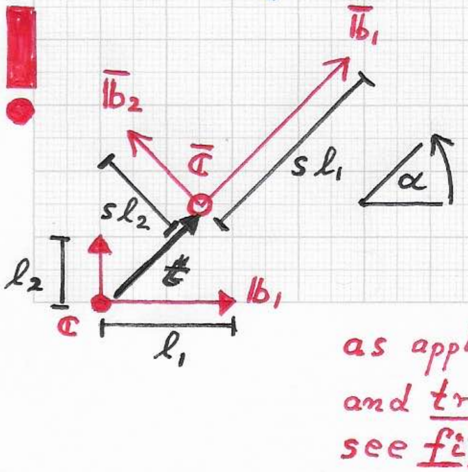
■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks:... In summary, one can understand the geometry of a next-generation box as the result of concatenating scaling, rotation and translation. In other words, it is necessary to design and implement the GA's computational steps in such a way that the geometry of a next-generation box could also be obtained via a concatenation of a limited scaling, a limited rotation (change-in-orientation) and a limited translation. For example, a GA could (re-)combine center point data of two boxes to create the center point of a next-generation box (translation); (re-)combine orthogonal basis vector data of two boxes to create the orthogonal basis vectors of a next-generation box, understood as a change-in-orientation (rotation); and (re-)combine basis vector length (magnitude) data of two boxes to create basis vector lengths of a next-generation box (scaling). Thus, it should be required that the creation of a next-generation box system $\{\bar{\mathbb{c}}, \bar{\mathbb{b}}_1, \bar{\mathbb{b}}_2\}$ can algebraically understood as application of a uniform scaling S, rotation R and translation T applied to "some system" $\{\mathbb{c}, \mathbb{b}_1, \mathbb{b}_2\}$, see figure.



$$S = \begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$T = \begin{bmatrix} 1 & 0 & t_1 \\ 0 & 1 & t_2 \\ 0 & 0 & 1 \end{bmatrix}$$

...

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

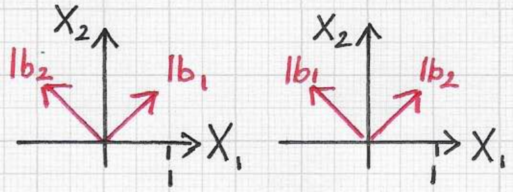
• Laplacian eigenfunctions and neural networks: ... Further, it is necessary to enforce that certain conditions regarding box scaling and box center point location are satisfied. For example, when two boxes are (re-) combined to create a new box of the next generation, the center point of the new box should remain in "relatively close proximity" of the center points of the two parent boxes being (re-) combined; similarly, the lengths of the basis vectors of the new box should be "close" to the lengths of the corresponding basis vector pairs of the two parent boxes. Concerning the change-in-orientation of a new box, it does not seem to be desirable or necessary to restrict it – as long as the basis vectors of a new box are mutually orthogonal and can be viewed as the result of applying a rotation matrix $\mathbf{R}$, with $|\det \mathbf{R}| = 1$, to an original orthogonal basis vector set.

• **Note.** The sign of $\det R$ of a rotation matrix R indicates whether R defines a "proper" ($\det R = 1$) or "improper" ($\det R = -1$) rotation / change-in-orientation. In other words, the columns of R define the orthonormal basis vectors of a "right-handed" or "left-handed" coordinate system (following the terminology used in 3D space).

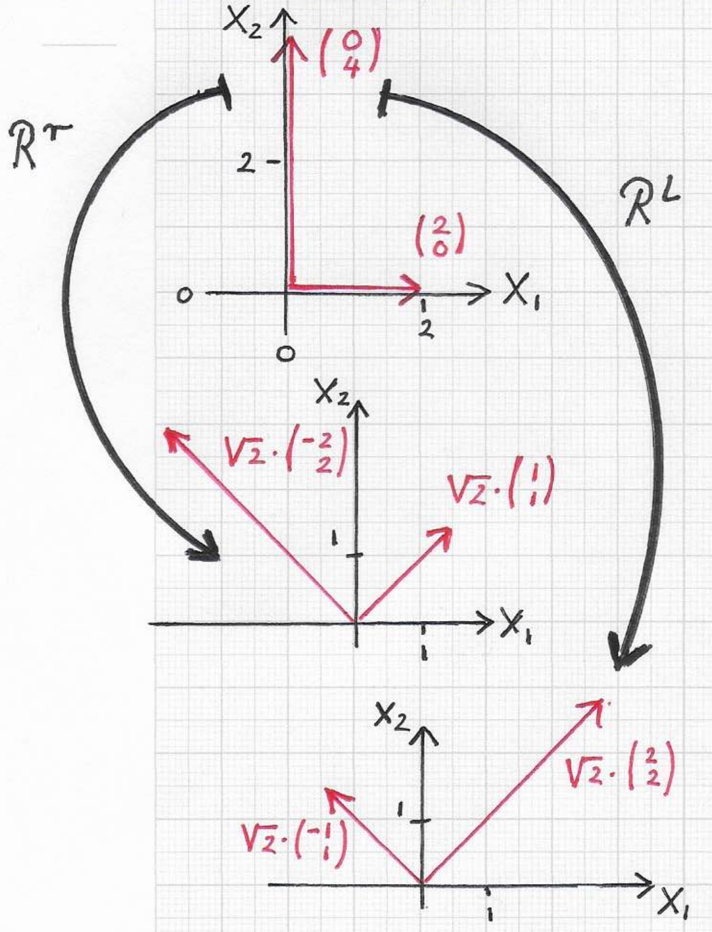
...

▪ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks:...



$$\frac{\sqrt{2}}{2}\cdot\begin{bmatrix}1 & -1\\ 1 & 1\end{bmatrix} = R^{right} = R^{r} \qquad \frac{\sqrt{2}}{2}\cdot\begin{bmatrix}-1 & 1\\ 1 & 1\end{bmatrix} = R^{left} = R^{L}$$



Acceptable orientation changes.

For example, the matrix R^r (R^L) defines a right-handed (left-handed) coordinate system's basis vector set (rotation by 45°), see left figure. Thus, $\det R^r = 1$ and $\det R^L = -1$. When applying R^r (R^L) to the two box basis vectors $(2,0)^T$ and $(0,4)^T$, for example, one obtains:

$$\sqrt{2}/2\cdot\begin{bmatrix}1 & -1\\ 1 & 1\end{bmatrix}\begin{bmatrix}2 & 0\\ 0 & 4\end{bmatrix} = \sqrt{2}\cdot\begin{bmatrix}1 & -2\\ 1 & 2\end{bmatrix},$$

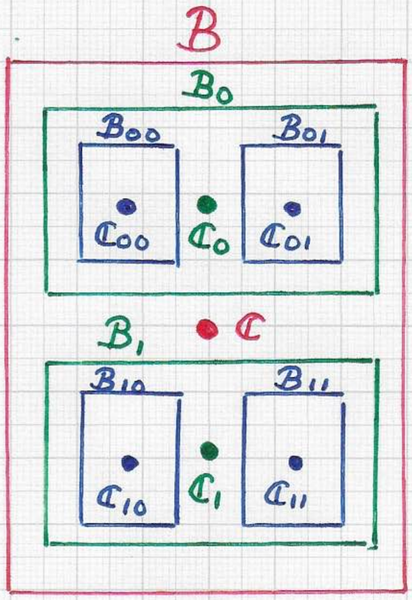
$$\sqrt{2}/2\cdot\begin{bmatrix}-1 & 1\\ 1 & 1\end{bmatrix}\begin{bmatrix}2 & 0\\ 0 & 4\end{bmatrix} = \sqrt{2}\cdot\begin{bmatrix}-1 & 2\\ 1 & 2\end{bmatrix}.$$

The left figure illustrates this scenario. For our purposes, it is irrelevant whether the sign of $\det R$ is positive or negative; as long as the columns of R define a set of unit-length and mutually orthogonal basis vectors of D-dimensional space, R defines an acceptable change-in-orientation.

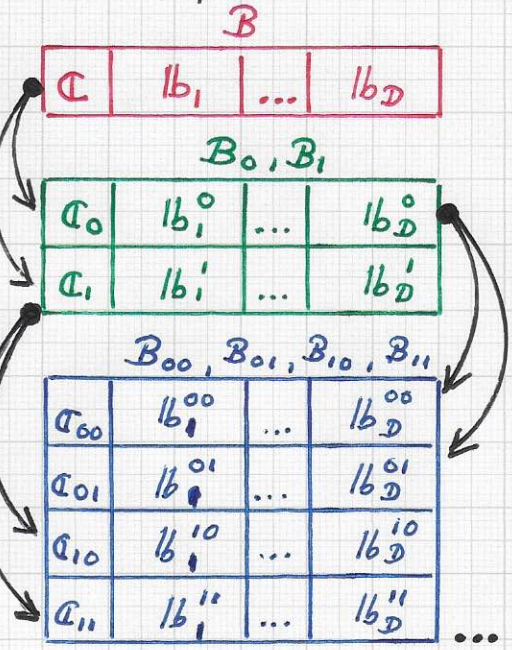
...

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks:...



Abstract tree structure used for all trees in a population.



Box coordinate system data stored in nodes.

A population of box trees is a forest of box trees, and each tree is assumed to be based on the same underlying binary tree node structure, as shown in the left figure. Thus, all trees have the same total number of nodes and tree levels (depth). A genetic (re-) combination operation involving two parent trees uses only the data stored for the same node in the identical tree structure. While the top-left figure only shows the individual box coordinate system center points, the bottom-left figure also includes the D mutually orthogonal box basis vectors that are stored ("represented in encoded format") for all nodes, and for all trees. When (re-) combining the two orthogonal basis vector sets of a node of two trees a new ORTHOGONAL BASIS must result.

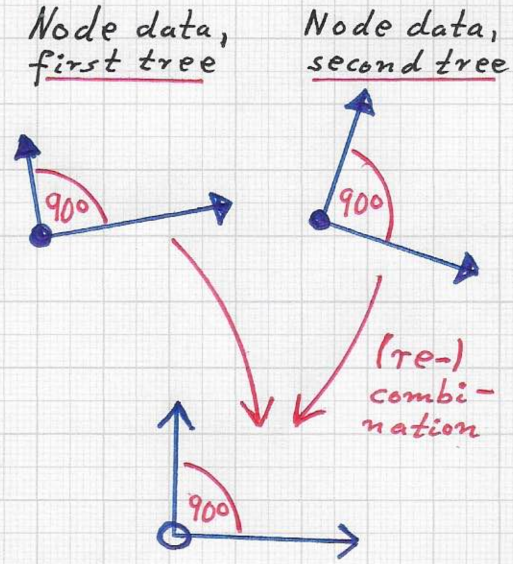
...

Bernd Hamann
Stratovan

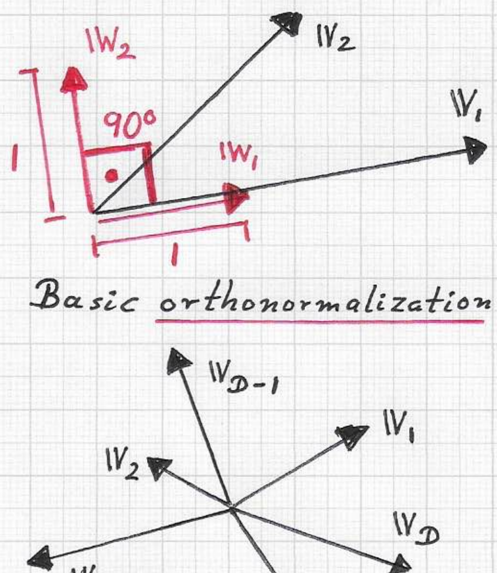
8/17/2023

OBJECT AND MATERIAL EIGENFUNCTIONS - Cont'd.

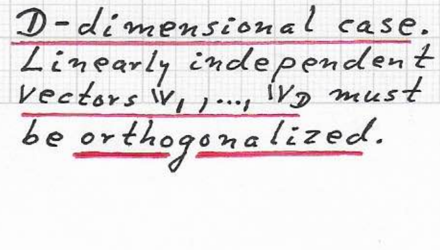
- Laplacian eigenfunctions and neural networks: ...



Orthogonal system results.



Basic orthonormalization.



D-dimensional case. Linearly independent vectors $\mathbb{v}_1, \ldots, \mathbb{v}_D$ must be orthogonalized.

As a (desired) consequence of the stochastic, random behavior of a GA (mutations), the (re-) combination of two sets of mutually orthogonal box basis vectors can only produce a new set of orthogonal basis vectors when proper orthogonality-enforcing steps are carried out. The left figure illustrates this goal. GRAM-SCHMIDT ORTHONORMALIZATION is a standard method used to generate a set of D mutually orthogonal and normalized vectors $\mathbb{w}_1, \ldots, \mathbb{w}_D$ from a set of given linearly independent vectors $\mathbb{v}_1, \ldots, \mathbb{v}_D$, with $\mathbb{v}_i \neq \mathbb{0}$, $i = 1 \ldots D$. The middle-left figure shows orthonormalization for the 2D case:

- $\mathbb{w}_1 := \mathbb{v}_1$; $\mathbb{w}_1 := \mathbb{w}_1 / \|\mathbb{w}_1\|$;
- $\mathbb{w}_2 := \mathbb{v}_2 - \langle \mathbb{v}_2, \mathbb{w}_1 \rangle \mathbb{w}_1$; $\mathbb{w}_2 := \mathbb{w}_2 / \|\mathbb{w}_2\|$.

...