Vector Factors and Vector Cryptography.

Factorising of Vectors

Factoring (factorising) of a vector as a mathematical method is the personal invention of this writer. It can be added to the existing vector methodology in mathematics as a new tool. It is not to be found in any of the usual course books or information sources it was copyrighted registered by me in the year 2000.

The vectors in question here are the three-dimensional physical vectors that are used to represent physical quantities that have both direction and magnitude such as velocity and acceleration but especially in the cryptography that follows – *displacement* is one of special interest.

Displacement is used as a temporary substitution in an obfuscation process that hides the legitimate plaintext items to all but the proper entities of a secure communications scheme. They alone are able to invert and recover the plaintext items from their transformed representation form as ciphertext.

The 'factors' of a vector are infinite in number and they occur as ordered pairs that always multiply out in the vector or 'cross' product to give the same result i.e. the vector being factorised – call this vector $\underline{N}$ say. Although they are taken in pairs, each one of the pair is a standalone single factor of $\underline{N}$ in its own right.

How it works.

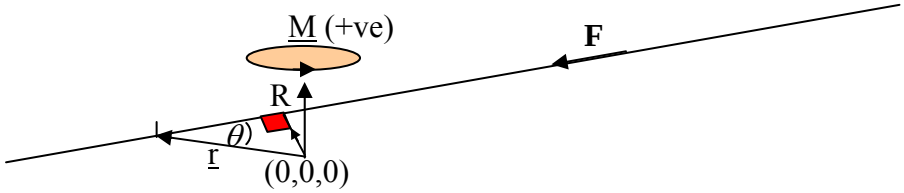
A vector to be factorised is proposed as being the defining normal vector of a plane that passes through the universal origin at (0,0,0) in the standard orthogonal frame of reference (X,Y,Z). Factor-lines may be defined within that plane such that any pair $\underline{V}n$ and $\underline{V}(n-1)$ on a factor line (taken in that order) are factors of $\underline{N}$. How factor lines are found methodically is described here – 'n' is any point on the line.

Then,

$$\underline{V}n \times \underline{V}(n-1) \equiv \underline{N} \text{ – true for all n.}$$

Vector Factorising.

The notion of factoring a vector came to me from seeing the **Force Line** at work.



$\underline{M}$ is the turning moment (turning effect) imposed on the vertical axis (passing through (0,0,0)) by the force $\underline{F}$ acting at any point '$\underline{r}$' on the line.

R is the effective crank (lever length) of 'r' and equals $|r|$ sine θ.

$|\underline{M}| \equiv |F||r|$ sine θ which is the same thing mathematically as the magnitude of the cross-product of $\underline{F} \times \underline{r}$. $\underline{M} = (\underline{F} \times \underline{r})$ has by convention the left-hand direction shown in the diagram.

$\underline{F}$, $\underline{r}$ and $\underline{M}$ form a right handed set
($\underline{r}$ is the position vector of any point on the line, $\underline{F}$ is a 'sliding' vector that has the same effect at any point of application defined by $\underline{r}$).

All of this is incidental to 'factoring' but the model suggests that the line of $\underline{F}$ could have ***any*** direction in the plane and $\underline{r}$ could be any point in the plane on the line. This model suggests that the pairs of $\underline{F}$ and $\underline{r}$ can be likened to ordered pairs of factors i.e. a factor and a cofactor (similar to the factors of a composite number) that when used as the operands of a cross-product always give the same result i.e. $\underline{M}$ in the diagram.

This suggests that if a vector is proposed as being the normal vector of a plane that passes through the origin then there are pairs of vectors within the plane (there for the finding) that when taken in a certain order are 'factors' per se of that vector. The experiment to find these factors is described here.

Digressing.

The current methodology of vector arithmetic is,
1) Vector addition.
2) Subtraction.
3) Multiplication in the Dot Product.
4) Multiplication in the Vector or Cross Product.
5) Multiplication by a scalar.

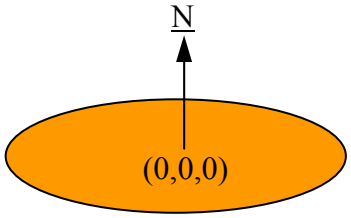
I am adding one more here,namely,
6) *Factorising (factoring) of a Vector.

Note: Vector division is not generally defined in vector methodology.

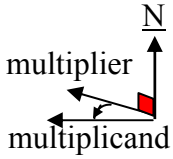
The vectors I am dealing with here are three-dimensional ones used to represent physical quantities such as displacement and velocity for instance.

Vector factoring.

A vector for factoring is proposed as being the defining 'normal' vector of a plane - albeit always shown here as a horizontal model the reader should understand that it can be expected to be inclined in any attitude in space in practice.



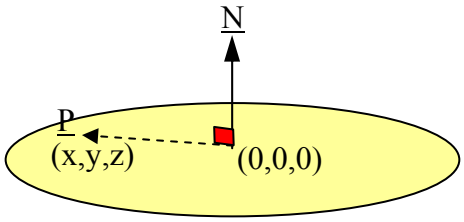
These are ordered 'factors' of $\underline{N}$ which is the vector or 'cross' product of the two.



vector product

The plane passes through the origin i.e. it contains the point (0,0,0).

Equation of the plane.



The point P (x,y,z) represents ***any*** point in the plane.
The direction ratios of the vector P relative to origin at (0,0,0) are (x-0), (y-0), (z-0).

Lemma_1.

The dot product of two mutually perpendicular vectors is always zero => $\underline{P} \bullet \underline{N} = 0$.

So, Letting $\underline{N} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$ and $\underline{P} = \begin{pmatrix} (x-0) \\ (y-0) \\ (z-0) \end{pmatrix}$

$$\underline{P} \bullet \underline{N} = \begin{pmatrix} (x-0) \\ (y-0) \\ (z-0) \end{pmatrix} \bullet \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} \rightarrow (\alpha \cdot x + \beta \cdot y + \gamma \cdot z) = 0$$

This is the standard Cartesian equation for a plane that passes through the origin, i.e. the equation of the plane is always.

$$(\alpha \cdot x + \beta \cdot y + \gamma \cdot z) = 0$$

i.e. this is the rule that must be satisfied by any point claiming to be '***in***' the plane.

Lemma_2.

Two non-parallel planes intersect along a straight line.

For clarity - harking back for a moment to two dimensions such as specifically the conventional (X,Y) plane the axes may be proposed as being intersected by a straight line. The intercept of each axis is then a single point. The adage for students is "Y intercepts occur when x = 0" and " X intercepts occur when y = 0". Importantly the intercept is a ***point.***

In three dimensions however the axes per se are instead formed by the bounding planes of the orthogonal frame of reference and any inclined plane, furthermore, the intercept is now a ***line*** that emanates from the intercept of each of the containing planes of the frame of reference as they meet with the inclined plane defined by N.

The Factors.

The factors are named $\underline{V}_0$ (VeeZero) and $\underline{V}_1$ (VeeOne) - these are a 'seeding' pair that propagate and lead to multiple factor lines.

Recapping on the discussion model to hand. I have three orthogonal planes that comprise the standard reference frame that is being intercepted by some inclined plane that is jointly defined by $\underline{N}$ (a defining normal vector) and the origin at (0.0.0).

Finding $\underline{V}_0$ (at the ZY intercept).

I need to find a point in the inclined plane,
the equation of the plane is,

$$(\alpha \cdot x + \beta \cdot y + \gamma \cdot z) = 0$$

I will guess that there is a point that has its x coefficient = 0 (certain to be the case? when $\underline{N}$ has three non-zero coefficients).

$$\text{So, } (\alpha \cdot 0 + \beta \cdot y + \gamma \cdot z) = 0$$
$$(\beta \cdot y + \gamma \cdot z) = 0$$

For this to be true either,

1) $y = -\gamma$ and $z = +\beta$

or

2) $y = +\gamma$ and $z = -\beta$

These are two options here that the reader may take. I have opted for 2) and this will always be used in these notes here in future. Note, option 1) is not to be discarded completely – it can be used additionally so as to provide extra options if that is ever required in the future.

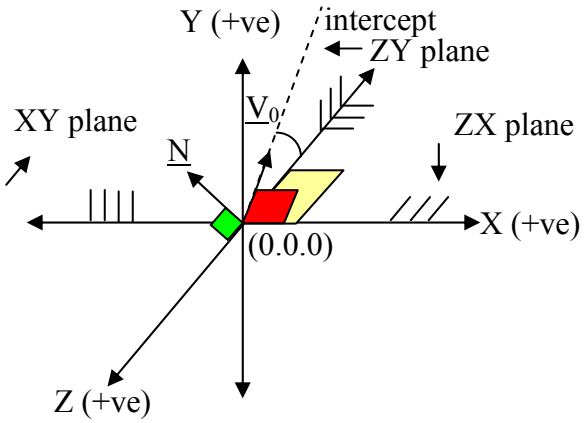
So, $\underline{V}_0$ can be written algebraically as $\begin{pmatrix} 0 \\ \gamma \\ -\beta \end{pmatrix}$ when x is put at 0.

I do not want $\underline{V}_0$ to have any scalar 'take-out' factor so I divide throughout by the GCD of $\{\beta, \gamma\}$ and I shall call this GCD $'\varepsilon_x'$ (Epsilon_x <= x being made zero).

$$\underline{V}_0 = \begin{pmatrix} 0 \\ \gamma / \varepsilon_x \\ -\beta / \varepsilon_x \end{pmatrix}$$

* This is the standard form that $\underline{V}_0$ will always take in the future.

Graphical aid.



To find the cofactor $\underline{V_1}$.

$\underline{V_1}$ needs to be found next. This is the position vector of another point nearby to $\underline{V_0}$, that is also in the same plane defined by $\underline{N}$ such that $\underline{V_1} \times \underline{V_0} = \underline{N}$.

$$\text{Putting } V_1 = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \text{ (for the time being),}$$

$$\underline{V_1} \text{ x } \underline{V_0} = \underline{N} \rightarrow \begin{pmatrix} x \\ y \\ z \end{pmatrix} \times \begin{pmatrix} 0 \\ \gamma/\varepsilon_x \\ -\beta/\varepsilon x \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$$

$$\begin{vmatrix} i & j & k \\ x & y & z \\ 0 & \gamma/\varepsilon_x & -\beta/\varepsilon_x \end{vmatrix} = \begin{pmatrix} - \\ - \\ \gamma \end{pmatrix}$$

Solving for the 'k' minor of the determinant gives me what I need to know here,

$$\text{Finding } \gamma \text{ on the RHS} \rightarrow (x \times \gamma/\varepsilon_x) - (y \times 0) = \gamma \rightarrow x = \varepsilon_x$$

($x = \varepsilon_x$ is fortuitous 'fallout' information from solving the 'k' minor of the determinant))

(I need to find y and z next)

Again, by the dot product,

$\underline{V}_1 \bullet \underline{N} \rightarrow \begin{pmatrix} x \\ y \\ z \end{pmatrix} \cdot \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \alpha \cdot x + \beta \cdot y + \gamma \cdot z = 0$ (the equation of the plane once more)

Having already found x,

1) $z = -(\alpha \cdot x + \beta \cdot y) / \gamma$ or 2) $y = -(\alpha \cdot x + \gamma \cdot z) / \beta$

Again, in this instance I also have two options as previously with finding $\underline{V}_0$

Taking option 1) and collecting terms,

$x = \varepsilon_x$

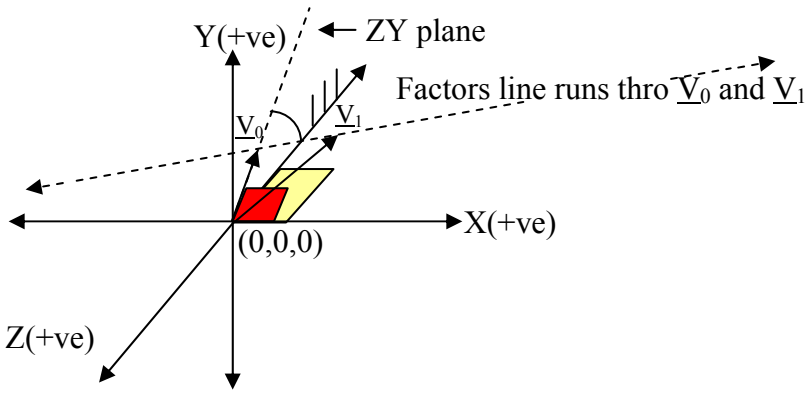
$y = y$

$z = -(\alpha \cdot x + \beta \cdot y) / \gamma$

$\underline{V}_1 = \begin{pmatrix} \varepsilon_x \\ y \\ -(\alpha \cdot \varepsilon_x + \beta \cdot y) / \gamma \end{pmatrix}$ (y decides z here - the first integer for y modulo γ (modulo gamma) will do fine but any other integer modulo γ is perfectly acceptable also.)

Altogether then, $\underline{V}_0 = \begin{pmatrix} 0 \\ \gamma / \varepsilon_x \\ -\beta / \varepsilon_x \end{pmatrix}$ $\underline{V}_1 = \begin{pmatrix} \varepsilon_x \\ y \\ -(\alpha \cdot \varepsilon_x + \beta \cdot y) / \gamma \end{pmatrix}$

This is the preferred form of these two 'primary' factors (*not prime factors) that evolve from considering the ZY intercept only. I shall always use these in the future unless otherwise stated for all workings.



Only one intercept has been taken into account so far i.e. the ZY intercept <= x =0
NB - Similar working is also possible by taking the other two intercepts into consideration (that working is not being repeated here but please see appendices – A, and B to view it),

1) i.e. letting $\underline{V}_0 = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ and <= putting y = 0 so that $\underline{V}_0 = \begin{pmatrix} x \\ 0 \\ z \end{pmatrix}$

2) i.e. letting $\underline{V}_0 = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ and <= putting z = 0 so that $\underline{V}_0 = \begin{pmatrix} x \\ y \\ 0 \end{pmatrix}$

This gives two more remaining results.

It is necessary now to prove this first result.

Proof of V_1 x V_0 (Evolving from the ZY intercept) = N.

The proof is simply to multiply out $\underline{V}_1$ and $\underline{V}_0$ algebraically in the vector or cross product method of vector multiplication.

The order of multiplication is always $\underline{V}_1 \times \underline{V}_0$

$$\begin{pmatrix} \varepsilon_x \\ y \\ -(\alpha.\varepsilon_x + \beta.y)/\gamma \end{pmatrix} \times \begin{pmatrix} 0 \\ \gamma/\varepsilon_x \\ -\beta/\varepsilon_x \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$$

using the conventional determinant method in the computing of a cross-product.

$$\begin{vmatrix} i & j & k \\ \varepsilon_x & y & -(\alpha.\varepsilon_x + \beta.y)/\gamma \\ 0 & \gamma/\varepsilon_x & -\beta/\varepsilon_x \end{vmatrix}$$

$$(+)\ \mathrm{i}\begin{vmatrix} y & -(\alpha.\varepsilon_x+\beta.y)/\gamma \\ \gamma/\varepsilon_x & -\beta/\varepsilon_x \end{vmatrix} = -\beta.y/\varepsilon_x + \alpha + \beta.y/\varepsilon_x = \alpha$$

$$(-)\ \mathrm{j}\begin{vmatrix} \varepsilon_x & -(\alpha\cdot\varepsilon_x+\beta\cdot y)/\gamma \\ 0 & -\beta./\varepsilon_x \end{vmatrix} = -\beta$$ (the -ve j minor of the determinant becomes + with change-of-sign convention $\rightarrow +\beta$.)

$$(+)\ \mathrm{k}\begin{vmatrix} \varepsilon_x & y \\ 0 & \gamma/\varepsilon_x \end{vmatrix} = \gamma$$

This demonstrates that $\begin{pmatrix} 0 \\ \gamma/\varepsilon_x \\ -\beta/\varepsilon_x \end{pmatrix}$ and $\begin{pmatrix} \varepsilon_x \\ y \\ -(\alpha.\varepsilon_x+\beta.y)/\gamma \end{pmatrix}$ are indeed proper primary algebraic factors of $\underline{N}$.

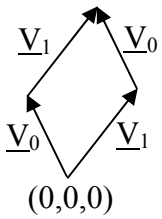
Lemma_4.

Equations of a straight line.

A particular line is uniquely located in space if,

a) it has a known direction and it passes through a known point,

b) it passes through two known points.



Equations of lines that can be defined immediately from any pair V_0 and $\underline{V}_1$ as found initially,

Factor Line 1 :- $\underline{V}n = \underline{V}_0 + n\,(\underline{V}_1 - \underline{V}_0)$

Factor Line 2 :- $\underline{V}n = \underline{V}_0 + n\,(\underline{V}_1 + \underline{V}_0)$

Factor Line 3 :- $\underline{V}n = V_0 + n\,(V_1)$

Factor Line 4 :- $\underline{V}n = \underline{V}_1 + n\,(-\underline{V}_0)$

Factor Lines. These factors of '$\underline{N}$' are being called 'primary' because they are indeed a 'seeding' pair that is extensible to the full infinite family of factors that exists for any '$\underline{N}$'.

Family indeed is the word for it. The pairs of vectors $\underline{V}_0$ and $\underline{V}_1$ wherever they are found initially are each a single salient result on their own but their main use is to define general factor lines that are indeed further usable in cryptogaraphy as directed number lines. In vector cryptography the numbers on the lines that traditionally went on the universal, arbitrary number line can instead be defined by the corresponding vector factors of $\underline{N}$ as position vectors of displacements in space. Although discussed as pairs here in the realisation of factors of $\underline{N}$ each one of the pair is a valid *standalone* factor of $\underline{N}$ in its own right . In general, only integer values of numbers are used here in these notes although the methodology is valid for both float and integer values alike.

It becomes a matter of using various ploys (mainly while programming) that suggest themselves easily to the user to explore the plane looking for fresh directions and points that can be used to create more and more factor lines. I repeat, although these factors of $\underline{N}$ are being talked about here in pairs each one of the pair is a standalone factor of $\underline{N}$ in its own right and can be quoted as such.

Collected Equations of basic Factor Lines

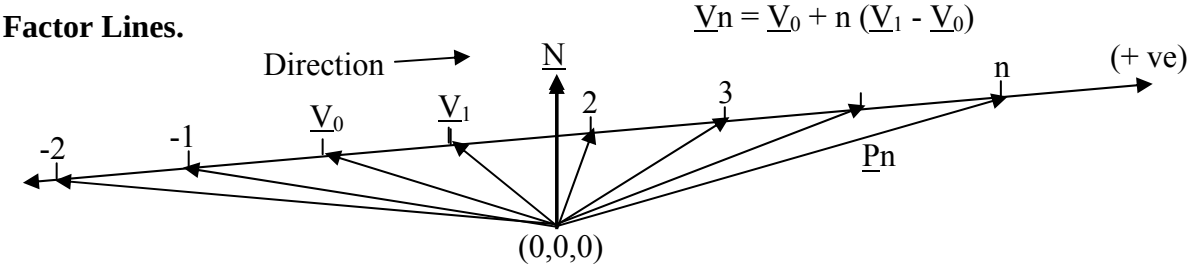
Factor Line 1 :- $\underline{V}n = \underline{V}_0 + n\,(\underline{V}_1 - \underline{V}_0)$
Factor Line 2 :- $\underline{V}n = \underline{V}_0 + n\,(\underline{V}_1 + \underline{V}_0)$
Factor Line 3 :- $\underline{V}n = V_0 + n\,(V_1)$
Factor Line 4 :- $\underline{V}n = \underline{V}_1 + n\,(-\underline{V}_0)$

Recursion.

Any pair of factors $\underline{V}(n\text{-}1)$ and $V(n)$ taken from any line may be recycled as $\underline{V}_0$ and $\underline{V}_1$ respectively in another line to form the basis of totally new explicit equations of fresh factor lines *provided that they are not repeating their own line equation*. For example, a pair taken thus from Factor Line 1) may be used in the explicit equations of the remaining three lines but not in their own line 1). The latter would simply be tantamount to giving a change-of-origin to a previously existing line.

A typical Factor Line

Factor Lines.



Equation of this line, $\underline{V}n = V_0 + n \cdot$ (direction vector)

Note. The direction vector can be the sum or the difference of the primary pair $\underline{V}_0$ and $\underline{V}_1$ in each equation – in this instance shown here the (direction vector) = ($\underline{V}_1$ – $\underline{V}_0$)). It could be in accordance with any of the four equations show above.

Claim is :- $\underline{V}_n \times \underline{V}_{(n-1)} = \underline{N}$ is true for all 'n'

This has to be proved.

Lemma – 1

The area of a triangle is half the base times the perpendicular height.

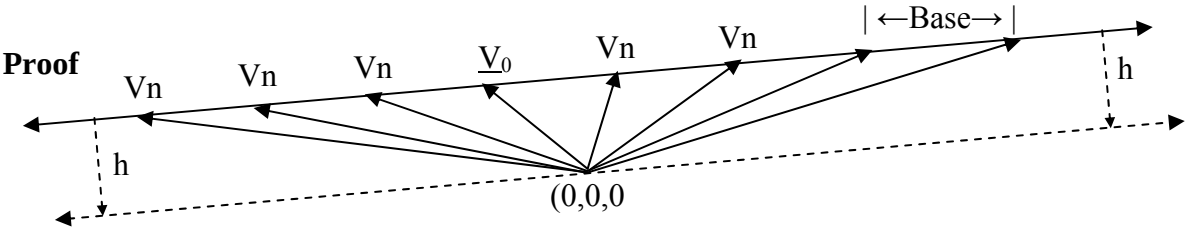
Lemma – 2

The area of a triangle is half the magnitude of the vector cross product of any two sides = half the magnitude of $\underline{N}$ in the diagram.

Lemma – 3

Vectors that have the same magnitude and direction are equal.

Proof



The construction lines together with the theory already expounded show that each triangle has the same base and the same perpendicular height => by lemma -1 these triangles are equal in area.

Lemma - 2 enables the same thing to be said in vector parlance,

$|\underline{V}n \times Vn\text{-}1|$ is true for all n

$|\underline{V}n \times Vn\text{-}1| = |\underline{N}|$ from $\underline{V}_1 \times \underline{V}_0 = \underline{N}$

⇨ The magnitudes of all the vector products $\underline{V}n \times Vn\text{-}1$ are equal for all 'n'.

⇨ Because all of the operands (multiplier and multiplicand) of the vector products | $\underline{V}n \times V_{n-1}$| are in the same plane their respective directions are all coincident with the defining normal of the plane $\underline{N}$ and they are also equal.

By lemma – 3 , having the same magnitude and the same direction all of the vector products $\underline{V}n \times V_{n-1}$ are equal to $\underline{N}$

$\therefore$ **$\underline{V}n \times V_{n-1} = \underline{N}$ is true for all n**

The special property of the cross product that underpins the claims being made here is that the cross product of two vectors is always a third vector that has a direction that is perpendicular to the plane of the two factors i.e the plane that contains the two operands, the multiplier and multiplicand, as factors of $\underline{N}$. This is a very, very useful property of the vector cross-product whenever it is needed in mathematics.

Comments.

1) $\underline{V}_0$ is the natural mid-point of a factor line i.e. initially it always occurs right on the line of interception.

2) In general the mid-point of a factor line is not unique and may be varied at will.

1) All of these triangles are (totally dissimilar) 'scalene' triangles i.e. no two angles of any triangle are ever equal and no two sides are ever equal as a consequence.

2) The separate position vectors '$\underline{V}n$' have coefficients that are always a co-prime set of integers.

3) The position vectors themselves are line analogues of 'n'.

4) The triangles bounded by the position vectors $\underline{V}n$ and $\underline{V}(n-1)$ are area analogues of 'n' in each case.

5) An inclined plane defined by $\underline{N}$ can be tiled by any one of these triangles.

6) the zero vector $\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$ has "any pair of parallel vectors" as its factors.

These facts can be used to advantage by designers in many disciplines.
The foregoing exercise focused on the ZY intercept and initiated a factor finding scheme that emanated from that intercept. Identical working will provide similar

results working from the ZX and XY intercepts and this is shown briefly in Appendix_B and Appendix_C. All three intercepts are needed to cover the entire possibilities space of vector factors for a given vector.

If a vector for factoring has three non-zero coefficients then there will be three distinct intercepts and three sets of working will be required to find the entire set of factors for that vector.

Appendix_A - Considering the ZX intercept.

The equation of the plane is again,

$$\alpha \cdot x + \beta \cdot y + \gamma \cdot z = 0$$

I am looking for a point in the plane that will satisfy the equation of the plane.

I am going to guess a point that has y = 0 and then express the remaining two, each in terms of the other.

$$\alpha \cdot x + \gamma \cdot z = 0$$

For this to be true either,

1) x = $+\gamma$ and z = $-\alpha$

or

2) x = - γ and z = $+\alpha$

These are two options here that the reader may take. I have opted for 2) and this will always be used in these notes here in future. Note, option 1) is not to be discarded completely – it can be used additionally so as to provide an extra directed number-line if that is ever required in some special case in the future.

So, $\underline{V}_0$ is written algebraically as $\begin{pmatrix} -\gamma \\ 0 \\ \alpha \end{pmatrix}$ when x is put at 0.

I do not want $\underline{V}_0$ to have any scalar take-out factor so I divide throughout by the GCD of $\{\alpha, \gamma\}$ and I shall call this GCD 'ε_y' (Epsilon_y <= y = 0).

$$\underline{V}_0 = \begin{pmatrix} -\gamma/\varepsilon_y \\ 0 \\ \alpha/\varepsilon_y \end{pmatrix}$$

* This is the standard form that $\underline{V}_0$ will always take in the future.

To find the cofactor $\underline{V}_1$.

$\underline{V}_1$ needs to be found next. This is the position vector of another point nearby to this $\underline{V}_0$, that is also in the plane defined by $\underline{N}$ such that $\underline{V}_1 \times \underline{V}_0 = \underline{N}$.

$$\text{Putting } V_1 = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \text{ for time being}$$

$$\underline{V}_1 \text{ x } \underline{V}_0 = \underline{N} \rightarrow \begin{pmatrix} x \\ y \\ z \end{pmatrix} \times \begin{pmatrix} -\gamma/\varepsilon_y \\ 0 \\ \alpha/\varepsilon_y \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$$

$$\begin{vmatrix} i & j & k \\ x & y & z \\ -\gamma/\varepsilon_y & 0 & \alpha/\varepsilon_y \end{vmatrix} = \begin{pmatrix} \alpha \\ - \\ - \end{pmatrix}$$

Solving for the i minor of the determinant gives me all I need to know here,

$$\text{Finding } \alpha \text{ on the RHS} \rightarrow (y \times \alpha/\varepsilon_y) - (z \times 0) = \alpha$$

$$y = \varepsilon_y$$

(I need to find x and z next)

Again, by the dot product,

$$\underline{V}_1 \bullet \underline{N} \rightarrow \begin{pmatrix} x \\ y \\ z \end{pmatrix} \cdot \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \alpha \cdot x + \beta \cdot y + \gamma \cdot z = 0 \text{ (the equation of the plane once more)}$$

$$1)\ x = -(\beta \cdot y + \gamma \cdot z)/\alpha \text{ or } 2)\ z = -(\alpha \cdot x + \beta \cdot y)/\gamma$$

Again, in this instance I also have two options as previously with finding $\underline{V}_0$

Taking option 1) and collecting terms,

$$x = -(\beta \cdot \varepsilon_y + \gamma \cdot z)/\alpha$$

$$y = \varepsilon_y$$

$$z = z$$

$$V_1 = \begin{pmatrix} -(\beta \cdot \varepsilon_y + \gamma \cdot z)/\alpha \\ \varepsilon_y \\ z \end{pmatrix} \text{ (z decides x)}$$

Altogether then, at the ZX intercept, $\underline{V}_0 = \begin{pmatrix} -\gamma/\varepsilon_y \\ 0 \\ \alpha/\varepsilon_y \end{pmatrix}$ $\underline{V}_1 = \begin{pmatrix} -(\beta \cdot \varepsilon_y + \gamma \cdot z)/\alpha \\ \varepsilon_y \\ z \end{pmatrix}$

This is the preferred form of these two 'primary' factors (not prime factors) that evolve from considering the ZX intercept only. I shall always use these in the future unless otherwise stated for all workings.

Proof that this pair V_1 x V_0 = N.

The proof is simply to multiply out $\underline{V}_1$ and $\underline{V}_0$ algebraically in the vector or cross product method of vector multiplication.

The order of multiplication is always $\underline{V}_1 \times \underline{V}_0$

$$\begin{pmatrix} -(\beta \cdot \varepsilon_y + \gamma \cdot z)/\alpha \\ \varepsilon_y \\ z \end{pmatrix} \times \begin{pmatrix} -\gamma/\varepsilon_y \\ 0 \\ \alpha/\varepsilon_y \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$$

using the conventional determinant method in the computing of a cross-product.

$$\begin{vmatrix} i & j & k \\ -(\beta \cdot \varepsilon_y + \gamma \cdot z)/\alpha & \varepsilon_y & z \\ -\gamma/\varepsilon_y & 0 & \alpha/\varepsilon_y \end{vmatrix} 1$$

$$(+)\ i \begin{vmatrix} \varepsilon_y & z \\ 0 & \alpha/\varepsilon_y \end{vmatrix} = \alpha$$

$$(-)\ j \begin{vmatrix} -(\beta \cdot \varepsilon_y + \gamma \cdot z)/\alpha & z \\ -\gamma/\varepsilon_y & \alpha/\varepsilon_y \end{vmatrix} = -\beta \text{ (j minor of the determinant}$$

$$-(\beta\cdot\varepsilon_y+\gamma.z)\Big/\alpha\times\alpha\Big/\varepsilon_y-(-\gamma\cdot z\Big/\varepsilon_y)=-\beta-(\gamma\cdot z)\Big/\varepsilon_y+(\gamma\cdot z)\Big/\varepsilon_y=-\beta$$

$-\beta$ becomes (+) β in accordance with the usual change-of-sign).

$$(+)\,k\begin{vmatrix}-(\beta\cdot\varepsilon_y+\gamma\cdot z)\Big/\alpha & \varepsilon_y\\ -\gamma\Big/\varepsilon_y & 0\end{vmatrix}=\gamma$$

This demonstrates that $\begin{pmatrix}-\gamma\Big/\varepsilon_y\\ 0\\ \alpha\Big/\varepsilon_y\end{pmatrix}$ and $\begin{pmatrix}-(\beta\cdot\varepsilon_y+\gamma\cdot z)\Big/\alpha\\ \varepsilon_y\\ z\end{pmatrix}$ are indeed de facto primary algebraic factors of N.

Appendix_B - Considering the XY intercept.

Finding V_0 (at the XY intercept).

Once more I need to find a point in the inclined plane -
the equation of the plane to hand is,

$$(\alpha\cdot x+\beta\cdot y+\gamma\cdot z)=0$$

I will guess that there is a point in the plane that has the z coefficient = 0 (certain to be the case? when N has three non-zero coefficients).

$$\text{So, } (\alpha\cdot x+\beta\cdot y+\gamma\cdot 0)\ =0$$
$$(\alpha\cdot x+\beta\cdot y)=0$$

For this to be true either,

1) y = $+\alpha$ and x = $-\beta$

or

2) y = $-\alpha$ and x = $+\beta$

These are two options here that the reader may take. I have opted for 2) and this will always be used in these notes here in future. Note, option 1) is not to be discarded completely – it can be used additionally so as to provide an extra directed number-line if that is ever required in some special case in the future.

So, $\underline{V}_0$ is written algebraically as $\begin{pmatrix} B \\ -\alpha \\ 0 \end{pmatrix}$ when z is put at 0.

I do not want $\underline{V}_0$ to have any scalar take-out factor so I divide throughout by the GCD of $\{\alpha, \beta\}$ and I shall call this GCD $'\varepsilon_z'$ (Epsilon_z <= z = 0).

$$\underline{V}_0 = \begin{pmatrix} \beta/\varepsilon_z \\ -\alpha/\varepsilon_z \\ 0 \end{pmatrix}$$

* This is the standard form that $\underline{V}_0$ will always take in the future.

To find the cofactor $\underline{V}_1$.

$\underline{V}_1$ needs to be found next. This is the position vector of another point nearby to $\underline{V}_0$, also in the plane defined by $\underline{N}$ such that $\underline{V}_1 \times \underline{V}_0 = \underline{N}$.

Putting $V_1 = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ for time being

$$\underline{V}_1 \text{ x } \underline{V}_0 = \underline{N} \rightarrow \begin{pmatrix} x \\ y \\ z \end{pmatrix} \times \begin{pmatrix} \beta/\varepsilon_z \\ -\alpha/\varepsilon_z \\ 0 \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$$

$$\begin{vmatrix} i & j & k \\ x & y & z \\ \beta/\varepsilon_z & -\alpha/\varepsilon_z & 0 \end{vmatrix} = \begin{pmatrix} \alpha \\ - \\ - \end{pmatrix}$$

Solving for the i minor of the determinant gives me all I need to know here,

$$\text{Finding } \alpha \text{ on the RHS} \rightarrow (y \cdot 0) - (^{-\alpha}/_{\varepsilon_z} \cdot z) = \alpha$$

$$z = \varepsilon_z$$

(I need to find y and z next)

Again, by the dot product,

$\underline{V}_1 \bullet \underline{N} \rightarrow \begin{pmatrix} x \\ y \\ z \end{pmatrix} \cdot \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \alpha \cdot x + \beta \cdot y + \gamma \cdot z = 0$ (the equation of the plane once more)

$$1)\ y = -(\gamma \cdot z + \alpha \cdot x) \Big/ \beta \text{ or } 2)\ x = -(\gamma \cdot z + \beta \cdot y) \Big/ \alpha$$

Again, in this instance I also have two options as previously with finding $\underline{V}_0$

Taking option 1) and collecting terms,

$$z = \varepsilon_z$$

$$x = x$$

$$y = -(\gamma \cdot \varepsilon_z + \alpha \cdot x) \Big/ \beta$$

$$\underline{V}_1 = \begin{pmatrix} x \\ -(\gamma \cdot \varepsilon_z + \alpha \cdot x) \Big/ \beta \\ \varepsilon_z \end{pmatrix} \text{ (x decides y)}$$

Altogether then, $\underline{V}_0 = \begin{pmatrix} \beta \Big/ \varepsilon_z \\ -\alpha \Big/ \varepsilon_z \\ 0 \end{pmatrix}$ $\underline{V}_1 = \begin{pmatrix} x \\ -(\gamma \cdot \varepsilon_z + \alpha \cdot x) \Big/ \beta \\ \varepsilon_z \end{pmatrix}$

This is the preferred form of these two 'primary' factors (not prime factors) that evolve from considering the XY intercept only. I shall always use these in the future unless otherwise stated for all workings.

Proof that this pair V_1 x V_0 = N.

The proof is simply to multiply out $\underline{V}_1$ and $\underline{V}_0$ algebraically in the vector or cross product method of vector multiplication.

The order of multiplication is always $\underline{V}_1 \times \underline{V}_0$

$$\begin{pmatrix} x \\ -(\gamma \cdot \varepsilon_z + \alpha \cdot x) \Big/ \beta \\ \varepsilon_z \end{pmatrix} \times \begin{pmatrix} \beta \Big/ \varepsilon_z \\ -\alpha \Big/ \varepsilon_z \\ 0 \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$$

using the conventional determinant method in the computing of a cross-product.

$$\begin{vmatrix} i & j & k \\ x & -(\gamma\cdot\varepsilon_z+\alpha\cdot x)/\beta & \varepsilon_z \\ \beta/\varepsilon_z & -\alpha/\varepsilon_z & 0 \end{vmatrix}$$

$$(+)\ \mathrm{i}\begin{vmatrix} -(\gamma\cdot\varepsilon_z+\alpha\cdot x)/\beta & \varepsilon_z \\ -\alpha/\varepsilon_z & 0 \end{vmatrix} = \alpha$$

$$(-)\ \mathrm{j}\begin{vmatrix} x & \varepsilon_z \\ \beta/\varepsilon_z & 0 \end{vmatrix} = -\beta$$ (the j minor of the determinant becomes $+\beta$ with change-of-sign)

$$(+)\mathrm{k}\begin{vmatrix} x & -(\gamma\cdot\varepsilon_z+\alpha\cdot x)/\beta \\ \beta/\varepsilon_z & -\alpha/\varepsilon_z \end{vmatrix} = \left(-\alpha\cdot x/\varepsilon_z\right) - \left(-(\gamma\cdot\varepsilon_z+\alpha\cdot x)/\beta \times \beta/\varepsilon_z\right) = \gamma$$

This demonstrates that $\begin{pmatrix} \beta/\varepsilon_z \\ -\alpha/\varepsilon_z \\ 0 \end{pmatrix}$ and $\begin{pmatrix} x \\ -(\gamma\cdot\varepsilon_z+\alpha\cdot x)/\beta \\ \varepsilon_z \end{pmatrix}$ are indeed de facto primary algebraic factors of $\underline{N}$.

Appendix - C. A Worked Example of Factoring.

Let us say that on this occasion the vector to be factored is, $\underline{N} = 8\hat{i} + 6\hat{j} - 15\hat{k}$. This '$\underline{N}$' is used to define a plane that passes through the origin (0, 0, 0) of the standard frame of reference frame (X, Y, Z).

Algebraically $\underline{N} = \alpha\,\hat{i} + \beta\,\hat{j} + \gamma\,\hat{k}$.

$$\text{Then, } \underline{V}_0 = \begin{pmatrix} 0 \\ \gamma/\varepsilon_x \\ -\beta/\varepsilon_x \end{pmatrix} \text{ and } \underline{V}_1 = \begin{pmatrix} \varepsilon_x \\ y \\ -(\alpha\,\varepsilon_x + \beta\,y)/\gamma \end{pmatrix}$$

ε_x is the GCD of β and γ

$\underline{V}_0$ and $\underline{V}n$ are the algebraic primary factors of $\underline{N}$.

The factor line that will be used (one of the four available lines) has the vector equation,

$$\underline{V}n = \underline{V}_0 + n\,(\underline{V}_1 - \underline{V}_0)$$

The normal vector being chosen for this demonstration is,

$$\underline{N} = \begin{pmatrix} 8 \\ 6 \\ -15 \end{pmatrix}$$

i.e. $\alpha = 8, \beta = 6, \gamma = -15$ and $\varepsilon_x = 3$

The user factorises this $\underline{N}$ to find the primary pair $\underline{V}_0$ and $\underline{V}_1$ at the ZY axis.

$$\underline{V}_0 = \begin{pmatrix} 0 \\ \gamma/\varepsilon x \\ -\beta/\varepsilon_x \end{pmatrix} = \begin{pmatrix} 0 \\ -15/3 \\ -6/3 \end{pmatrix} = \begin{pmatrix} 0 \\ -5 \\ -2 \end{pmatrix}$$

And

$$\underline{V}_1 = \begin{pmatrix} \varepsilon_x \\ y \\ -(\alpha.\varepsilon_x + \beta.y)/\gamma \end{pmatrix} = \begin{pmatrix} 3 \\ y \\ -(8\times 3 + 6\times y)/-15 \end{pmatrix} *$$

* y decides z here - any integer value of y satisfies z

when y = 1 $z = {}^{-30}\!/_{-15} = 2$

So,

$$\underline{V}_1 = \begin{pmatrix} 3 \\ 1 \\ 2 \end{pmatrix}$$

- In the expression above, $z = -(8 \times 3 + 6 \times y) / (-15)$, y decides z and they are both integers. In this highly contrived demonstration example it is very easy using just mental arithmetic to see what integer value of y satisfies the equation but when the operands are large this becomes a very, very difficult task. The computer program to hand uses a vector factoring program that has a specially designed utility procedure that is dedicated to finding 'y' in all cases no matter how difficult so there is no problem to the human user.

Note: It is good practice to always check that,

$\underline{V}_1$ x $\underline{V}_0 = \underline{N}$, after finding those two.

$$\underline{V}_0 = \begin{pmatrix} 0 \\ -5 \\ -2 \end{pmatrix} \text{ and } \underline{V}_1 = \begin{pmatrix} 3 \\ 1 \\ 2 \end{pmatrix}$$

$$\begin{pmatrix} 3 \\ 1 \\ 2 \end{pmatrix} \text{ x } \begin{pmatrix} 0 \\ -5 \\ -2 \end{pmatrix} = \begin{pmatrix} 8 \\ 6 \\ -15 \end{pmatrix}$$

They are true factors!

Reminder, the chosen line is,

$$\underline{V}n = \underline{V}_0 + n\,(\underline{V}_1 - \underline{V}_0)$$

$$(\underline{V}_1 - \underline{V}_0) \text{ is } \begin{pmatrix} 3 \\ 1 \\ 2 \end{pmatrix} - \begin{pmatrix} 0 \\ -5 \\ -2 \end{pmatrix} = \begin{pmatrix} 3 \\ 6 \\ 4 \end{pmatrix}$$

Explicitly then,

The chosen line is,

$$\underline{V}n = \begin{pmatrix} 0 \\ -5 \\ -2 \end{pmatrix} + n \begin{pmatrix} 3 \\ 6 \\ 4 \end{pmatrix}$$

The position vector $\underline{V}n$ of any number 'n' can be found by substituting in the appropriate value of 'n' in the RHS.

Let us say that $n = 489$. $\Rightarrow n\text{-}1 = 488$

Substituting 'n' into the equation of the number line,

$$\underline{V}n = \begin{pmatrix} 0 \\ -5 \\ -2 \end{pmatrix} + n \begin{pmatrix} 3 \\ 6 \\ 4 \end{pmatrix}$$

$$\Rightarrow \underline{P}\,489 = \begin{pmatrix} 0 \\ -5 \\ -2 \end{pmatrix} + 489 \begin{pmatrix} 3 \\ 6 \\ 4 \end{pmatrix}$$

(The position vector is given the variable name P now)

$$\Rightarrow \underline{P}\,489 = \begin{pmatrix} 1467 \\ 2929 \\ 1954 \end{pmatrix}$$

$$\Rightarrow P(n\text{-}1) = \underline{P}\,488 = \begin{pmatrix} 0 \\ -5 \\ -2 \end{pmatrix} + 488 \begin{pmatrix} 3 \\ 6 \\ 4 \end{pmatrix}$$

$$\Rightarrow P488 = \begin{pmatrix} 1464 \\ 2923 \\ 1950 \end{pmatrix}$$

Checking that $\underline{P}n \times \underline{P}(n\text{-}1) = \underline{N}$,

$$\begin{pmatrix} 1467 \\ 2929 \\ 1954 \end{pmatrix} \times \begin{pmatrix} 1464 \\ 2923 \\ 1950 \end{pmatrix} = \begin{pmatrix} 8 \\ 6 \\ -15 \end{pmatrix}$$ - correct.

* $\underline{P}n$ and $P(n\text{-}1)$ are an ordered pair of bona fide factors of $\underline{N}$

(This example is related to vector factoring).

When a seeding pair is found then the explicit equations of multiple factor lines may be found according to the equations described.

Vector Cryptography.

Introduction.

In this cryptography the entities use a ***directed*** number-line to model the code points of plaintexts being encrypted instead of the arbitrary straight line of the traditional number system. A different line i.e. having a different direction is used for each and every encryption of a single plaintext before being discarded for the next plaintext. This makes available for full use by cryptographers the entire infinite set of directions in three-dimensional space as the encryption domain.

The upshot of doing this is that they can customise a unique number system for the encryption of each and every plaintext of a secure message to which they alone are privy and it becomes impossible for any adversary to invert their ciphertext and break the secrecy of message texts.

The direction of the line being used is unique and the periodicity (the space between integers) of the line is fixed also by being a function (the magnitude) of the direction vector of the line in hand. Every number-line is defined by its 'vector equation' which may be public knowledge to adversaries. A different line having a fresh direction is used for each and every new plaintext being encrypted but the general algebraic equation of the line remains the same for the duration of the session in hand albeit changeable as an option in future sessions).

Numbers, i.e. integer code-points of plaintext representations on the encryption number line, are defined by the position vector Pn of each number 'n' on the line. Pn is the displacement of the code point 'n' relative to the origin at (0,0,0) as defined by the position vector Pn.

The cryptography can be variously called, "Vector Cryptography", "Displacement Cryptography", "Spatial Cryptography" and "Skew Line Cryptography" since all of these names are appropriate.

The cipher benefits from my personal invention of "Vector Factoring" which is a new mathematical tool in vector methodology (already described).

The primary transformation of each plaintext which includes 'digital signature' is not described here and this is not important to the description of the cipher algorithm. It can be assumed that it has been done separately elsewhere by the sending entity (Alice). No cryptographic strength is ascribed to this early transformation, it is simply primary conditioning of the plaintext code-point value as the next operand to be used by the cipher proper.

The cryptography is then all about obfuscating this integer 'n' while it is in transit to Bob (the receiving entity) with the full acceptance that it may be intercepted by an adversary (Eve) who is determined to decrypt the ciphertext by every means available to her.

The entities use 'mutually synchronised' databases that enable Bob to call up the exactly same data being used by Alice each time.

The contrived 'entanglement' i.e. the planned, confusing obfuscating methodology being described is based entirely on the algebraic geometry that evolves naturally from my invention of vector factoring although the cipher algorithm makes only slight but important use of only a part of 'vector factoring' in this cryptography.

Scalar take-out factors are not allowed and have been designed out in the cipher algorithm.

A working cipher is up and running on the writer's website at http://www.adacryptpages.com in the title boxes,

"SureCrypt" Cipher - Text Files & Email Encryptions. Windows XP, Vista, 7 & 8.

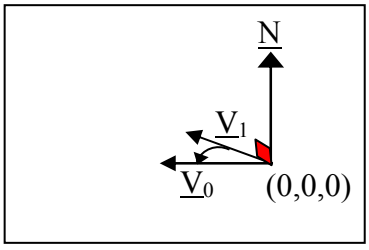
"Special Descriptive Document for Vector Cryptography"

The cipher is designed to encrypt the entire ASCII code set i.e. elements 0 to 255 incl.

How It Works.

Let 'n' be the primary transformation of a plaintext into an integer following the primary transformation, 'n' becomes the operand of the main cipher core algorithm.

Let $\underline{N}$ be the defining normal of a plane that passes through the origin i.e. the plane contains the point (0,0,0).



In this diagram, V_1 and V_0 are the multiplier and the multiplicand respectively in the ‘cross-product’ operation that yields the third vector $\underline{N}$. $\underline{N}$ is always mutually perpendicular to these two in the direction shown when they are multiplied out in that order i.e. $\underline{V}_1 \times \underline{V}_0 = +\underline{N}$. These operands although in pairs are also separate ‘factors’ of $\underline{N}$ i.e. vector factors.

The converse of vector multiplication.

In this cryptography (factoring) is the converse of the vector cross product.

Let $\underline{N} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$ and let $\varepsilon_x \equiv \text{GCD}\ \{\ \beta, \gamma\ \}$

Then,

$\underline{N} = \underline{V}_1 \times \underline{V}_0$ where $\underline{V}_0 = \begin{pmatrix} 0 \\ \gamma/\varepsilon_x \\ -\beta/\varepsilon_x \end{pmatrix}$ and $\underline{V}_1 = \begin{pmatrix} \varepsilon_x \\ y \\ -(\alpha.\varepsilon_x + \beta.y)/\gamma \end{pmatrix}$ are factors.

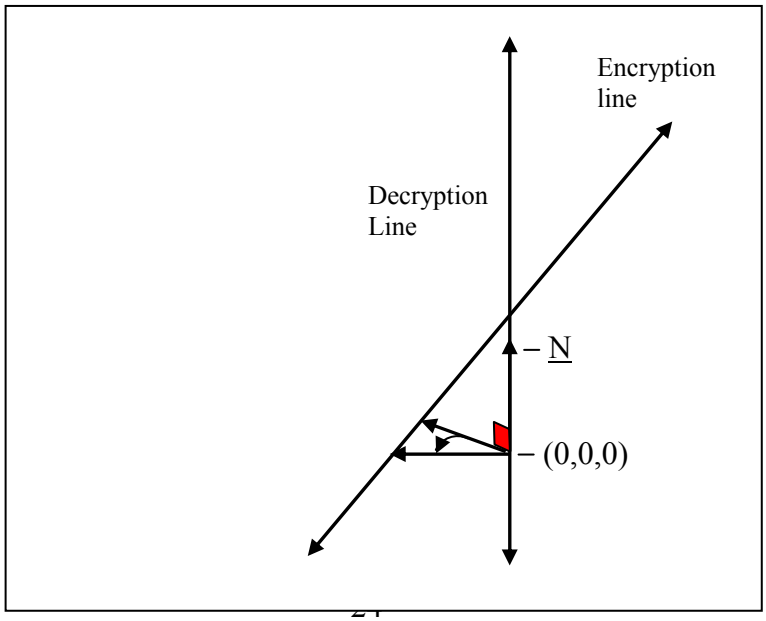
Dedicated number lines for use in cryptography.

$\underline{V}_0$ and $\underline{V}_1$ are used next to define a line that passes through both of them. For the purposes of this cryptography this general line is commissioned as a 'directed' 'number-line' that has specific direction. $\underline{N}$ is *key* material – by means of its factors $\underline{N}$ defines the plane that contains the number-line that Alice uses to encrypt. The factors also define the **direction** of that number line. $\underline{N}$ further defines the direction of another line (the vertical line) that is commissioned as the number-line that Bob uses to decrypt.

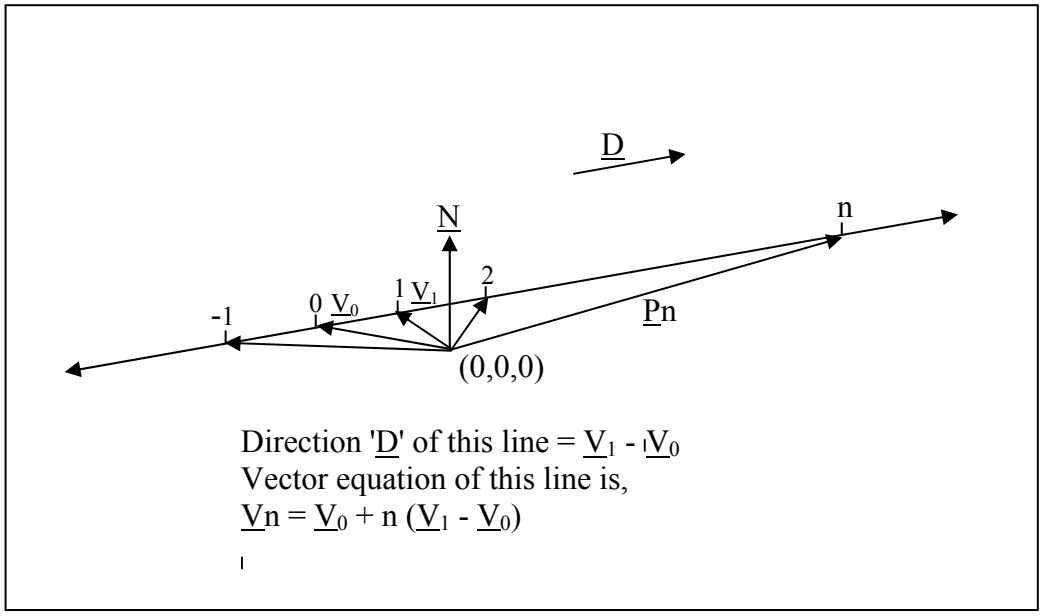
These number lines are both skew and orthogonal (*Orthogonal Skew Lines*) at the same time hence the description sometimes used to describe this cryptography as "Skew Line Encryptions".

Note: This is the core model that is the symmetric basis that is used for both encryption and decryption simultaneously in this cipher.

There is a very happy coincidence to this cryptography in 1) the perpendicularity of the vector cross-product, 2) the orthogonality of the skew lines, 3) the correlation of the factors of $\underline{N}$, i.e. $\underline{V}_0$ and $\underline{P}$n respectively both being factors of $\underline{N}$ that together enable Bob to transform by rotation any number (n) that Alice assigns to her encryption number line on to his own decryption line and evaluate it before decoding 'n' back into the plaintext character that Alice wishes him to know.



Encryption theory.



Alice computes the particular 'n' corresponding to the plaintext in hand in her usual preliminary transformation. She next assigns this 'n' to the number line and calls it Pn i.e. the number that has position vector Pn.

Pn is the position vector of 'n' *relative to the conventional origin at (0,0,0).* Alice doesn't like this origin because it is too transparent to Eve (the interloper) so she colludes with Bob to express 'n' as being relative to a different point (x,y,z). This is quite permissible mathematically. In other words Alice gives a change-of-origin to the position vector that defines the displacement of 'n'.

The new position vector of 'n' relative to (x,y,z) is,

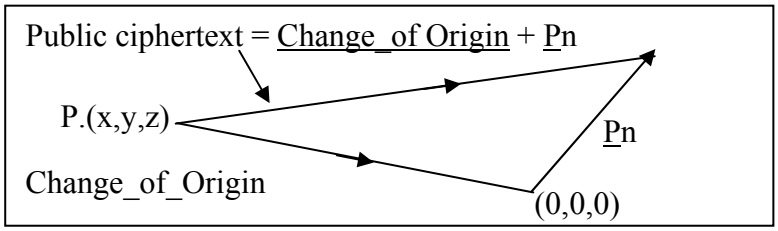
Change-of -Origin + Pn

This ploy gives the entities a powerful extra cipher key.

The public ciphertext therefore is;

Ciphertext = Change-of -Origin Vector + Pn

Graphically.



Comment.

Effectively it could be said that the ciphertext is computed initially from the true origin and is finalised from a confusing 'off-set' origin. The amount of offset is a vector called "Change-of-Origin" – a powerful key that is known only to the entities and is totally impossible for anybody not-in-the-know to deduce by any means whatever.

Collecting the important keys.

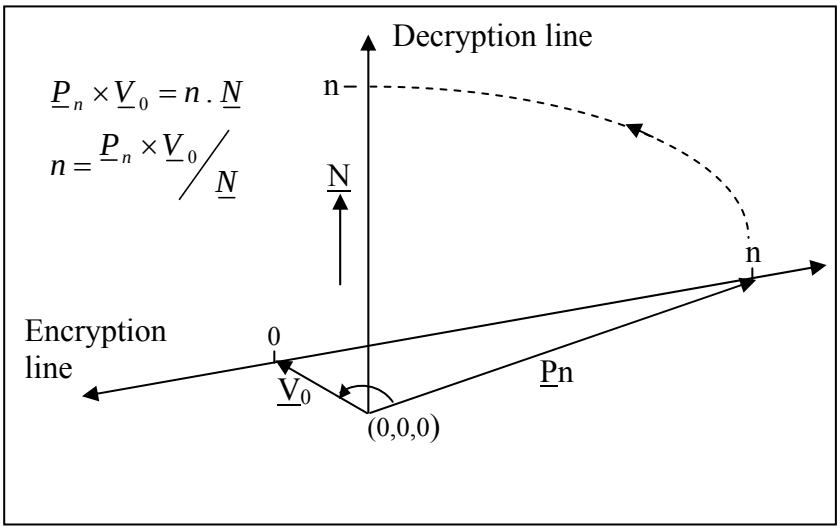
N is a prepared key (1000 stored in an array)

Change-of-Origin is a prepared key (50,000 stored in an array)

Pn is a variable key (evolves in real-time from computation of variables)

$\mathbf{V_0}$ is a variable key (evolves from varying N's that are called at runtime.)

Decryption theory.



Recapping.

Encryption,

Ciphertext = Change-of-Origin vector + Pn

Decryption,

$$\underline{P}_n \times \underline{V}_0 = n \,.\, \underline{N}$$

$$n = \underline{P}_n \times \underline{V}_0 \Big/ \underline{N}$$

'n' is decoded back into the plaintext that Alice wishes Bob to know.

Cryptanalysis.
Let us say that Eve the adversary has somehow intercepted a large slice of the ciphertext string while it is in transit to Bob and is busy trying to decrypt it.

Each item of ciphertext is an independent 'Resultant' vector. It is the sum of two vectors namely a Change-of-origin vector and a Pn. Before all else Eve must correctly resolve this item of ciphertext into the *exact* pair of components that was used to create it. Her task is impossible and even if she guessed the right pair she has no means of telling this.

Assuming that against all odds Eve has resolved the ciphertext as described she must next correctly guess N i.e. the defining normal of the particular plane that contains Pn, again her task is impossible.

Indirect attack.
This means a statistical attempt at mapping ciphertext to plaintext on the basis of characteristic frequency of the English language. This attack is completely foiled by the fact that the ciphertext string is studiously made totally non-repeating, Eve (the adversary) has nothing to work on statistically in effect in that respect.

Class.
On the face of this impossible intractability, the claim here is that this cryptography is in the ultimate class of "theoretically Unbreakable" cryptographic strength.

Criticism.
Having three large (seven-digit) coefficients to each item of ciphertext the 'ciphertext expansion ratio' (plaintext to ciphertext volumetric ratio) is high and this would have been a problem in days past but given the cheapness of modern computer power and the cheapness of the extra memory required it is not considered a serious problem today. Instead it is considered to be only a small price to pay for unbreakable cryptography.

Appendix_A - a worked example.

Let us say that on this occasion Alice wants to encrypt the character Q (Capital Q).

Q has the decimal value of 81 in ASCII => n = 81

Let us say that she chooses the normal vector $\underline{N} = \begin{pmatrix} 7 \\ 2 \\ 6 \end{pmatrix}$.

$\alpha = 7, \beta = 2, \gamma = 6$. $\varepsilon_x = 2$

$$\underline{V}_0 = \begin{pmatrix} 0 \\ 6/2 \\ -2/2 \end{pmatrix} = \begin{pmatrix} 2 \\ 3 \\ -1 \end{pmatrix}$$

and $\underline{V}_1 = \begin{pmatrix} 2 \\ y \\ -(7.2 + 2.y/6) \end{pmatrix}$

Putting y = 5 => z = $-(14+10)/6 = -4$

$$\underline{V}_1 = \begin{pmatrix} 2 \\ 5 \\ -4 \end{pmatrix}$$

Always check this vector cross-product,

Ensure that $\underline{V}_1 \times \underline{V}_0 = \underline{N}$

$$\begin{pmatrix} 2 \\ 5 \\ -4 \end{pmatrix} \times \begin{pmatrix} 0 \\ 3 \\ -1 \end{pmatrix} = \begin{pmatrix} 7 \\ 2 \\ 6 \end{pmatrix}$$ (yes it's ok)

Encryption.

Reminder: Alice's encryption line is $\underline{V}n = \underline{V}_0 + n\,(\underline{V}_1 - \underline{V}_0)$

$$\underline{V}_{81} = \begin{pmatrix} 0 \\ 3 \\ -1 \end{pmatrix} + 81. \begin{pmatrix} 2 \\ 2 \\ -3 \end{pmatrix}$$

i.e. $\underline{P}n = \begin{pmatrix} 162 \\ 165 \\ -244 \end{pmatrix}$

Let us say, Alice decides that the Change-of-Origin vector is $\begin{pmatrix} 217 \\ 371 \\ 284 \end{pmatrix}$

Ciphertext is $\begin{pmatrix}217\\371\\284\end{pmatrix} + \begin{pmatrix}162\\165\\-244\end{pmatrix} = \begin{pmatrix}379\\536\\40\end{pmatrix}$

Decryption.

Bob removes the Change-of -Origin vector.

$$\begin{pmatrix}379\\536\\40\end{pmatrix} - \begin{pmatrix}217\\371\\284\end{pmatrix} = \begin{pmatrix}162\\165\\-244\end{pmatrix}$$

$\underline{P}n = \begin{pmatrix}162\\165\\-244\end{pmatrix}$

Then,

$\underline{P}n \times \underline{V}_0 = n \,.\, \underline{N}$

$n = \underline{P}_n \times \underline{V}_0 \,/\, \underline{N}$

$$n = \begin{pmatrix}162\\165\\-244\end{pmatrix} \times \begin{pmatrix}0\\3\\-1\end{pmatrix} \div \begin{pmatrix}7\\2\\6\end{pmatrix} = \begin{pmatrix}567\\162\\486\end{pmatrix} \div \begin{pmatrix}7\\2\\6\end{pmatrix}$$

(Vector division is not generally defined in vector methodology but in this case dividing corresponding coefficients is in order because they are known beforehand to be parallel vectors).

$567/7 = 162/2 = 486/6 = 81$ as you would expect.

81 is decoded back into its ASCII element as Q.

Discussion.

The reader will notice that although quite small values of coefficients were conveniently taken for the key $\underline{N}$ in this example the longhand computations quickly became quite difficult in the workings and any visible correlation that might have existed between numbers at the outset was quickly lost as the computation went on. This 'entanglement' in vector cryptography is enormous and helps create tertiary intractability. It always seems rather miraculous to me how the value of 'n' always pans out correctly at the end of the obscure decryption process.

Note also:

In the computation of $\underline{V}_1$,

$$\underline{V}_1 = \begin{pmatrix} 2 \\ y \\ -(7.2 + 2.y/6) \end{pmatrix}$$

Putting y = 5 => z = $-(14+10)/6 = -4$

y decides z in this expression and one has to keep on testing different values of 'y' until an integer value for 'z' emerges, 5 in this case. It is not mandatory to accept the first value that produces an integer value of z but that is usually done. Other values are quite OK.

When the numbers are small 'y' can be found empirically by longhand testing with reasonable ease but when the coefficients of $\underline{N}$ are large this can be become horrendously difficult. The cipher computer program does it all effortlessly of course and is very speedy overall. The reader has nothing to worry about.

An encryption/decryption rate of about 38000 characters of plaintext per second is about par for the course on my typical home computer.

Austin O'Byrne

Appendix_D. Discussion.

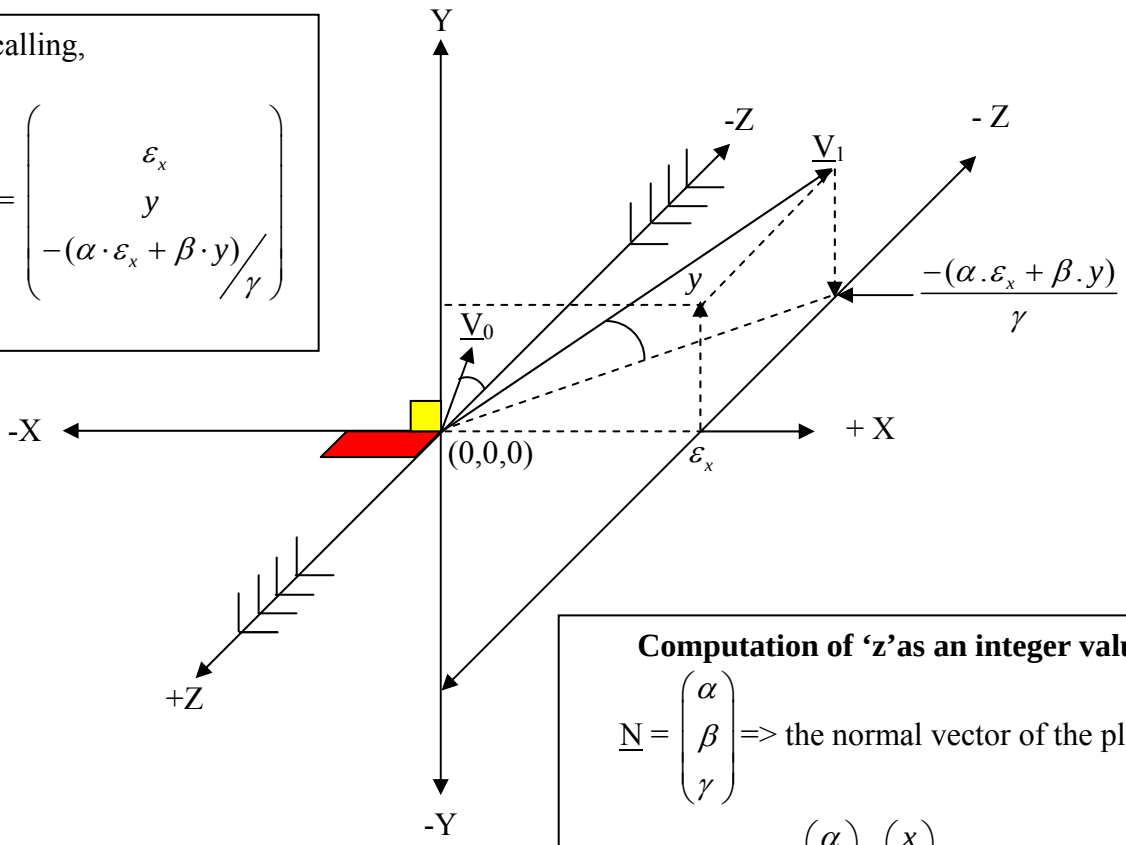
This application to cryptography ushers in the physical analogue '*displacement*' in three-dimensional space as a future basis for main stream encryption algorithms that use vector methodology. This is not the only application of vector factoring and it is very likely that there will be other currently unforeseen uses for it in the sciences and technologies that use vectors a lot such as vector mechanics, in the future. It is thought to have potential as a tool for design engineers and physicists. It is thought also by me that this cipher could be used as a defining rule in what is to be called "Theoretically Unbreakable" cryptography in the future i.e. having a demonstrably infinite key-space as a design condition . Anything else must be taken as being brute-force-able because of having a *finite* key space.

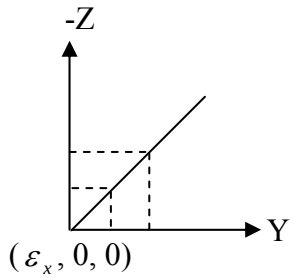
Appendix E:-

Demonstrating V_0 and V_1 Graphically.

Recalling,

$$\underline{V}_1 = \begin{pmatrix} \varepsilon_x \\ y \\ -(\alpha \cdot \varepsilon_x + \beta \cdot y)/\gamma \end{pmatrix}$$





Integer y for integer z

Computation of 'z'as an integer value

$\underline{N} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$ => the normal vector of the plane

$$\begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} = 0$$

$\alpha.\varepsilon_x$ is known , and let 'y 'vary in $\beta \cdot$y then,

$$z = -(\alpha.x + \beta.y)/\gamma$$

(i.e. y decides z here)

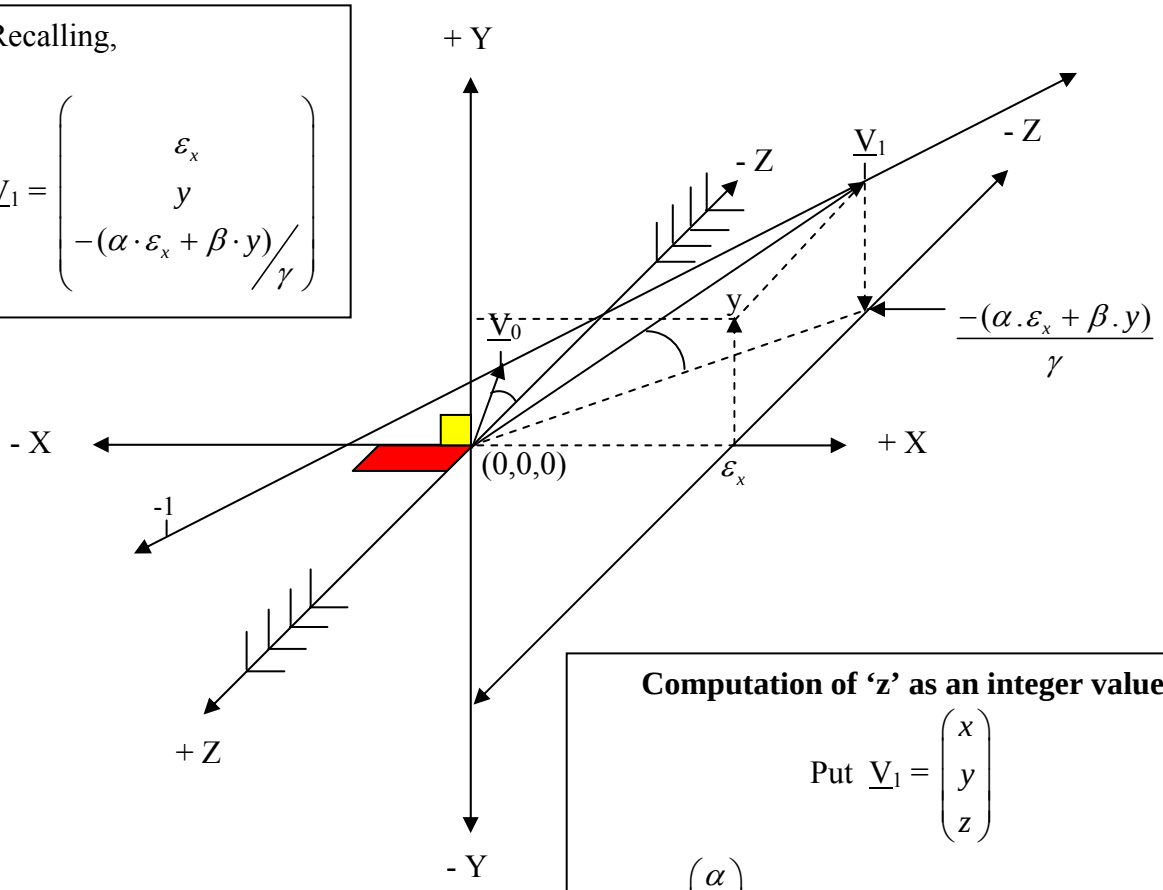
'y' is progressively increased (or decreased) in steps of 1 in the computer program until the outcome of the equation is an integer value on the Z axis. Any value of 'y' that does this is in order – not necessarily the first. The only thing to watch is that the ensuing computations at a later stage of the encryption/decryption process do not become too large for the computer to store as integers.

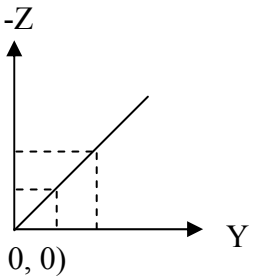
Demonstrating V_0 and V_1 Graphically.

V_0 and V_1 & a number-line that they might define

Recalling,

$$\underline{V_1} = \begin{pmatrix} \varepsilon_x \\ y \\ -(\alpha \cdot \varepsilon_x + \beta \cdot y)/\gamma \end{pmatrix}$$





Integer y for integer z

Computation of 'z' as an integer value

Put $\underline{V_1} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$

$\underline{N} = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$ => the normal vector of the plane

$$\begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} = 0$$

$\alpha.\varepsilon_x$ is known , and let 'y 'vary in $\beta \cdot$y then,

$$z = -(\alpha.x + \beta.y)/\gamma$$

(i.e. y decides z here)

'y' is progressively increased (or decreased) in steps of 1 in the computer program until the outcome of the equation is an integer value on the Z axis. Any value of y is in order – not necessarily the first. The only thing to watch is that the ensuing computations at a later stage of the encryption/decryption process do not become too large for the computer to store as integers.

Vector Cryptography – Discussion Model

$$Let\ \underline{N} = (\alpha.\hat{i} + \beta.\hat{j} + \gamma.\hat{k}) \equiv \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$$

Let, $\varepsilon_x = GCD\{\beta, \gamma\}$

$\varepsilon_y = GCD\{\alpha, \gamma\}$

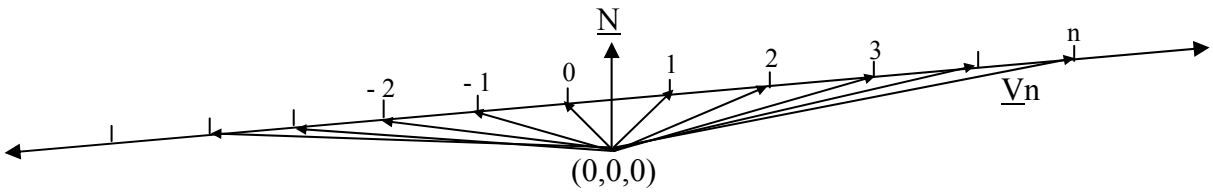
$\varepsilon_z = GCD\{\alpha, \beta\}$

$Let\ \alpha: +ve\ \text{int}, 10 <= \alpha <= 999$

$Let\ \beta: +ve\ \text{int}, 10 <= \beta <= 999$

$Let\ \gamma: +ve\ \text{int}, 10 <= \gamma <= 999$

Encryption Line.



$\underline{Vn} = \underline{V_0} + n(\underline{V_1} - \underline{V_0})$

$\underline{N}$ kick-starts the encryption transformation, it is used only once before being replaced for the next item.

$\underline{V_0}$ is the position vector of the number '0' on the directed number-line shown.

$\underline{V_1}$ is the position vector of the number '1' on the directed number-line shown.

The number line here is defined by the fact that it passes through these two known points.

$\underline{V_0}$ and $\underline{V_1}$ are a primary or 'seeding' pair of factors of $\underline{N}$.

$\underline{V_1}$ (pronounced VeeOne) is the cofactor of $\underline{V_0}$ (VeeZero) such that $\underline{V_1} \times \underline{V_0} = \underline{N}$ when multiplied out in that order in the vector or 'cross' product method of vector multiplication.

Subsequently, any pair $\underline{Vn}$ and V_{n-1} on any factor line (related lines derived later) are vector factors of $\underline{N}$ when taken in this same order.

Although discussed as pairs of factors here, $\underline{V_0}$ and $\underline{V_1}$ are each distinct 'standalone' factors of $\underline{N}$ in their own right.

There are three options for $\underline{V_0}$ and six options for $\underline{V_1}$ in defining equations of derived factor lines.

The integer representation (codepoint) of each plaintext currently being encrypted is assigned to this line; the position vector of the number on the line then becomes a reversible analogue of the number in question.

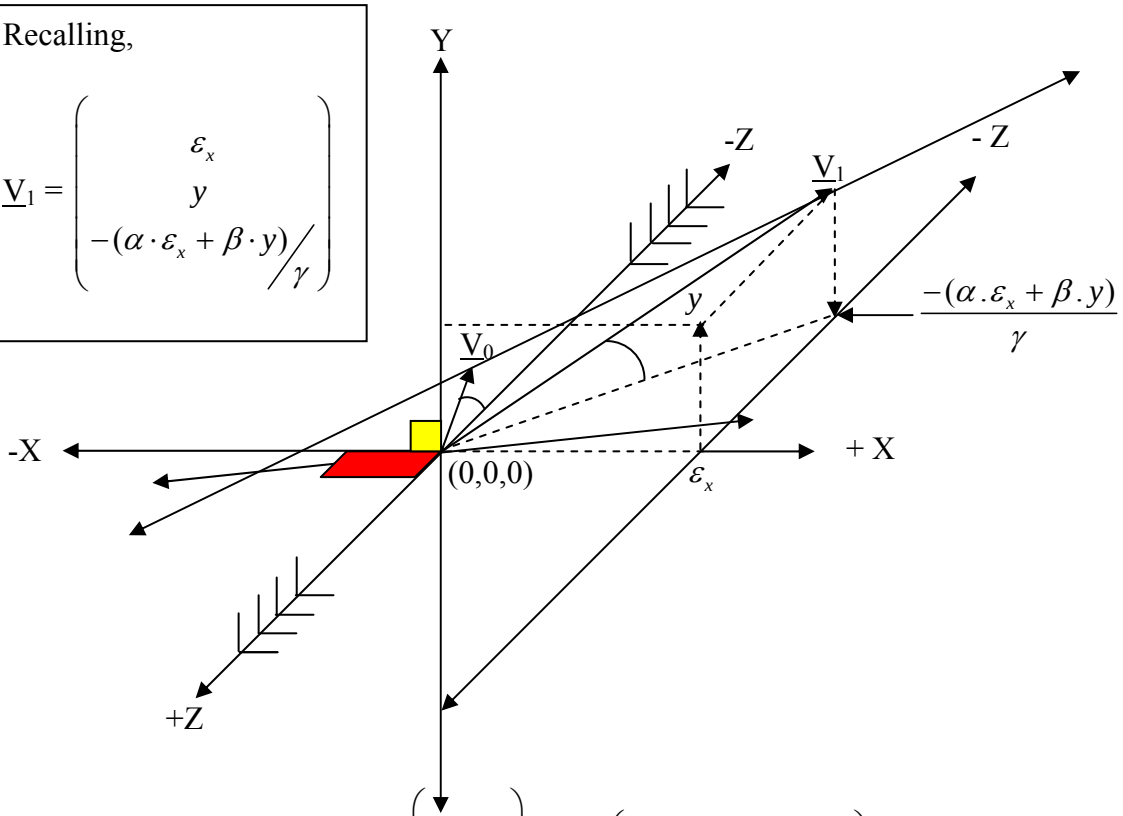
The line through $\underline{N}$, i.e. the line that has the same direction as $\underline{N}$ and the inclined number line now comprise a pair of orthogonal skew lines. The upshot of this fact is that the vector or cross product of pairs of position vectors of numbers on the inclined number line (the exact details of which need not be known) can conveniently be used to rotate the important information, i.e.the 'n' of Pn, onto the pairing skew line through $\underline{N}$ where it can be easily read. The vector or crossproduct has a characteristic transforming effect in that the resulting product of the multiplier and the multiplicand of two vectors is always a third vector that is at 90 degrees to the plane of the first two when they are all taken in the correct order of a right-handed triad. This latter caveat is essential in order to qualify uniquely.

Vector cryptography makes heavy use of this very useful fact.

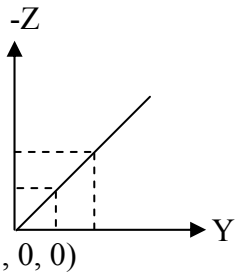
Demonstrating V_0 and V_1 Graphically. And a Number Line They Might Define.

Recalling,

$$\underline{V}_1 = \begin{pmatrix} \varepsilon_x \\ y \\ -(\alpha \cdot \varepsilon_x + \beta \cdot y)/\gamma \end{pmatrix}$$



$$\underline{V}_0 = \begin{pmatrix} -Y\,0 \\ \gamma/\varepsilon_x \\ -\beta/\varepsilon_x \end{pmatrix} \quad \underline{V}_1 = \begin{pmatrix} \varepsilon_x \\ y \\ -(\alpha.\varepsilon_x + \beta.y)/\gamma \end{pmatrix} \text{ or } \underline{V}_1 = \begin{pmatrix} \varepsilon_x \\ -(\alpha.\varepsilon_x + \gamma.z)/\beta \\ z \end{pmatrix}$$



Integer y for integer z

$$\underline{V}_0 = \begin{pmatrix} -\gamma/\varepsilon_y \\ 0 \\ \alpha/\varepsilon_y \end{pmatrix} \quad \underline{V}_1 = \begin{pmatrix} -(\beta.\varepsilon_y + \gamma.z)/\alpha \\ \varepsilon_y \\ z \end{pmatrix} \text{ or } \underline{V}_1 = \begin{pmatrix} x \\ \varepsilon_y \\ -(\beta.\varepsilon_y + \alpha.x)/\gamma \end{pmatrix}$$

Equations of lines that may be used as factor lines.

$\underline{V}n = \underline{V}_0 + n(\underline{V}_1 - \underline{V}_0)$

$\underline{V}n = \underline{V}_0 + n(\underline{V}_1 + \underline{V}_0)$

$\underline{V}n = \underline{V}_0 + n(\underline{V}_1)$

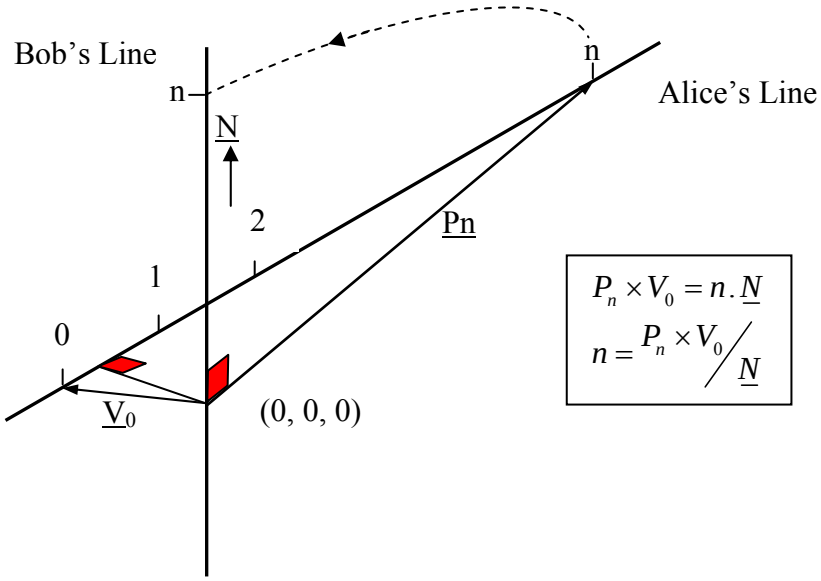
$\underline{V}n = \underline{V}_1 + n(-\underline{V}_0)$

$$\underline{V}_0 = \begin{pmatrix} \beta/\varepsilon_z \\ -\alpha/\varepsilon_z \\ 0 \end{pmatrix} \quad \underline{V}_1 = \begin{pmatrix} x \\ -(\gamma.\varepsilon_z + \alpha.x)/\beta \\ \varepsilon_z \end{pmatrix} \text{ or } \underline{V}_1 = \begin{pmatrix} -(\gamma.\varepsilon_z + \beta.y)/\alpha \\ y \\ \varepsilon_z \end{pmatrix}$$

The Talking Transformer.

This is a salient parody on the way in which Alice and Bob communicate using vector cryptography. That is by means of a geometric tool used for 'distance communication' in a vector cipher. Alice's parameters are number inputs to the 'tool' that enable Bob to understand what she is saying cryptographically. The scheme makes use of vector factoring and orthogonal skew lines.

The model is a pair of orthogonal skew lines that are related to each other in the amount of skew by means of the factoring of vectors methodology already described.



Any number that Alice puts or her line, Bob can transfer and 'read' it by rotating it on to his own line. The mathematical transfer operator that enables this to happen is the simple cross-product. It is very elegant use of the perpendicularity property.

Alice creates a line to her liking and factorises it etc. and finds Pn for the current character that she is enciphering. She gives this a hefty change-of-origin and sends it to Bob (Please see the Discussion Model for a view of her cipher-text string). Bob removes the change-of-origin to uncover the real origin and then,

$$\underline{P}n \times \underline{V}_0 = n.\underline{N}$$

(in words just to clarify this, Pn cross VeeZero = n times N)

Dividing corresponding coefficients left and right of the equals sign gives 'n' for decoding back to its ASCII evaluation by Bob.

Discussion

Theoretically Unbreakable Class

Additional to being new, this cryptography is also meant to be a demonstration of cryptography that is in the ultimate class of "Theoretically Unbreakable". That claim requires some explanation to the reader straight away.

There is no definition to be found anywhere in mathematics that defines a 'One-Way' function. The relevance of this to theoretically unbreakable cryptography is that there is no handy benchmark that one can invoke to support a parallel, endless state in cryptology.

I am going to put my head on the euphemistic block and say that theoretically unbreakable cryptography is defined by demonstrating that it uses at least one *infinitely large key space because any key space less than this is breakable by traditional brute force methods (exhaustive testing of every key in a finite key space).

It can be demonstrated that vector cryptography uses two infinitely large key spaces. These respectively are 1) her infinitely large selection domain of normal vectors that Alice (pseudonym for the sending entity) can use to define construction planes and 2) her infinite selection domain of change-of-origin vectors that she may use in her computed ciphertext to confound cryptanalysis by adversaries.

On the back of these facts I have no compunction in claiming that my Vector Cryptography belongs in the ultimate class of "Theoretically Unbreakable".

A downside of this cryptography.

The ciphertext to plaintext expansion ratio is high.

Currently, this rather high at as much as 20 to 1

The maximum normal ratio would not be expected more than about 10 to 1.

That however is not an insurmountable drawback and is simply a problem to be solved – several solutions are in the melting pot.

Appendix_F. - Some Vector Arithmetic Used here.

A vector written horizontally in component form may conveniently be also written in vertical column form:

$$\underline{V} = \hat{i} + \hat{j} + \hat{k} \text{ is the same as } \underline{V} = \begin{pmatrix} i \\ j \\ k \end{pmatrix}$$

In these examples $\underline{V}_1 = \begin{pmatrix} a \\ b \\ c \end{pmatrix}$ and $\underline{V}_2 = \begin{pmatrix} d \\ e \\ f \end{pmatrix}$

Vector Multiplication or the Vector Product of two vectors is so-called because the product is always a vector (this form of multiplication is also called the cross product). Importantly this operation is non-commutative. That means the operands must be written in a particular order to get the correct result.

It means that ($\underline{V}_1$ x $\underline{V}_2$) /= ($\underline{V}_2$ x $\underline{V}_1$)

When the multiplicand and the multiplier are changed around then the product becomes the (–ve) of what it was prior to the change so that,

($\underline{V}_2$ x $\underline{V}_1$) = - ($\underline{V}_1$ x $\underline{V}_2$)

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} \text{x} \begin{pmatrix} d \\ e \\ f \end{pmatrix} = \begin{pmatrix} (bf - ce) \\ -(af - cd) \\ (ae - bd) \end{pmatrix}$$

A useful rule for finding the Cross Product directly by inspection,

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} \text{x} \begin{pmatrix} d \\ e \\ f \end{pmatrix} = \begin{pmatrix} - \\ - \\ - \end{pmatrix}$$

To find a coefficient (-) on the RHS cover the corresponding row on the LHS (use your ballpoint pen to do this) and mentally solve the determinant formed by the remaining two rows. Don't forget the change of sign with the middle one.

Example,

$$\begin{pmatrix} 4 \\ 2 \\ -1 \end{pmatrix} \text{x} \begin{pmatrix} 3 \\ -2 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ -7 \\ -14 \end{pmatrix}$$

Multiplication of a vector by a scalar.

Simply multiply each coefficient by the scalar,

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} \text{ x 5 (say)} = \begin{pmatrix} (5a) \\ (5b) \\ (5c) \end{pmatrix}$$

Scalar multiplication or Dot Product of two vectors.

Simply multiply corresponding coefficients and add the three products.

$\begin{pmatrix} a \\ b \\ c \end{pmatrix} \bullet \begin{pmatrix} d \\ e \\ f \end{pmatrix}$ = (ad) + (be) + (cf) Example, $\begin{pmatrix} 3 \\ 5 \\ 8 \end{pmatrix} \bullet \begin{pmatrix} -7 \\ 2 \\ 1 \end{pmatrix}$ = -21 + 10 + 8 = -3

(note: the dot product is always a scalar)

Addition of two Vectors

Simply add corresponding coefficients,

$\begin{pmatrix} a \\ b \\ c \end{pmatrix} + \begin{pmatrix} d \\ e \\ f \end{pmatrix} = \begin{pmatrix} (a+d) \\ (b+e) \\ (c+f) \end{pmatrix}$ Example, $\begin{pmatrix} 6 \\ 3 \\ 1 \end{pmatrix} + \begin{pmatrix} 2 \\ 3 \\ -2 \end{pmatrix} = \begin{pmatrix} 8 \\ 6 \\ -1 \end{pmatrix}$

Subtraction of two vectors

$\begin{pmatrix} a \\ b \\ c \end{pmatrix} - \begin{pmatrix} d \\ e \\ f \end{pmatrix} = \begin{pmatrix} (a-d) \\ (b-e) \\ (c-f) \end{pmatrix}$ Example, $\begin{pmatrix} -4 \\ 6 \\ 9 \end{pmatrix} - \begin{pmatrix} -2 \\ 4 \\ 5 \end{pmatrix} = \begin{pmatrix} -2 \\ 2 \\ 4 \end{pmatrix}$

Vector Division is Undefined

There is no general algorithm for it in everyday mathematics.

Austin O'Byrne.

October 2015.