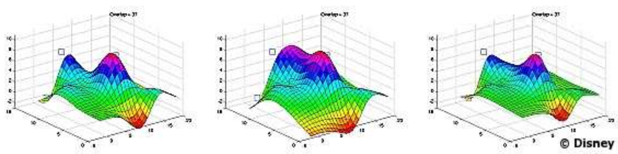
Evaluation of the Radial Basis Function Space

Gene S. Lee*
Walt Disney Animation Studios



(a) ϕ = GS (b) ϕ = MQ (c) ϕ = IMQ

Figure 1: *Height-fields for three kernels from Viz. 9. The x-, y-, and z-axes represent the $\Re^2$ driver and RBF value, respectively.*

1 Introduction

Radial basis functions (RBFs) are important in computer graphics, with widespread use in reconstruction, animation, etc. In *pose space deformation* (PSD) [Lewis et al. 2000], RBFs are used to perform shape interpolation for pose-driven deformations. The behavior of an RBF depends on its particular definition. For artists, the relationship between definition and behavior is not intuitive. We suggest a method for artists to better understand RBF behavior through visualization and to evaluate RBF functions according to the requirements of a production environment.

Key Parameters In general, an RBF is a function of the form

$$s(x) = p(x) + \sum_{i=1}^{N} \lambda_i \phi(\|x - x_i\|) \quad (1)$$

where p is a polynomial of low-degree, ϕ is a real-valued kernel function (some having size σ), x_i are the locations of the kernel in a vector space and λ_i are weights [Cheney and Light 1999]. In PSD, the x_i represent specific poses in pose space.

The task of a technical director (TD) is to define, p, ϕ, x_i and optionally σ, to achieve an aesthetically pleasing result. Unfortunately, the non-intuitive nature of the RBF leave TDs ill-prepared to efficiently create an RBF suited to each task.

2 Visualizing Parameter Change

One solution is to visualize the RBF over a sampling of the RBF space. These visualizations allow TDs to qualitatively assess the behavior of the RBF and better anticipate its impact on PSD. For each component of the RBF, we selected a discrete set of choices and then visualized combinations of those parameters.

- The Gaussian (G), multiquadric (MQ), inverse multiquadric (IMQ), and linear (LN) kernel functions were examined for ϕ, with values for σ causing kernels to overlap its neighbors, or a universal, uniform value for σ causing gaps or overlap, depending on the distribution of x_i; no σ exist for LN.
- The zero and constant functions were included for p.
- The distribution of x_i divided into three coarse categories: regular, irregular, sparse, reflecting how uniformly the poses are distributed in pose space.
- x_i were set in either $\Re^1$ or $\Re^2$. In practice, TDs rarely used more than three dimensions. (And visualizing a function in $\Re^3$ or greater is problematic.)

Tables 1 and 2 show the evaluation results; a "✓" identifies interpolation considered reasonable. Figure 1 presents three sample visualizations from Viz 9. The height-field interpolates the x_i values.

*gene.s.lee@disneyanimation.com

Viz	x_i	σ	p	ϕ=G	MQ	IMQ	LN
1	regular	uniform	0			✓	
2	irregular	uniform	0		✓		
3	irregular	overlap	0		✓	✓	N/A
4	sparse	overlap	0		✓	✓	
5	regular	uniform	1			✓	
6	irregular	uniform	1		✓		
7	irregular	overlap	1		✓	✓	N/A
8	sparse	overlap	1		✓	✓	

Table 1: *Visualizations with $x_i \in \Re^1$*

Viz	x_i	σ	p	ϕ=G	MQ	IMQ	LN
9	regular	uniform	0		✓	✓	
10	irregular	uniform	0		✓		✓
11	irregular	overlap	0	✓	✓	✓	N/A
12	sparse	overlap	0	✓	✓	✓	✓
13	regular	uniform	1				✓
14	irregular	uniform	1		✓		✓
15	irregular	overlap	1	✓	✓		N/A
16	sparse	overlap	1		✓		✓

Table 2: *Visualizations with $x_i \in \Re^2$*

3 Evaluation and Findings

Visualizations of the configurations in the tables were prepared for TDs and qualitatively evaluated according to the following criteria:

- Smooth: The function has no sharp peaks or valleys.
- Predictable: It is straightforward to anticipate the results of small modifications, such as adding a pose.
- Clamp-able: The function optionally supports non-zero values beyond the kernel core. This allows a deformation defined at a particular location to remain in effect beyond where it was explicitly defined.

From these visualizations, TDs garnered a greater understanding of the relationship between RBF definitions and the resultant interpolation behavior. This suggested targeted experiments, which led to the following practical observations:

- Tuning σ values, both manually and algorithmically, proved difficult and generally produced poor results.
- Gaussian and inverse multiquadric ϕ's produced undesirable results. These kernels introduce undesirable fluctuations, perceived as exaggerated hills and valleys in the visualizations.
- Multiquadric ϕ is useful occasionally and relatively insensitive to the changes in σ.
- Linear ϕ produces reasonable results, including minor persistence, and is also insensitive to σ size. However, it can produce "popping" in the PSD, which correspond to sharp peaks in the visualization.
- Linear ϕ works best with constant p. Popping disappears and the interpolation is smooth. This combination was favored most in practice.

References

CHENEY, E., AND LIGHT, W. 1999. *A Course in Approximation Theory*. Brooks Cole.

LEWIS, J. P., CORDNER, M., AND FONG, N. 2000. Pose space deformation: A unified approach to shape interpolation and skeleton-driven deformation. In *Proceedings of ACM SIGGRAPH 2000*, ACM Press, ACM, 165–172.