Introduction to Geometric Algebra

Extra II

Leandro A. F. Fernandes
laffernandes@inf.ufrgs.br

Manuel M. Oliveira
oliveira@inf.ufrgs.br

Visgraf - Summer School in Computer Graphics - 2010

CG UFRGS

.inf INSTITUTO DE INFORMÁTICA UFRGS

UFRGS UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

CNPq Conselho Nacional de Desenvolvimento Científico e Tecnológico

Extra II

Implementation Approaches

Implementation approaches

- Isomorphic matrix algebras
  - All elements become 2^n x 2^n matrices
  - The outer product and the contractions are not isomorphic to matrix algebra
- Irreducible matrix implementation
  - It is like the isomorphic matrix algebra, but using smaller matrices

Implementation approaches

- Factored representation
  - k-Blades and k-versors are stored as lists of k vectors
  - It seems a viable for high-dimensional algebras
- Multivector representation
  - 2^n coefficients
  - The number of basic operations is quite large
  - Blades and versors are sparse multivectors

Representing unit basis blades with bitmaps

Basis Blade	Index (Decimal)	Bitmap (Binary)
1	0	0000_b
$\mathbf{e}_1$	1	0001_b
$\mathbf{e}_2$	2	0010_b
$\mathbf{e}_1 \wedge \mathbf{e}_2$	3	0011_b
$\mathbf{e}_3$	4	0100_b
$\mathbf{e}_1 \wedge \mathbf{e}_3$	5	0101_b
$\mathbf{e}_2 \wedge \mathbf{e}_3$	6	0110_b
$\mathbf{e}_1 \wedge \mathbf{e}_2 \wedge \mathbf{e}_3$	7	0111_b
$\mathbf{e}_4$	8	1000_b
$\vdots$	$\vdots$	$\vdots$

Geometric and outer product of basis blades in Euclidean metric

```
Input: coefficient₁, coefficient₂, bitmap₁, and bitmap₂

If computing the outer product and (bitmap₁ & bitmap₂) != 0
    Return 0.0, and 0_b

// Compute the resulting bitmap
bitmap = bitmap₁ ^ bitmap₂

// Compute the sign change due to reordering
sign = canonical_reordering(bitmap₁, bitmap₂)

Return (sign * coefficient₁ * coefficient₂), and bitmap
```

Bitwise boolean "and" (the `&` operator)

Bitwise boolean "exclusive or" (the `^` operator)

Sign change due to reordering of two basis blades into canonical order

```
Input: bitmap₁, and bitmap₂

// Count the number of basis vectors swaps
sum = 0

bitmap₁ = bitmap₁ >> 1

While bitmap₁ != 0 do

    sum = sum + bit_count(bitmap₁ & bitmap₂)

    bitmap₁ = bitmap₁ >> 1

End loop

// + for even number of swaps or - for odd number of swaps
Return ((sum & 1) == 0) ? 1.0 : -1.0
```

Bitwise "shift right"

Extra II

Libraries and Toolkits

Libraries and tookits

- GABLE, by Dorst (Home Page)
  - MATLAB learning environment
  - 3-D Euclidean metric
- GA package for Maple, by Ashdown (Home Page)
  - Non-degenerated signatures
- CLUCalc, by Perwass (Home Page)
  - 3-D visualization and scientific calculation
  - Interprets a script language called CLUScript

Libraries and toolkits

- GluCat, by Leopardi and collaborators (Home Page)
  - C++ library of template classes
  - Non-degenerated signatures
- Gaigen 2, by Fontijine (Home Page)
  - Stand-alone application for generation GA libraries for a target language (*e.g.*, C++, Java)
  - Efficient code is achieved after some profiling and code re-generation

Libraries and toolkits

- Geometric Algebra Template Library, by Fernandes
  - C++ library of template classes
  - MATLAB wrapper
  - Compile-time code optimization
  - One of the most complete libraries
  - Compilation time may be an issue

- Geometrics Ltd. (Home Page)
  - Game company