Factorizing RSA Keys
An Improved Analogue Solution

Ed Blakey

Oxford University Computing Laboratory, Wolfson Building,
Parks Road, Oxford, OX1 3QD, United Kingdom
edward.blakey@queens.ox.ac.uk

Abstract. Factorization is notoriously difficult. Though the problem is not known to be **NP**-hard, neither efficient, *algorithmic* solution nor technologically practicable, *quantum-computer* solution has been found. This apparent complexity, which renders infeasible the factorization of sufficiently large values, makes secure the RSA cryptographic system.
Given the lack of a practicable factorization system from algorithmic or quantum-computing models, we ask whether efficient solution exists elsewhere; this motivates the *analogue* system presented here. The system's complexity is prohibitive of its factorizing arbitrary, natural numbers, though the problem is mitigated when factorizing $n = pq$ for primes p and q of similar size, and hence when factorizing *RSA keys*.
Ultimately, though, we argue that the system's polynomial time and space complexities are testament not to its power, but to the inadequacy of traditional, Turing-machine-based complexity theory; we propose *precision complexity* (defined in [3]) as a more relevant measure.
Keywords: FACTORIZATION; ANALOGUE; COMPLEXITY; CRYPTOGRAPHY.

1 Introduction

1.1 What is Computation?

We discuss factorization in the context of non-standard (that is, non-Turing-machine) computation. Given the prevalence of the digital computer—an implementation of the standard computational model—, it may be beneficial for the reader to consider the preliminary question: *what is computation?*

It is convenient to think of *computation* as 'that which is performed by a computer', though there is a crucial caveat: we must interpret 'computer' to be more general than its common usage (namely, shorthand for 'digital computer') suggests. Specifically, we may view as a computer any system to which input values can be supplied and (possibly after processing by the system) from which output values can be drawn; then the *computation* is the relation—often a function—mapping input values to output values.

Whereas, in the digital case, input and output may take the form of the user's typing in values and reading values from a screen, our more general view of computation sees the user *manipulating parameters* of the system (so as to effect

input) and *measuring parameters* (to obtain output); e.g., a chemical computer may require input values and offer output values encoded in the concentrations of solutions—the user supplies a solution the concentration of which is suitably chosen to reflect (i.e., *encode*) his desired input value, and is (after the reactions, etc. of the system) presented with a solution the concentration of which reflects (encodes) the corresponding output value.

This extended view of computation—which accommodates not only standard, algorithmic (e.g., digital) computers, but also more natural, physical systems: quantum, DNA, analogue and optical computers amongst others—allows us to approach traditionally difficult problems in new ways.[1] Specifically, we consider here an *analogue-computer* solution to the problem of factorization.

1.2 Motivation

Though the factorization problem is easily posed—given $n \in \mathbb{N}$, what natural numbers exactly divide n?—, it seems inherently difficult to solve.

In [7], we see that the run-time of the best, known, *algorithmic* factorization methods grows exponentially with the size (i.e., number of digits) of the value factorized; this exponential time complexity renders infeasible the factorization of numbers beyond a certain size. So, whilst the equivalent decision problem[2] is not known to be **NP**-hard, neither do we expect an imminent, efficient algorithm.

The best, known, *quantum-computing* methods are technologically hard to implement; notably, Shor's algorithm ([10]), despite having run-time polynomial in the input value's size, has yet to factorize in practice a value greater than 15.

This apparent difficulty of factorizing via algorithmic or quantum means, along with the comments of Sect. 1.1, lead us to ask whether other computation models offer efficient solution. This question motivates the analogue system of [2] and the improved system below.

1.3 Original Analogue System

We recall the analogue factorizing system of [2]. Though a full description is deferred ([2] offers details of the system; [4] is the associated pending patent), we now outline some salient points.

Factorizing n is exactly the task of finding integer solutions x, y to $y = \frac{n}{x}$; i.e., finding points (x, y) in the integer grid $\mathbb{Z} \times \mathbb{Z}$ and on the curve $y = \frac{n}{x}$ (which curve is a hyperbola and, a fortiori, a conic section). Factorization, then, is the search for points in the intersection of a planar grid and a cone.

---

[1] As we shall see, however, part of the cost of this is the need for innovative forms of complexity analysis.

[2] The *decision* factorization problem asks, given $n \in \mathbb{N}$ and $l \leq n$, whether n has a factor a with $1 < a < l$. Ability efficiently to decide this implies ability efficiently to factorize: only logarithmically many decisions are needed to identify a factor.

Radiation can be reflected such that its interference pattern has a regular structure, of which the points of maximal wave activity may model $\mathbb{Z} \times \mathbb{Z}$.[3]

A second source of radiation, together with a part-circular sensor, can be used to model the cone: the source is the vertex, and the sensor's circle a cross section, of the cone. Diminishment of second-source radiation as it passes through an integer (i.e., maximally active) point in the first-source interference pattern is detected at the sensor; the coordinates of points of such detection can, Turing-computationally trivially, be converted into the coordinates of the sought integer points on the cone. These coordinates are factors of n. (We reiterate that full details of this system are available in [2].)

Of interest here is that the system's time and space complexities are *constant* in the size of n: the factorization of n requires neither more physical space nor more time as n increases. This is in contrast with known factorization algorithms, of which the required time is *exponential* in the size of n (see [7]).

This apparent efficiency is not, however, testament to the system's power, but rather exemplifies the inadequacy of traditional complexity theory for capturing the true complexity of non-Turing (e.g., analogue) computers. The system *does* require increasing—*exponentially* increasing—resource as n increases: for larger n, the user is required to position the apparatus and measure the position of points on the sensor with increasing *precision*. Intuitively, then, the system has non-constant *precision complexity*; this intuition is formalized in [3], in which precision complexity is motivated, defined and investigated.

1.4 Improving the Analogue System

We wish to improve this system in light of the impediment to its performance due to its precision complexity. A drawback of the system's design is the 'artificial' implementation of the cone[4]—essentially manual positioning of the vertex and a cross-sectional arc. Direct implementation of the cone by some physical phenomenon would, one expects, improve the precision of the apparatus. We seek, therefore, a naturally occurring, precisely shaped cone.

Consider a burst at time 0 of radiation from a point source in a plane P, and suppose that the radiation propagates at constant velocity v, regardless of direction or distance from source. The points reached by the radiation at time $t \geq 0$ describe a circle of radius vt (see Fig. 1(a)). Regarding this increasing circle in the space with two spatial axes (those of P) and one temporal axis, we have a perfectly constructed cone (see Fig. 1(b)).

The cone's original implementation, which we seek to replace with this radiation burst, exists in three spatial dimensions; relative to this, the proposed

[3] In fact, subset $\left\{(x, y) \in \mathbb{Z} \times \mathbb{Z} \mid 0 \leq x \leq y \leq n \wedge \frac{x+y}{2} \in \mathbb{Z}\right\}$ is modelled. This suffices as (a) factors of n need be sought only in $\{1, \ldots, n\}$; (b) having found factor pair $(p, n/p)$, finding $(n/p, p)$ is unnecessary; and (c) we assume that n is odd (see [2]).

[4] The drawback is crucial: a small imprecision in the shape/position of the cone leads only to a small imprecision in the resultant conic section, but this leads to the system's reporting wildly incorrect 'factors', in part because a small perturbation in n can cause a large perturbation in the *factors of* n.

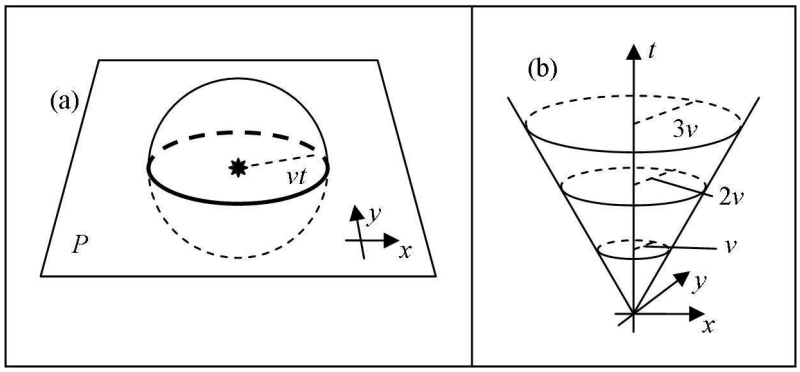

Fig. 1. (a) The circle of radiation in P at time t. (b) The circle plotted against t.

replacement is effectively rotated so as to span two spatial dimensions and one temporal. We now rotate the grid—in the intersection of which with the cone we are interested—similarly. The plane Q of the grid must lie parallel to and positively distant from the cone's axis, so that the resultant conic section is the required hyperbola; accordingly, in the radiation-burst implementation, Q is modelled as a permanent line in P, positively distant from the point source (Q thus spans one spatial and one temporal dimension). Within Q, the grid itself consists of points equally spaced along both axes: existing instantaneously at regular time intervals, and at regularly spaced points along the spatial axis.

The principle, then, is as with the original analogue system—we seek points of intersection of a cone and a planar grid—, though the implementation is different: before, the cone and grid existed in three-space; now, they exist in the plane, the cone as a steadily increasing circle, the grid as a flashing row of points. Further, since we seek intersection points with the grid, we need consider only those instants when the grid points are 'on', at which instants (together) the cone is modelled as a family of concentric circles with arithmetically progressional radii; *we seek the intersection of a row of points and a nest of concentric circles.*

The improved, analogue factorization system, of which the design is motivated by the informal comments of Sect. 1.4, is now formally described.

2 Analogue Factorization System

2.1 Apparatus

We describe the apparatus as lying in a plane; a physical realization would see the relevant features (X, Y, a and b) extended into the third dimension, though with readings being taken, etc. within the plane.

Definition 1 (provisional; see Definition 2).

– *Let n be the positive, natural number to be factorized.*[5]

[5] In [2], we assume for convenience that n is odd; we make no such assumption here.

– *Let X be an opaque screen occupying the line $(\mathbb{R}\setminus\{0, 2\sqrt{n}\})\times\{0\}$. The breaks at $(0, 0)$ and $(2\sqrt{n}, 0)$ are slits in the screen; call these a and b respectively.*
– *Let S be a source at $(\sqrt{n}, -\sqrt{n})$ of (say, e.m.) radiation of wavelength 1.*
– *Let Y be a screen occupying the line $\{0\}\times\mathbb{R}^+$.*[6] *The negligible width of slit a in X lies to the $x > 0$ side of Y.*

The intention is to observe radiation incident on Y from S via a and b. Where we find constructive interference at a point E on Y, we have that the respective distances from E to a and to b are integer multiples of the wavelength of S (that is, the distances are integers).[7] E is therefore both on one of the family of circles of integer radius and with centre b, and coincident with an integer point on the y-axis. This implementation, then, allows identification of the sought points described informally in Sect. 1.4 (we seek, recall, the intersection of a row of points and a nest of concentric circles).

Note that the layout of the apparatus (specifically the distance separating a and b, and the position of S) depends on n. In order to use the same apparatus to factorize different values, then, we scale each axis in the plane by a factor of $\frac{1}{2\sqrt{n}}$; accordingly, we replace the previous definition by the following.

Definition 2 (to replace Definition 1).

– *Let n be the positive, natural number to be factorized.*
– *Let X be an opaque screen occupying the line $(\mathbb{R}\setminus\{0, 1\})\times\{0\}$. The breaks at $(0, 0)$ and $(1, 0)$ are slits in the screen; call these a and b respectively.*
– *Let S be a source at $\left(\frac{1}{2}, -\frac{1}{2}\right)$ of (say, e.m.) radiation of wavelength $\frac{1}{2\sqrt{n}}$.*
– *Let Y be a screen occupying the line $\{0\}\times\mathbb{R}^+$. The negligible width of slit a in X lies to the $x > 0$ side of Y.*

See Fig. 2.

In so replacing Definition 1, we may reuse the apparatus to factorize any n, having to change only the wavelength of S rather than the system's layout.

2.2 Input to the System

As alluded to above, n is supplied to the system by altering to $\frac{1}{2\sqrt{n}}$ the wavelength of the radiation from S.

Note that the operations used in calculating this wavelength can be performed by a Turing machine in time polynomial in the size of n: there is no 'sleight of hand' whereby costly calculation is tacitly assumed to come for free.

[6] Note that $0 \in \mathbb{R}^+ := [0, \infty)$.

[7] In fact, we have that the distances' *difference* is a multiple of the wavelength. We henceforth assume, however, that we have instantiated along Y a standing wave or similar with maximally active points at wavelength spacing; we consider points at which constructive interference combines with this maximal standing-wave activity. This does not affect the complexity of the system, which we discuss later.

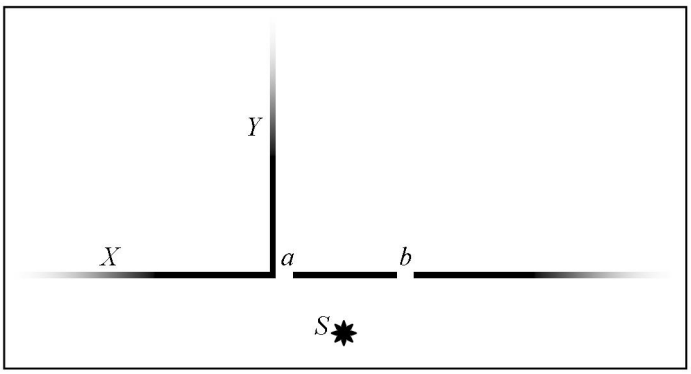

Fig. 2. The apparatus described in Definition 2.

2.3 Output from the System

Having set up the apparatus as in Definition 2 (including provision to the system of n, encoded in the wavelength of S), an interference pattern is produced on Y (since, on the $y > 0$ side of X, a and b act as separate, mutually coherent sources,[8] of which the respective waves interfere).

Since effective sources a and b are in phase, a point E on Y exhibits full, constructive interference (and so maximal brightness) if and only if the respective distances from E to a and to b are integer multiples of the wavelength $\frac{1}{2\sqrt{n}}$.

Proposition 1. *This maximal brightness is attained.*

Proof. The respective distances from $E := \left(0, \frac{n-1}{2\sqrt{n}}\right)$ to a and to b are $\frac{n-1}{2\sqrt{n}}$ and $\sqrt{\left(\frac{n-1}{2\sqrt{n}}\right)^2 + 1^2} = \frac{n+1}{2\sqrt{n}}$, each an integer multiple of $\frac{1}{2\sqrt{n}}$. Hence, E exhibits full, constructive interference. □

The process whereby output is read from the system consists of: identifying a maximally bright point E on Y, measuring the y-coordinate h of E, and calculating the values $p := \sqrt{n}\left(\sqrt{h^2+1} + h\right)$ and $\frac{n}{p}$.

As is proven below, the values p and $\frac{n}{p}$[9] so corresponding to any point of maximal brightness are factors of n. Further, each factor of n occurs as such p or $\frac{n}{p}$ for some maximally bright point. Thus, by processing all such points as described here, all factors of n are found.

As during input, there is no tacit presumption of a computationally complex operation: the process of finding p and $\frac{n}{p}$ given h is algorithmically efficient.

[8] This coherence is because S lies on the perpendicular bisector of ab.
[9] Since $h \geq 0$, we have that $\frac{n}{p} \leq \sqrt{n} \leq p$.

2.4 Proof of the System's Correct Functioning

Proposition 2. *A point on Y is maximally bright if and only if the corresponding value p is a factor of n no less than $\sqrt{n}$ (and so $\frac{n}{p}$ a factor at most $\sqrt{n}$).*

Proof. If $E := (0, h)$ is maximally bright, then the respective distances—h and $\sqrt{h^2+1}$—from E to a and to b are integer multiples of $\frac{1}{2\sqrt{n}}$; that is, $\alpha := 2\sqrt{n}h$ and $\beta := 2\sqrt{n}\sqrt{h^2+1}$ are integers (as, hence, are α^2 and β^2). Now

$$\beta^2 - \alpha^2 = 4n\left(h^2+1\right) - 4nh^2 = 4n \ , \tag{1}$$

which is even; so α^2 and β^2 have the same parity, as do α and β. Hence, $\beta \pm \alpha$ are even, and $\frac{\beta \pm \alpha}{2}$ are integers, with product $\frac{\beta+\alpha}{2} \cdot \frac{\beta-\alpha}{2} = \frac{\beta^2-\alpha^2}{4} \overset{(1)}{=} n$; that is, $\frac{\beta \pm \alpha}{2}$ are factors of n. Now $p := \sqrt{n}\left(\sqrt{h^2+1}+h\right)$ is exactly $\frac{\beta+\alpha}{2}$, a factor of n. Further, where $q = \sqrt{n}\left(\sqrt{h^2+1}-h\right)$, pq is $n\left(\sqrt{h^2+1}+h\right)\left(\sqrt{h^2+1}-h\right) = n\left(h^2+1-h^2\right) = n$, and $p \geq q$, so $p \geq \sqrt{n}$, as required.

Conversely, suppose that $E := (0, h)$ is such that $p := \sqrt{n}\left(\sqrt{h^2+1}+h\right)$ is a factor of n ($p \geq \sqrt{n}$ since $h \geq 0$). Let $q = \sqrt{n}\left(\sqrt{h^2+1}-h\right)$; $pq = n$, and so (since $p|n$) $q \in \mathbb{Z}$. p and q are integers, as are $p \pm q$, and $p+q = 2\sqrt{n}\sqrt{h^2+1}$ and $p-q = 2\sqrt{n}h$; i.e., $\sqrt{h^2+1}$ and h—the respective distances from E to b and to a—are integer multiples of $\frac{1}{2\sqrt{n}}$. Thus, E is maximally bright, as required. □

Having set up the system as in Definition 2, including having input n (encoded as a wavelength), the factors of n are found by measuring the y-coordinates of maximally bright points on Y and converting these into values p and $\frac{n}{p}$; Proposition 2 guarantees that the values so produced from all maximally bright points on Y are the factors of n and only the factors of n.

2.5 Practical Considerations

The description given of the system is an abstraction of any physical realization: aspects of the description require modification before the system can be practically implemented. We note above that the confinement of the apparatus to a plane is one such aspect; the screens X and Y, and slits a and b, should actually have positive, z-axis height, while S should remain as much as is practicable a point source, in the plane of which measurements on Y are taken.

Further, we cannot physically realize the infinitely long screen X. However, since we require of X and S only that, on the $y > 0$ side of X, a and b act as mutually coherent sources, we may replace X with a finite box

$$\begin{aligned}\left(\{(x,0) \mid -1 \leq x \leq 2\} \setminus \{(0,0),(1,0)\}\right) \cup \left\{(x, x-2) \mid \tfrac{1}{2} \leq x \leq 2\right\} \\ \cup \left\{(x, -x-1) \mid -1 \leq x \leq \tfrac{1}{2}\right\} \ ,\end{aligned}$$

sealed but for a and b, and containing S (which retains its position); we assume the interior to be non-reflective, absorbing radiation from S that does not directly

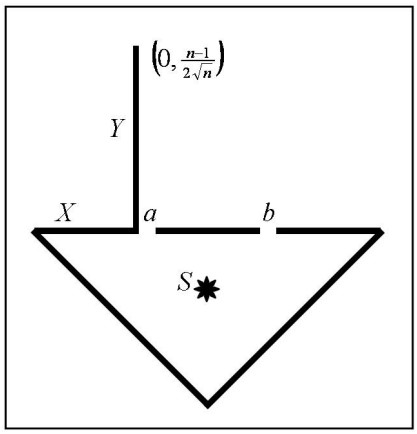

Fig. 3. The apparatus as modified in Sect. 2.5.

reach a or b. (Of course, the box has positive height in light of the preceding paragraph.) Fig. 3 shows the apparatus after this and the next modification.

Similarly, we cannot realize the infinitely long screen Y. It suffices, though, for Y to occupy the y-axis from 0 to $\frac{n-1}{2\sqrt{n}}$, since factors p of n (where, recall, $p \geq \frac{n}{p}$), which are in $\{\lceil\sqrt{n}\rceil, \ldots, n\}$, correspond (as in the output process of Sect. 2.3) to maximally bright points with y-coordinates in $\left[0, \frac{n-1}{2\sqrt{n}}\right]$.[10]

(A practical consideration not made in this theoretical context concerns the production of radiation of sufficiently short wavelength; as n increases, the required wavelength of S corresponds to an increasingly impractical energy.)

3 Complexity of the System; RSA Factorization

3.1 Time and Space Complexity

Consider first the system's time complexity. Use of the system consists of (a) provision to the system of the value n to be factorized, by way of adjustment of the wavelength of source S; (b) 'processing' by the system of n, by forming an interference pattern on Y; (c) measurement of the y-coordinates of maximally bright points on Y; and (d) conversion of these coordinates into factors of n. Of these, only (a) and (d) take longer as n increases (because of the Turing-machine-realm processing of n to find the corresponding wavelength and of y-coordinates to find factors), and even these take only polynomially long in the size $\log(n)$ of

[10] In making this observation, we remove the problem of Y being infinite, but reintroduce the undesirable property of the system's layout depending on n. This renders the system unsuitable for factorizing arbitrary, natural numbers, but is not a problem when factorizing numbers satisfying certain criteria; it is common, furthermore, for public keys used in the RSA cryptographic system to satisfy these criteria.

n; thus, the processing, physical-computing stages have *constant* time complexity, and their algorithmic 'harness', which prepares input and interprets output, has *polynomial* time complexity (as does the system as a whole, therefore).[11]

Consider now the system's space complexity. The apparatus has negligible, constant height (along the z-axis), a constant width (along the x-axis) of three (due to the box that replaces X), and a depth (along the y-axis) of $\frac{n+3\sqrt{n}-1}{2\sqrt{n}} \in \mathcal{O}\left(\sqrt{n}\right)$ (due to the box that replaces X and the shortened screen Y); the volume of the apparatus lies in $\mathcal{O}\left(\sqrt{n}\right)$, and is, hence, *exponential* in the size of n.

Let us put this in perspective. To factorize an arbitrary, 100-digit number (a relatively unambitious aim in the context of factorizing RSA keys, for example), we expect an apparatus depth of a 50-digit number of units and a width of three. The apparatus must be of the order of 10^{50} times as deep as it is wide; it is necessarily either too deep to be practicably accommodated, too narrow for slits with sufficiently small spacing feasibly to be manufactured, or both.

These considerations render the system unsuitable as a practical solution to the general problem of factorization. We now consider a subproblem, restriction to which greatly mitigates the problems discussed here.

3.2 RSA Factorization

From maximally bright point $(0, h)$ on Y, we find factors $p := \sqrt{n}\left(\sqrt{h^2+1}+h\right)$ and $\frac{n}{p}$ of n; but there is no a priori, n-independent upper bound for h: factors of n are found from values of h as large as $\frac{n-1}{2\sqrt{n}}$, which tends to ∞ as n does.

However, as h increases, the corresponding factors p and $\frac{n}{p}$ grow apart. When n is a square, there is a maximally bright point at $(0, 0)$, corresponding to the factorization $n = \sqrt{n}\sqrt{n}$; small h give close factor pairs. At the other extreme, for any positive, natural n, there is by Proposition 1 a maximally bright point at $\left(0, \frac{n-1}{2\sqrt{n}}\right)$, corresponding to the factorization $n = 1n$; large h give greatly differing pairs of factors. In fact, we have the following.

Proposition 3. *The factors* $p := \sqrt{n}\left(\sqrt{h^2+1}+h\right)$ *and* $q := \frac{n}{p}$ *corresponding to maximally bright point* $(0, h)$ *on* Y *differ by* $2\sqrt{n}h$.

Proof. $q = \sqrt{n}\left(\sqrt{h^2+1}-h\right)$, for then (as in the proof of Proposition 2) $pq = n$. So $p - q = \sqrt{n}\left(\sqrt{h^2+1}+h\right) - \sqrt{n}\left(\sqrt{h^2+1}-h\right) = 2\sqrt{n}h$, as required. □

Suppose now that we modify the system so that the size of Y is bounded; specifically, suppose that Y occupies the line segment $\{0\} \times [0, l]$ for some fixed (and, hence, n-independent), positive, real l. From Sect. 3.1, then, the system has constant space complexity.[12]

[11] Note that, under certain implementations, the sensor that identifies maximally bright points on Y is required to 'scan' Y in time linear in the length of Y. This length is in $\mathcal{O}\left(\sqrt{n}\right)$, rendering the system's time complexity exponential in the size $\log(n)$ of n. In Sect. 3.2, however, we modify the system so that this is no longer a concern.

[12] Its time complexity, further, is polynomial under all models of the sensor on Y.

For sufficiently large n (i.e., those with $\frac{n-1}{2\sqrt{n}} > l$), Y is no longer large enough to accommodate all maximally bright points corresponding to factors of n: those factor pairs corresponding to maximally bright points $(0, h)$ with $h > l$ are overlooked. However, we have the following.

Proposition 4. *If a pair (p, q) of factors of n (with $pq = n$ and $p \geq q$) satisfies $p \leq mq$, where $m = 2l + 1$, then these factors are not overlooked.*

Proof. Required is that the y-coordinate h of the maximally bright point corresponding to the factor pair (p, q) does not exceed l, for then this point falls on the modified (i.e., shortened) screen Y.

Since, by hypothesis, $p \leq mq$, $p - q \leq (m-1)\, q$; since, again by hypothesis, $pq = n$ and $p \geq q$, $q \leq \frac{n}{q}$, whence $q \leq \sqrt{n}$. Together, these give that

$$p - q \leq (m-1)\sqrt{n} \ . \tag{2}$$

Since, by definition, $m = 2l + 1$,

$$l = \frac{m-1}{2} \ . \tag{3}$$

Hence, $h \overset{\text{Prop. } 3}{=} \frac{p-q}{2\sqrt{n}} \overset{(2)}{\leq} \frac{(m-1)\sqrt{n}}{2\sqrt{n}} = \frac{m-1}{2} \overset{(3)}{=} l$; $h \leq l$, as required. □

Corollary 1. *If $l \geq \frac{1}{2}$, then all factor pairs (p, q) ($pq = n$, $p \geq q$) with $p \leq 2q$ are found by the modified system.*

Proof. If $l \geq \frac{1}{2}$, then $m := 2l + 1 \geq 2$; so $p \leq 2q$ implies that $p \leq mq$, whence Proposition 4 can be invoked. □

In modifying the system so that Y occupies $\{0\} \times \left[0, \frac{1}{2}\right]$, we lose the ability to factorize arbitrary, natural numbers; by Corollary 1, however, we can still factorize those values n of which each factor p no less than $\sqrt{n}$ (but strictly less than n[13]) satisfies $p \leq \frac{2n}{p}$. Further, we note that, for a public key $n = pq$ (with p and q prime) of the RSA system, the situation in which $q \leq p \leq 2q$ is common.[14]

Having noted an impracticality of the system when factorizing arbitrary, natural numbers (namely that, as n grows, the required ratio of the system's depth to its breadth grows exponentially), we have nonetheless found a subproblem—factorizing RSA keys—that this impracticality does not hinder.[15]

[13] We excuse the system for omitting to demonstrate that $n = 1n$.

[14] Were we even to weaken this condition to $q \leq p \leq 10q$, say—a conservative requirement of RSA keys—, then we can, by Proposition 4, still factorize n provided that Y spans $\{0\} \times \left[0, \frac{9}{2}\right]$. This proviso causes no difficulty in implementation.

[15] Further, in a sense, the subproblem captures the 'hardest' instances of (general) factorization: traditional, general-factorization algorithms typically take longest on input of the form pq, where p and q are primes of the same approximate size.

3.3 Precision Complexity

Having restricted factorization to RSA keys, we have a system with constant space and polynomial time complexity. The system, though, suffers from its *precision* complexity (see [3] for a formal account of precision): if the system rounds n to an integer, then the 'allowed', corrigible input error is $\pm\frac{1}{2}$; hence, the wavelength of S must be set in the interval $\left(\frac{1}{2\sqrt{n+\frac{1}{2}}}, \frac{1}{2\sqrt{n-\frac{1}{2}}}\right]$, which shrinks exponentially in the size of n. Containing the error to this extent requires precision (whilst setting the wavelength of S) *exponential* in the size of n.

Rather than an efficient system, we have yet further motivation[16] for extension of complexity theory beyond the essentially algorithmic. By introducing notions of complexity that, for certain physical computers, cater better than those of the traditional theory, [3] represents the beginnings of such extension.

(We suggest that the lack of such extension before [3] is because nearly all actual computation conforms to an algorithmic model (real-world computers are typically digital, running programs that implement algorithms), and also because of an overestimation of the ambit of Church's Thesis[17] (which ambit, we suggest, falls under computability rather than complexity); consequently, resource is typically taken to be a property—run-time, space, or another measure satisfying Blum's axioms[18]—of an algorithm, Turing machine, or equivalent.)

4 Conclusion

4.1 Summary

In Sect. 1, we note the apparent difficulty of algorithmic/quantum-computing factorization, and ask whether other computation models offer efficient solution. We recall an analogue system ([2]), but note its prohibitive complexity and seek to improve the system, motivated by a consideration of naturally occurring cones.

In Sect. 2, we define an improved analogue factorization system, detailing its apparatus and input/output processes. We discuss and resolve some, though not all, of the practical difficulties with the system's implementation.

In Sect. 3, we note the system's favourable time complexity, but that its space complexity impedes implementation. We resolve this by restricting the apparatus, though this restricts accordingly the problem solved by the system: while an implementable system for factorizing arbitrary, natural numbers seems out of reach, we present a system for factorizing a certain type of natural number—which type includes RSA keys—that is not subject to some of these concerns.

[16] We add this example to the soap bubble method ([9]) for finding minimum-length spanning networks connecting given vertices, the DNA computing technique ([1]) for tackling the directed Hamiltonian path problem, etc.

[17] Church's Thesis is introduced in [8] and discussed in [6], [11] and many others.

[18] Blum's axioms (see [5]) ensure that (a) a measure of resource is defined exactly at inputs at which the measured computation is defined, and (b) it is decidable whether a given value is indeed the measure of resource corresponding to a given input.

However, we note that the system's impressive time and space complexities are testament not to the system's efficiency, but to the inability of traditional, Turing-machine complexity theory to accommodate certain physical computers; we suggest that the notions of [3] better formalize the system's complexity.

Acknowledgements. We thank Bob Coecke and Joël Ouaknine (at Oxford) for their support and supervision, the IWNC reviewers for their detailed comments, and Rebecca Palmer (at Leeds) for noticing the omission rectified in footnote 7.

7.iii.2008

References

1. Adleman, L. M.: *Molecular Computation of Solutions to Combinatorial Problems.* Science **266** (1994) pp. 1021–1024
2. Blakey, E.: *An Analogue Solution to the Problem of Factorization.* Oxford University Computing Science Research Report CS-RR-07-04 (2007) `ftp://ftp.comlab.ox.ac.uk/pub/Documents/techreports/RR-07-04.pdf`
3. Blakey, E.: *On the Computational Complexity of Physical Computing Systems.* Unconventional Computing proceedings (2007) pp. 95–115 `http://users.ox.ac.uk/~quee1871/uc07_paper.pdf`
4. Blakey, E.: *System and Method for Finding Integer Solutions.* United States patent application 20070165313 (2007)
5. Blum, M.: *A Machine-Independent Theory of the Complexity of Recursive Functions.* J. of the Assoc. for Computing Machinery **14** no. 2 (1967) pp. 322–336
6. Bovet, D. P.; Crescenzi, P.: *Introduction to the Theory of Complexity.* Prentice Hall (1994)
7. Brent, R. P.: *Recent Progress and Prospects for Integer Factorisation Algorithms.* Lecture Notes in Computer Science **1858** (2000) pp. 3–20
8. Church, A.: *An Unsolvable Problem of Elementary Number Theory.* American J. of Math. **58** (1936) pp. 345–363
9. Miehle, W.: *Link-Length Minimization in Networks.* Operations Research **6** no. 2 (1958) pp. 232–243
10. Shor, P. W.: *Polynomial Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer.* SIAM J. Computing **26** (1997) pp. 1484–1509
11. Sipser, M.: *Introduction to the Theory of Computation.* PWS (1997)