Geometric Algebra Algorithm representing an Orthogonal Isomorphism in versor form v1.

Allan Cortzen, c@ctz.dk

Abstract

Given an orthogonal isomorphism as a linear transformation of an n-dimensional vector space with indefinite regular metric the goal is to construct an algorithm for a versor representing the orthogonal isomorphism without exceptions.

Introduction Preliminaries

The symbolic is the same as in [1], and the reader is assumed familiar with some of the essential results from this book.
Let $\mathcal{G}(\mathbb{R}^{p,q})$ be the regular geometric algebra over $\mathbb{R}^{p,q}$, where (p, q) is the signature (p positive) and $n = p + q$ the dimension of this vector space. We denote reversion by $R^{\sim}$ and grade involution by $R^{\wedge}$.
For an orthogonal isomorphism ψ of $\mathbb{R}^{p,q}$ the algorithm calculates V in $\psi(x) = V^{\wedge}\, x\, V^{-1}$.
The algorithm is not meant as an actual speed optimized numerical implementation.

General considerations

The versor V is constructed from simple reflections and rotations. In calculations we avoid reflections $a \to b$, when a and b are nearly identical, and rotations $a \to b$, when a and b are nearly opposite. Such transformations may give a serious precision loss.
A reflection along may give loss of relative accuracy for a close to b due to the subtraction.

The versor formula for a reflection along u is $x \to -u\, x\, /u$, where $u \neq 0$ is an invertible vector in $\mathbb{R}^{p,q}$.
If $(a-b)^2 \neq 0$, a reflection along $a - b$ maps $a \to b$, as $-(a-b)\, a = b\,(a-b)$, and thus $-(a-b)\, a\, /(a-b) = b$.
Such a reflection is only used for $\left|(a-b)^2\right| \geq \delta$ to avoid too much loss of relative accuracy.

Now assume $\left|(a-b)^2\right| < \delta$. Then a rotation $a \to b$ is used constructed from a reflection $a \to -b$ followed by a reflection along b.
Let $0 < \delta \leq 2$. Smaller δ gives fewer of the more calculation costly rotations.
Let $\kappa = a^2 = b^2 = \pm 1$. Then $(a \pm b)^2 = 2\,(\kappa \pm a \cdot b)$ and $(a-b)^2 + (a+b)^2 = 4\,\kappa$.
As $\kappa^2 = 1$, $(a-b)^2 \in (-\delta, \delta) \Rightarrow (a+b)^2\, \kappa = 4 - \kappa(a-b)^2 \in (4-\delta, 4+\delta) \Rightarrow \left|(a+b)^2\right| > 2$.
Thus $a + b$ is a versor without special numeric problems, and the rotation $a \to b$ is with done with versor $b\,(a+b)$.

The pseudo code for the algorithm

The algorithm do not require $a_i^2 = b_i^2$ to be normalized.
The number $\delta \in (0, 2]$ is selected according to a choice between precision and speed.

```
V = 1
for i = 1 to n
   R = a_i − b_i
   d = |R^2|
   if d ≥ δ |b_i^2|   (* Reflection part *)
      for k = i + 1 to n
         a_k = −R a_k / R
   else
      if i == n
         break
      r = a_i + b_i      (* The rotation part *)
      R = b_i r
      for k = i + 1 to n
         a_k = R a_k / R
   V = R V
```

The result is a versor V for ψ.
As shown later, loosing max. $s < r$ decimals of significance in $a_i - b_i$ is secured by choosing $\delta = n \cdot 10^{-2s}$.

Proof of correctness

Let B_i be the orthogonal subspace to $(b_1, \ldots, b_{i-1})$, i. e. $B_i = \text{span}\{b_i, \ldots, b_n\}$.
First disregard the part "**if** $i == n$, **break**".
We use the following loop invariant of the first **for** loop:

At the start of each iteration $\Theta(i): \ (a_k = b_k \text{ for } k < i) \text{ and } (a_k \in B_i \text{ for } k \geq i)$

Initialization: There is nothing to prove, as $B_1 = \mathbb{R}^{p,q}$.
Maintenance: We assume $\Theta(i)$ just before the ith iteration. Thus $(a_k \in B_i \text{ for } k \geq i)$
In this loop R is is constructed from vectors in B_i, and such that $a_i \to b_i$ by $f_i(x) = \pm R\, x / R$. As f_i fixes vectors orthogonal to B_i, it also fixes $b_k = a_k$ for $k < i$, and calculation can be avoided.
At the end of the loop $\Theta(i+1)$ is satisfied, as $a_i = b_i$ and, when $k \geq i+1$, $a_k \in B_i$ and is orthogonal to $a_i = b_i$.
Notice that V is the total transformation versor of the original (a_i) frame.
Termination: At termination the original (a_i) frame is transformed with V, and the result is the (b_i) frame

The part "**if** $i == n$, **break**" is just a simplification:
$\Theta(n)$ implies $a_n \in B_n$ or $a_n = \pm b_n$. If $a_n = b_n$, then the the rotation part is superfluous, and $a_n = -b_n$ is not possible, as the reflection along b_n has been effectuated, since $|\,d| = 4 \geq \delta$.

Norm definition and application to loss of significance by subtraction $a - b$ alone.

Let (e_i) be an orthonormal frame e.g. (b_i). An euclidean metric is defined by (e_i) being euclidean orthonormal. This gives a norm such that $x = \Sigma \beta_i\, e_i \Rightarrow \|x\|^2 = \Sigma \beta_i^2$, and obviously, as $\Sigma \beta_i^2 \geq \Sigma\, e_i^2\, \beta_i^2$,

$$\|x\|^2 \geq |x^2|.$$

That coordinates and this norm are *relative to the same orthonormal frame* is essential in indefinite metrics.

Loss of significance by subtraction $a - b$ alone.
Assume the unit vectors a, b are represented with uncertainty $\xi = 0.5 \cdot 10^{-r}$, that is $\|\Delta\, a\| = \|\Delta\, b\| = \xi$. Then $\|\Delta(\, a - b)\| \leq 2\, \xi$.

As $||a-b||^2 \geq (a-b)^2 \geq \delta$, the relative uncertainty for $a-b$ in the worst case is $2\,\xi \big/ \sqrt{\delta}$.

Loosing max. $s<r$ decimals of significance is secured by choosing $\delta = 4\cdot 10^{-2s}$ in the algorithm, as $2\,\xi \big/ \sqrt{\delta} \leq 10^s\,\xi \Leftrightarrow \delta \geq 4\cdot 10^{-2s}$.

Conclusion

If it is known from start exactly that $a_i = b_i$ for some indexes these can be omitted.

It is also possible also from a later step, but "exactly" may not be meaningful.

The difference degradation has been brought under control, which is important for general applications.

The algorithm work for all regular signatures.

Appendix

Truncated algorithm

The algorithm has a version without rotation using a norm condition for approximate equality and valid for *definite spaces* only. It corresponds to algorithm 1 in [2]. Here $x^2 = ||x||^2,\ x \in \mathbb{R}^{p,q} = \mathbb{R}^n$.

$V = 1$
for $i = 1$ **to** n
  $R = a_i - b_i$
  if $R^2 \geq b_i{}^2\,\delta$ (* Reflection part*)
    for $k = i+1$ **to** n
      $a_k = -R\,a_k/R$
  (***else** nothing, as a_i and b_i are considered equal. *)
  $V = R\,V$

The algorithm is of course faster than the full. Even if δ is optimized half of the precision may be lost in certain cases, and otherwise more.

Loss of precision by subtraction $a-b$ alone

Assume that unit vectors a and b has uncertainty $10^{-h},\ h>2$ and that the condition for equality of a and b is $||a-b|| \leq 10^{-k}$, $0<k\leq h$. Then there are two types of degradation, also present in [2, p. 63-78] algorithm 1 and 2.

1. (Difference degradation) Assume $||a-b|| = 1.01\times 10^{-h}$ then the reflection is executed. As the relative uncertainty for $a-b$ is roughly 10^{k-h}, a rotation an angle of this size may be lost.

2. (Equality degradation) Assume $||a-b|| = 0.99\times 10^{-h}$ then e.g. roughly a rotation an angle 10^h may be lost.

The balance point for this dilemma is $k = h/2$. Thus it is not possible to avoid a precision loss of $h/2$ digits.

Probability considerations for the main algorithm

The purpose is to calculate the probability for use of the rotational code of course dependent of δ and certain conditions e.g. uniformity of distribution. It will show that rotations are strongly decreasing with loops are remaining and with decreasing δ.

Binary truncation is close to decimal rounding.

In the definite case loosing decimals by subtraction $R = a_i - b_i$ are given in the table and found from the formula last in this paragraph:

decimals	δ	Probability for rotation choice	
max ½ decimal	$\delta = 0.4$:	$(0.10,\ 0.025,\ 0.0067)$	when (2,3,4) loops are remaining
max 1 decimal	$\delta = 0.04$:	$(0.032,\ 0.0025,\ 0.00021)$	when (2,3,4) loops are remaining
max 1½ decimal	$\delta = 0.004$:	$\left(0.010,\ 0.00025,\ 6.7\times 10^{-6}\right)$	when (2,3,4) loops are remaining

Thus the occurrence of a rotation is mainly in the second to last loop and here with probability 3.2%, if loosing max 1 decimal.

Derivation of the result.

We assume here (b_i) is the standard basis for $\mathbb{R}^{p,q}$, and use concepts B_i and f_i from the proof section.

It may be of interest to find the rotation probability in loop i relative to the left-invariant Haar measure μ on the orthogonal group $\mathbb{G} = O(\mathbb{R}^{p,q})$, $n = p + q \geq 2$. Thus, if $g \in \mathbb{G}$, and $\mathbb{A}$ is a borel subset of $\mathbb{G}$, then $\mu(g\,\mathbb{A}) = \mu(\mathbb{A})$.

On elements in the product of unit surfaces $S_m = \left\{x \in \mathbb{R}^{p,q} \,\middle|\, \left|x^2\right| = 1\right\}^{\times k}$, denoted $\underline{x} = (x_1,\ \ldots,\ x_m)$, we have a natural left group action, $(g,\ \underline{x}) \to g\,\underline{x} = (g\,x_1,\ \ldots,\ g\,x_m),\ \ g \in \mathbb{G}$

The image of μ by the continuous mapping $f_0 : h \to h\,\underline{a},\ \ h \in \mathbb{G}$ gives a measure μ_0 on S_n.

If $g \in \mathbb{G}$ and $\mathbb{B}$ is a borel subset of S_n, then we have $\mu_0(g\,\mathbb{B}) = \mu\left(f_0^{-1}(g\,\mathbb{B})\right) = \mu\left(g\,f_0^{-1}(\mathbb{B})\right) = \mu\left(f_0^{-1}(\mathbb{B})\right) = \mu_0(\mathbb{B})$, which shows μ_0 is $\mathbb{G}$ left invariant.

Let $\mathbb{G}_i$ be the subgroup of $\mathbb{G}$ fixing $b_1,\ \ldots,\ b_{i-1}$ and $S^i = \{(b_1,\ \ldots,\ b_i)\}\times S_{n-i}$ and $\mathbb{B}_i$ is a borel subset of S^i.

The transformation f_1, acting on S_n, maps the measure μ_0 onto a measure μ_1 on $S^1 = \{b_1\}\times S_{n-1}$, which is $\mathbb{G}_1$ left invariant, as $\mu_1(g_1\,\mathbb{B}_1) = \mu_0\left(f_1^{-1}(g_1\,\mathbb{B}_1)\right) = \mu_0\left(g_1\,f_1^{-1}(\mathbb{B}_1)\right) = \mu_0\left(f_1^{-1}(\mathbb{B}_1)\right) = \mu_1(\mathbb{B}_1)$.

By induction follows easily:

The transformation f_i, acting on S_n, maps the measure μ_{i-1} onto a measure μ_i on S^i, which is $\mathbb{G}_i$ left invariant, as $\mu_i(g_i\,\mathbb{B}_i) = \mu_{i-1}\left(f_i^{-1}(g_i\,\mathbb{B}_i)\right) = \mu_{i-1}\left(g_i\,f_i^{-1}(\mathbb{B}_i)\right) = \mu_{i-1}\left(f_i^{-1}(\mathbb{B}_i)\right) = \mu_i(\mathbb{B}_i)$.

Precisely if the metric on B_i is definite, μ_i can be normalized to a probability measure on S^i or obviously on S_{n-i} by a bijective projection.

Considerations for $S_m = S_{n-i}$ under definite metrics on B_i.

The unit sphere $\left\{x \,\middle|\, \left|x^2\right| = 1\right\}$ is parametrized with angular coordinates $\varphi_1,\ \ldots\varphi_{m-2},\ |\varphi_{m-1}| \in [0,\ \pi]$ as

$x_1 = \cos(\varphi_1),\ \ldots,\ x_{m-1} = \sin(\varphi_1)\cdots\sin(\varphi_{m-2})\cos(\varphi_{m-1}),\ x_m = \sin(\varphi_1)\cdots\sin(\varphi_{m-2})\sin(\varphi_{m-1})$ with metric density $g \approx \left|\sin^{m-2}(\varphi_1)\ldots\sin^2(\varphi_{m-3})\sin(\varphi_{m-2})\right|$.

The marginal density for φ_1 is

$$F_m(\varphi) = \frac{1}{F_m}\int_0^{\varphi}\sin^{m-2}(\varphi_1)\,d\varphi_1,$$

where $F_m = \int_0^{\pi}\sin^{m-2}(\varphi_1)\,d\varphi_1 = \sqrt{\pi}\ \Gamma\left(\frac{m-1}{2}\right)\big/\Gamma\left(\frac{m}{2}\right)$,

valid also for $|\varphi_1|$, if $m = 2$. The angle to be used in F is $\varphi = \arcsin\sqrt{\delta - \delta^2/4}$.

References

[1] L. Dorst, D. Fontijne, and S. Mann. Geometric Algebra for Computer Science.
An Object-Oriented Approach to Geometry. Morgan Kaufman, 2007.

[2] Leo Dorst and Joan Lasenby, editors. Guide to Geometric Algebra in Practice. Springer London, 2011.