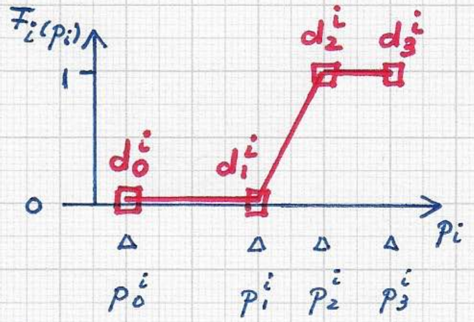
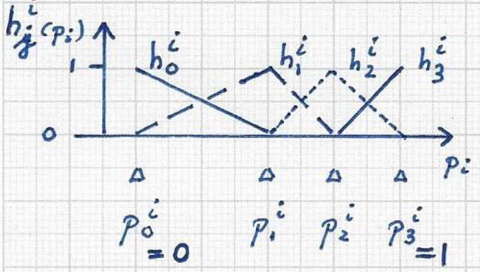
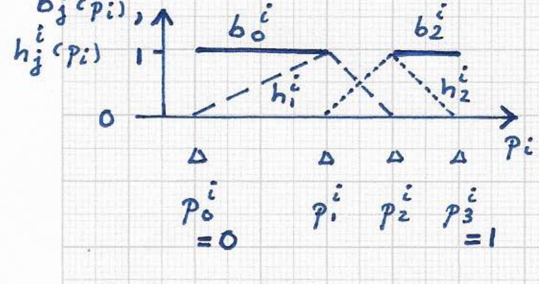
■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• *Laplacian eigenfunctions and neural networks: ...*





Interpretation of $F_i(p_i)$ as a function expressed via linear B-spline basis functions, i.e., hat functions. The values of p_j^i define the knots of the hat functions. The de Boor points d_j^i are the coefficients in the linear combination of $F_i(p_i)$.



Interpretation of $F_i(p_i)$ as a function expressed via constant and linear B-spline basis functions, i.e., box and hat functions.

It is possible to think of the described univariate decider functions $F_i(p_i)$ as functions being linear combinations of B-splines. For example, one could use the linear B-spline basis functions (hat functions h) to represent a function $F_i(p_i)$, see figure (Left). Using the indexing from the figure, one obtains

$$F_i(p_i) = \sum_{j=0}^{3} d_j^i \, h_j^i(p_i) .$$

Since $d_0^i = d_1^i = 0$ and $d_2^i = d_3^i = 1$, one can simplify this definition to

$$F_i(p_i) = h_2^i(p_i) + h_3^i(p_i) .$$

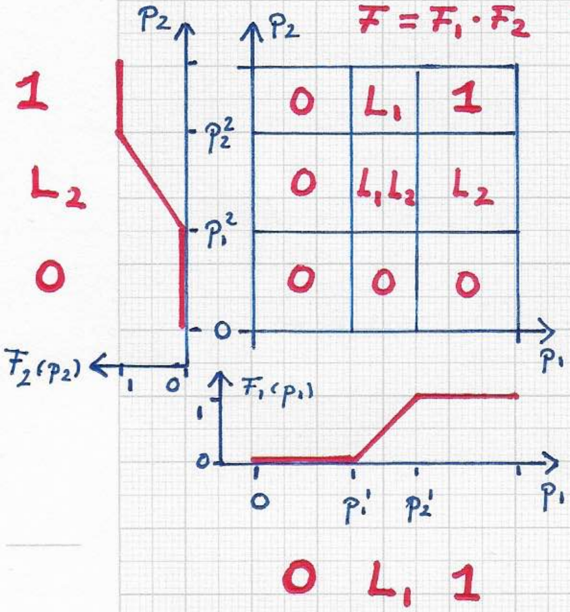
Thus, one can also think of this function, $F_i(p_i)$, as a **"BLaC"** (blending-of-linear-and-constant) function, with two constant and one linear spline segment. The resulting segment-wise definition is

$$F_i(p_i) = \begin{cases} 0 \cdot b_0^i(p_i) = 0 & , p_i \in [p_0^i, p_1^i] \\ 0 \cdot b_0^i(p_i) + 0 \cdot h_1^i(p_i) + 1 \cdot h_2^i(p_i) = h_2^i(p_i), & p_i \in (p_1^i, p_2^i) \\ 1 \cdot b_2^i(p_i) = b_2^i(p_i), & p_i \in [p_2^i, p_3^i] \end{cases}$$

...

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks: ...



Abstract visualization of the types of variation of the tensor product function $F = F_1 \cdot F_2$. The possible five types are:

- **0** constant 0
- **1** constant 1
- **L_1** linear variation from 0 to 1 in p_1-direction
- **L_2** linear variation from 0 to 1 in p_2-direction
- **L_1L_2** bilinear variation from 0 to 1 in a (p_1, p_2) Cartesian product domain square/rectangle

(This more general framework of B-splines is mentioned here, as it would also support the definition of univariate decider functions $F_i(p_i)$ consisting of an arbitrary number of polynomial spline segments of any order/degree. In some applications it might be advantageous to use this possibility.)

The simple knot sequence we consider is $0 = p_0^i < p_1^i < p_2^i < p_3^i = 1$.

Considering the tensor product construction of $F(p_1, p_2) = F_1(p_1) \cdot F_2(p_2)$, the following table summarizes the combinatorially possible "products of the segment types" of F_1 and F_2, i.e., 0, L (linear) and 1:

p_1	0	L_1	1	0	L_1	1	0	L_1	1
p_2	0	0	0	L_2	L_2	L_2	1	1	1
(p_1,p_2)	0	0	0	0	L_1L_2	L_2	0	L_1	1

In the p_1-domain (p_2-domain) an interval $[p_j^1, p_{j+1}^1]$ ($[p_k^2, p_{k+1}^2]$) is of type 0 (zero), L_1 (linear in p_1) or 1 (one) (0 (zero), L_2 (linear in p_2) or 1 (one)). **The types of the tensor product are 0, 1, linear or bilinear.**

...

Bernd Hamann

5/2/2022

Stratovan

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks: ...

The left figure shows a view onto the 3D, volumetric domain of a trivariate decider function obtained as the tensor product

$$F(p_1, p_2, p_3) = F_1(p_1)\, F_2(p_2)\, F_3(p_3).$$

The domain of F, $[0,1]^3$, consists of $3^3 = 27$ cuboids with associated tensor product types implied by the possible products of "base" types **0**, **L** and **1**:

$$0,\ 1,\ L_1,\ L_2,\ L_3,$$
$$L_1L_2,\ L_1L_3,\ L_2L_3,\ L_1L_2L_3.$$

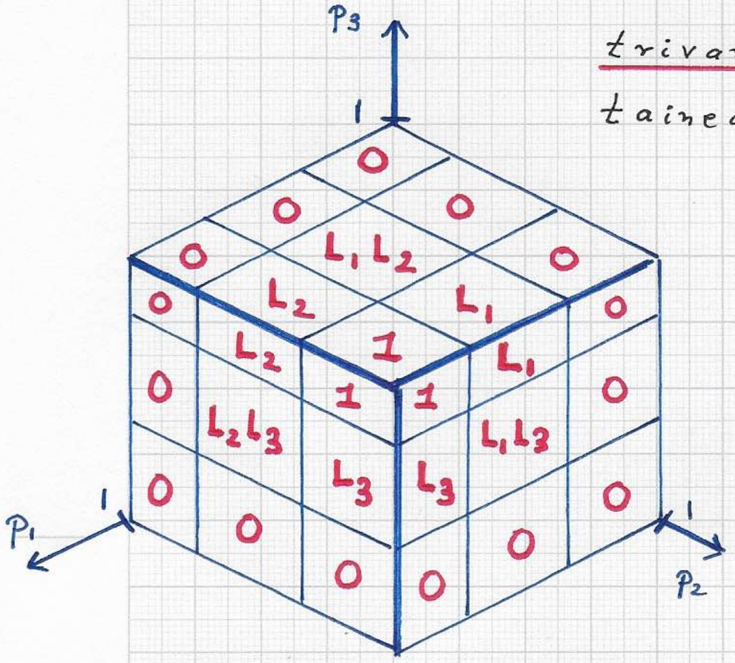


Illustration of domain cuboids of a trivariate decider function $F(p_1, p_2, p_3)$. The underlying individual decider functions $F_j(p_j)$, $j = 1,2,3$, have associated knots $0, p_1^j, p_2^j, 1$. The Cartesian products of the individual domain intervals define the sketched **27** cuboids.

The cuboid with associated trilinear variation of F is not shown, not indicated. It lies in the interior of the overall unit domain cube.

The ONE trilinear cuboid has the type $L_1L_2L_3$.

THE ONLY DATA TO BE STORED IS THE SET OF INTERIOR KNOTS, I.E.,

$$\left\{ p_1^j,\ p_2^j \right\}_{j=1}^{H}.$$

• **Note.** It is important to keep the fact in mind that a bilinear or trilinear (or, more generally, a multilinear) function is NOT a linear function. For example, the function L_1L_2 has a p_1p_2 term and the function $L_1L_2L_3$ has a $p_1p_2p_3$ term as part of the definition.

The values of the knots p_1^j, p_2^j, $j = 1 \ldots H$, completely define $F(p_1, \ldots, p_H)$. !

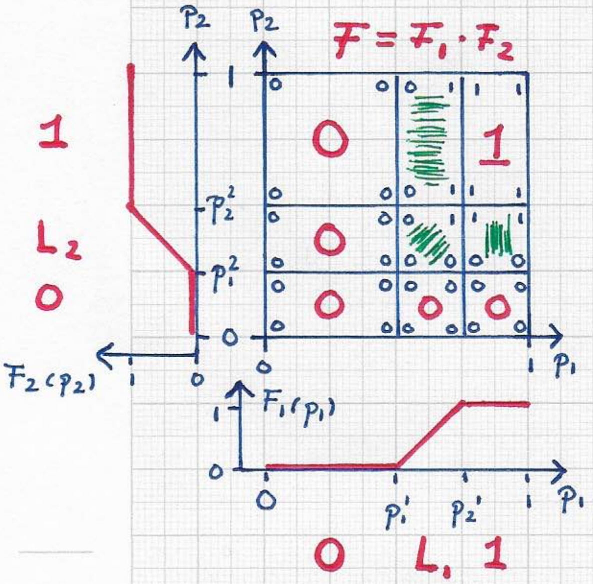
...

Bernd Hamann

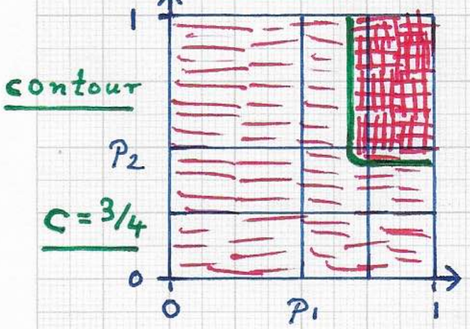
Stratovan

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks: ...



Rectangles defining the unit square subdivision in the bivariate case. Function values at rectangle corners are either 0 or 1. The bivariate decider function $F(p_1, p_2)$ is monotonically increasing in p_1- and p_2-direction. Only the shaded rectangles (green) can contain the contour $F = c$, with $0 < c < 1$.



The isoline for $F = 3/4$ separates the unit square into 2 regions. Vertex values: top figure.

In summary, when considering all H scales – defined by $p_1, \ldots, p_H$ and the associated univariate decider functions $F_i(p_i)$, $i = 1 \ldots H$ – the total number of hyper-cuboids in H-dimensional space is 3^H; their union is the unit domain hyper-cube $[0, 1]^H$. Further, the hyper-cuboids in the unit domain have the combinatorially possible tensor product types

- 0, 1 constant-0, constant-1
- $L_1, \ldots, L_H$ linear ("1-linear")
- $L_1 L_2, \ldots, L_{H-1} L_H$ bilinear ("2-linear")
- $L_1 L_2 L_3, \ldots, L_{H-2} L_{H-1} L_H$ trilinear ("3-linear")
- $L_1 \cdot \ldots \cdot L_H$ "H-linear"

Thus, the total number of multi-linear possibilities, including the two "0-linear" constant cases, is $1 + 1 + \binom{H}{1} + \binom{H}{2} + \ldots + \binom{H}{H} = 2^H + 1$.

FOR CLASSIFICATION PURPOSES A HIGHLY EFFICIENT EVALUATION OF $F(p_1, \ldots, p_H)$ IS CRUCIAL. ...

!

■ OBJECT AND MATERIAL EIGENFUNCTIONS – Cont'd.

• Laplacian eigenfunctions and neural networks: ...

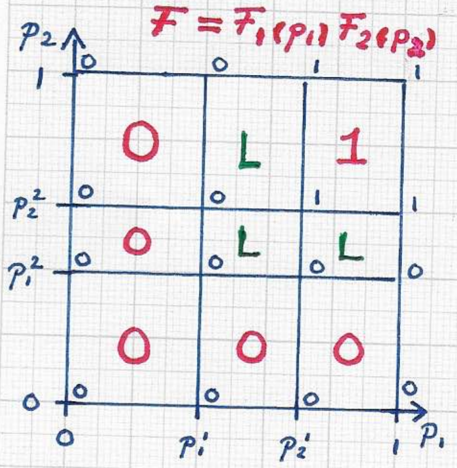


Illustration of the three basic cuboid types that arise via a tensor product function $F(p_1, p_2)$ in its unit hyper-cube domain.

The three possible types are:

- **O** constant zero
- **1** constant one
- **L** all cuboids with linear variation in at least one of the p_i-directions, i.e., 1 – linear, 2 – linear, ..., H – linear

The "important" cuboids are the **L**-cuboids since they contain the interface/boundary between **O** and **1**.

This combinatorial analysis of the types and numbers of possible "tensor product cuboids" is interesting and relevant for computational complexity considerations. If a tuple $(p_1, \dots, p_H)$ satisfies the condition

$$p_1 \le p_1^1 \vee p_2 \le p_1^2 \vee \dots \vee p_H \le p_1^H,$$

then the tuple yields the value $\boxed{F = 0}$. If a tuple $(p_1, \dots, p_H)$ satisfies

$$p_1 \ge p_2^1 \wedge p_2 \ge p_2^2 \wedge \dots \wedge p_H \ge p_2^H,$$

then the resulting decider function value will be $\boxed{F = 1}$. These are the two simple cases define the no-match $(F = 0)$ and match $(F = 1)$ scenarios. When a tuple component p_i satisfies $p_1^i < p_i < p_2^i$, the cuboid containing $(p_1, \dots, p_i, \dots, p_H)$ is of type **L**, see left figure; **F is multi-linear**, and is of sub-type **1-Linear**, ..., or **H-Linear**.

H	no. O	no. L	no. 1
1	1	1	1
2	5	3	1
3	19	7	1
H	$3^H - 2^H$	$2^H - 1$	1

Numbers of the 3 possible cuboid types.

The table (left) shows that the number of cuboids of type **L** is $2^H - 1$. These cuboids have vertex values **O** and **1** and require efficient multi-linear evaluation. ...