error in cpo, whatever units that works out as.
[7] Name from Gene Ward Smith, collected by Herman Miller, `tuning-math` March 3, 2008
[8] Discovered and named by Gene Ward Smith

It's related to miracle and you can call it harry[9]. The generator is 4 steps of a 58 note scale (with a 29 note period). Its STD complexity is $4.889$ and its STD error is $0.354$ cpo.

## 4.4 Over Broad Searches

Say that you're looking for temperaments in the vicinity of miracle. So you decide to do a search with $k_{\max} = 4$ and $E_{\max} = 0.5$ cpo. However, you forget the cpo bit and instead set a dimensionless (octaves per octave) $E_{\max} = 0.5$. This is a stupidly high error and there are loads and loads of rank 2 classes that can beat it with a complexity of within $4$.

The first eleven equal temperaments that come out all have a mapping that starts $\langle 1, 0, -1, \ldots |$. So what sense does this make? One note equal temperament? Sometimes a valid search may require such monstrosities. If you get too many of them it's a sign that there are so many rank 2 temperaments of the kind you asked for that you'll never be able to look at them all, so you should refine your search.

I implemented these searches in Python. With some older versions this kind of thing would hang the interpreter in a way that made it difficult to interrupt. Fortunately that doesn't seem to be the case any more. And, using these new fangled generators, it's possible to see the equal temperaments that come out without waiting for the full set. I didn't wait long enough to get a single rank 2 result.

Generally speaking, narrow searches are fast. All the examples in this section take up less than half a second on my bottom of the range laptop. So the execution time isn't really a problem. But it may become one if you want to do a general search and take the best results. This method isn't really suited for that.

## 5 General Applicability

Theorem 1 is fine if you want to search for temperament classes according to STD complexity and error. But what about the more likely case that you're interested in TOP-RMS error and scalar complexities

---

(*Primerr* Sections 2 and 5.5)?

$$E_{\mathrm{RMS}}^2 = \frac{\sigma_{M_0}^2 \sigma_{M_1}^2 - \sigma_{M_0 M_1}^2}{\langle M_0^2 \rangle \langle M_1^2 \rangle - \langle M_0 M_1 \rangle^2} \tag{24}$$

$$k_{\mathrm{scalar}}^2 = \langle M_0^2 \rangle \langle M_1^2 \rangle - \langle M_0 M_1 \rangle^2 \tag{25}$$

To find how closely they match, consider Equation 26 on p. 8 of *Primerr*.

$$E_{\mathrm{RMS}} = \sqrt{\frac{\langle w^2 \rangle - \langle w \rangle^2}{\langle w^2 \rangle}} \tag{26}$$

The numerator here is the STD error. The denominator is the RMS of the weighted primes, which can also be written in terms of weighted errors.

$$\sqrt{\langle w^2 \rangle} = \sqrt{1 + 2\langle e \rangle + \langle e^2 \rangle} \tag{27}$$

Hence Equation 26 is the same as

$$E_{\mathrm{RMS}} = \frac{E_{\mathrm{STD}}}{\sqrt{1 + 2\langle e \rangle + \langle e^2 \rangle}} \tag{28}$$

where $E_{\mathrm{RMS}}$ is the TOP-RMS error, $E_{\mathrm{STD}}$ is the STD error (optimal and Tenney weighted) and plain $e$ is the list of errors for some stretched TOP-RMS tuning. Alternatively,

$$\frac{E_{\mathrm{STD}}}{E_{\mathrm{RMS}}} = \sqrt{1 + 2\langle e \rangle + \langle e \rangle^2} \tag{29}$$

Where the errors are small, a binomial expansion of that gives

$$\frac{E_{\mathrm{STD}}}{E_{\mathrm{RMS}}} = 1 + \langle e \rangle + \frac{1}{2}\langle e^2 \rangle + O\left(\langle e \rangle^2\right) + O\left(\langle e^2 \rangle^2\right) \tag{30}$$

Whether the mean or RMS error dominates, the STD error is the first order approximation to the TOP-RMS error. That leads to a general rule:

**Rule of Thumb 3** *When searching for rank 2 temperament classes with TOP-RMS error less than $E_{\mathrm{RMS}}$ and half-range complexity less than $k_{\mathrm{scalar}}$, Theorem 1 applies if you set*

$$E_{\max} = (1 + \lambda E_{\mathrm{RMS}}) E_{\mathrm{RMS}} \tag{31}$$

$$k_{\max} = (1 + \lambda E_{\mathrm{RMS}}) k_{\mathrm{scalar}} \tag{32}$$

*where $\lambda$ is a confidence factor of around 1. The higher $\lambda$, the less likely you are to miss something.*

The logic is that you don't know the sign of $\langle e \rangle$ but you know it's about the magnitude of the optimal error, and so the two ways of measuring the error must agree to a factor proportional to the optimal error.

As I said in Section 5.5 of *Primerr*, the STD error multiplied by the STD complexity is the same as the TOP-RMS error multiplied by the scalar complexity. So, the two kinds of complexity will approximate each other to the same degree as the two kinds of errors.

Another popular error measure is the TOP-max, which is the smallest value of the magnitude of the worst Tenney-weighted error of the prime intervals (or some other set of intervals). A natural counterpart to such an error is the half-range complexity (*Primerr* Equation 72, p. 17).

$$k(M) = M_{00} \frac{\max(M_1) - \min(M_1)}{2} \qquad (33)$$

For these, the rule is very simple.

**Rule of Thumb 4** *To find rank 2 temperament classes with TOP-max error below $E_{\max}$ and half-range complexity below $k_{\max}$, apply Theorem 1 directly. The bound on the error of equal temperaments still refers to the STD error.*

STD complexity is the standard deviation of the same thing that the half-range complexity is half the range of. The STD is the RMS relative to the mean, and the half-range is the worst case relative to the middle of the range. Because averages are always smaller than worst cases, the STD complexity can't be bigger than the half-range complexity. So Theorem 1 still applies if you substitute one complexity for another.

Showing that the TOP-max error is always greater than the STD error is more difficult. The TOP-max error is the worst case for a particular scale stretch. Hence this is only a rule of thumb and not a theorem.

## 6 Conclusion

Searches according to Theorem 1 and Rule of Thumb 2 are a safe and efficient way to return all rank 2 temperament classes within given error and complexity bounds, provided the bounds are strict enough that only a few temperament classes qualify.

This is most valuable when you already know of some interesting temperament classes, and want to find out if other classes have similar properties. If you don't know of any temperament classes that might be returned, it's harder to make sure the criteria are generous enough to return something but strict enough to return within a reasonable amount of time.

The proof involves optimal Tenney-weighted STD errors and the corresponding complexities, but can be extended to other cases.

## A Python Code

To clarify how the algorithm for finding rank 2 temperaments is supposed to work, here's some source code in the Python programming language.[10] As it uses the *set* type, it requires at least version 2.4. The full code is in the file `complete.py` in the bundle that can be downloaded from `http://x31eq.com/temper/regular.zip` and also requires `regular.py` from the same bundle.

First, use `getEqualTemperaments` to get the mappings for some equal temperaments that represent all rank 2 temperaments with a complexity less than `kmax` and an error less than `Emax`. It follows Theorem 1 and supplies a default value for the largest equal temperament according to Rule of Thumb 2. It's nice and small because it uses the function `varLimitedMappings` to do the real work. That's too complicated to show here so you'll have to download the full source code to see it.

Once you have your equal temperaments, you use `getLinearTemperaments` find the rank 2 temperaments. This depends on the function `allMappings` and the two finder classes to do the work. They're so complicated I left them in `regular.py`. There's also a function `invariant` that returns something to uniquely identify the temperament class. It isn't that complicated, but not that interesting either, so I left it out.

The function at the top, `survey`, finds your rank

---

[10] See `http://www.python.org`

Listing 1: Excerpts from `complete.py`

```python
def survey(primes, kmax, Emax):
    ets = getEqualTemperaments(primes, kmax, Emax)
    return list(getLinearTemperaments(primes, ets, kmax, Emax))

def getEqualTemperaments(primes, kmax, Emax, dmax=None):
    if dmax is None:
        dmax = int(math.sqrt(kmax/Emax))
    # Check we get all period divisions
    dmax = max(dmax, largestOctaveDivision(primes, kmax))
    k2od2 = (kmax/(dmax+1))**2
    for nNotes in xrange(1, dmax+1):
        cutoff = k2od2/nNotes**2 + Emax**2
        for mapping in varLimitedMappings(nNotes, cutoff, primes):
            yield mapping

def getLinearTemperaments(primes, ets, kmax, Emax):
    oldInvariants = set()
    for et in ets:
        mappings = allMappings(primes, et, kmax, Emax,
                STDComplexityFinder, VarianceErrorFinder)
        for gen, permap, genmap in mappings:
            key = invariant(permap, genmap)
            if key not in oldInvariants:
                oldInvariants.add(key)
                yield permap, genmap

def largestOctaveDivision(primes, kmax):
    np = len(primes)
    return int(kmax*np*max(primes) / primes[0]/math.sqrt(np-1))
```

2 temperaments given only the prime-limit and the complexity and error limits. It's a simpler interface but you never know what equal temperaments it used. It also returns a list so it's more convenient but it'll hang if your requirements are too lax.

Note that errors are measured in dimensionless units, but in the text I talk about cpo. Divide cpo by 1200 to make them dimensionless.

There's also `largestOctaveDivision` to implement Theorem 2.

## B Proof of Theorem 2

For a given $M_{00}$ the smallest equal tuning will have $M_{00}$ notes to the octave and a generator of zero steps. That is $d = M_{00}$. For this equal temperament to be covered by the search, $d_{\max} \geq M_{00}$. The complexity constraint tells us (via Equation 3)

$$M_{00}\sigma_{M_1} \leq k_{\max} \tag{34}$$

Rearranging that gives a constraint on $M_{00}$ and therefore $d_{\max}$

$$d_{\max} \geq \frac{k_{\max}}{\sigma_{M_1}} \tag{35}$$

If $d_{\max}$ satisfies this inequality, we know it's big enough. Maybe a smaller $d_{\max}$ would still work but let's be generous.

So, the smallest safe value of $d_{\max}$ corresponds to the smallest possible value of $\sigma_{M_1}$. This is the standard deviation of the weighted generator mapping. It measures how far elements of the weighted mapping get from the mean. If all elements are zero it's

invalid because it doesn't refer to a true rank 2 temperament. The smallest finite value is when all elements are zero except the one with the least weight. In terms of buoyancy,

$$\frac{b_0}{\max(b)} \tag{36}$$

where $b$ is the list of buoyancies and $b_0$ is the buoyancy of the equivalence interval[11]. For Tenney weighting of prime limits that's the largest prime in the limit.

$$\max(b) = \log_2[p(n_p)] \tag{37}$$

where $n_p$ is the number of primes, $p(i)$ is the $i$th prime number, and buoyancy is measured in octaves. The standard deviation is

$$\sigma_{M_1} = \sqrt{\langle M_1^2 \rangle - \langle M_1 \rangle^2} \tag{38}$$

where the angled brackets represent means. In this case they're very simple:

$$\langle M_1 \rangle = \frac{b_0}{\max(b)} \tag{39}$$

$$\langle M_1^2 \rangle = \frac{b_0^2}{\max(b^2)} \tag{40}$$

The standard deviation follows as

$$\begin{aligned}
\sigma_{M_1} &= \sqrt{\frac{b_0^2}{n_p \max(b^2)} - \left(\frac{b_0}{n_p \max(b)}\right)^2} \\
&= \sqrt{\frac{b_0^2}{n_p \max(b^2)} - \frac{b_0^2}{n_p^2 \max(b^2)}} \\
&= \frac{\sqrt{n_p - 1}}{n_p} \frac{b_0}{\max(b)}
\end{aligned} \tag{41}$$

Substitute that into Equation 35 to get

$$d_{\max} \geq \frac{k_{\max} n_p}{\sqrt{n_p - 1}} \frac{\max(b)}{b_0} \tag{42}$$

We can safely round this down because both $d_{\max}$ and $M_{00}$ are constrained to be integers, to give Theorem 2.

---

[11]This is a normalization factor to ensure the choice of units cancels out. Depending on how you define the weighted mapping for inharmonic timbres, you might not need it.

## C Glossary

**Canonical mapping** A standard form of the mapping for a rank 2 temperament. The first element describes the primes according to the period, and the second describes them according to the generator.

**Equal temperamant** A regular temperament that divides the octave into a number of equal parts and has no notes outside this division.

**Generator** An interval distinct from the period used to define a rank 2 temperament.

**Period** An interval that equally divides the octave.

**RMS** The root mean squared.

**Rank 2 temperament** A regular temperament with two distinct step sizes.

**Regular temperament** A tuning approximating just intonation where each just interval maps to a single tempered interval.

**Regular temperament class** The set of all regular temperaments that share the same mapping from just intonation.

**STD** The standard deviation. That is, the RMS relative to the mean.

**STD complexity** The standard deviation of the Tenney weighted generator mapping, multiplied by the number of periods to an octave.

**STD error** The standard deviation of the Tenney weighted errors of a regular temperament. When applied to a temperament class, use the optimal value, but keep octaves pure.

**Tenney weighting** Weighting intervals according to their size, with larger intervals getting the less weight.

**Weighted mapping** The numbers indicating how many steps in a temperament there are to each prime interval, weighted according to complexity. It should be normalized so the octave has a weight of 1.