Latent Nonuniform Splines for Animation Approximation

Tomohiko Mukai*
Square Enix Co., Ltd.

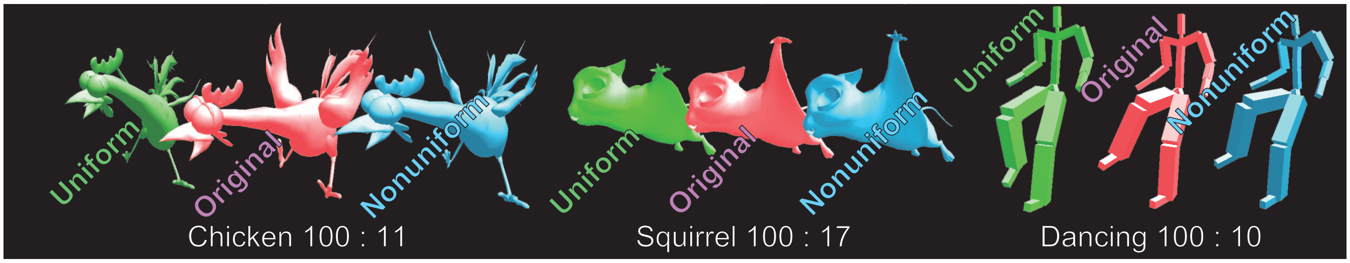

Figure 1: *Approximating animation sequences by uniform and nonuniform B-spline. (a) Chicken animation, (b) squirrel animation and (c) motion capture data of dancing are represented by 11%, 17% and 10% of original data, respectively. Our method using nonuniform B-spline provides more accurate approximation than using uniform B-spline.*

Abstract

This paper presents a new method to approximate animation sequences through a nonlinear analysis of the spatiotemporal data. The main idea is to find a spline curve which best approximates a multivariate animation sequence in a reduced subspace. Our method first eliminates data redundancy among multiple animation channels using principal component analysis (PCA). The reduced sequence of latent variables is then approximated by a nonuniform spline with free knots. To solve the highly-nonlinear multimodal problem of the knot optimization, we introduce a stochastic algorithm called covariance matrix adaptation evolution strategy (CMA-ES). Our method optimizes the control points and the free knots using least-square method and CMA-ES, which guarantees the best approximation for arbitrary animation sequences such as mesh animations and motion capture data. Moreover, our method is applicable to practical production pipeline because both PCA- and CMA-based algorithms are computationally stable, efficient, and quasi manual parameter-free. We demonstrate the capability of the proposed method through comparative experiments with a common approximation technique.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation

Keywords: animation approximation, nonuniform spline, covariance matrix adaptation evolution strategy

1 Introduction

Animation approximation is a common technique to compactly represent an animation sequence such as mesh animations and skeletal animations. By approximating an animation curve using a combination of basis functions, the original animation is described with a fewer number of function variables. The animation approximation is useful not only to compress the data, but also to reduce the degrees of freedom of animation control: the free variables of approximation function can be used as control handles to edit the animation curve. For example, designers prefer to edit a motion capture data by manipulating a sparse set of control handles rather than by directly editing the dense sequence of raw data.

Spline approximation is a general method to represent an animation sequence with a small number of variables, which utilizes a temporal coherence of the animation data. For example, B-spline approximation is widely used to reduce the number of free variables in optimization-based motion editing techniques [Gleicher 1998]. Although most of existing methods use uniform spline because of its efficiency and simple formulation, it often causes an increase of redundant data for complex sequences. For example, when an original animation contains a short subsequence which shows high-frequency oscillations, many control points have to be used for reproducing the high-frequency behavior even if the rest subsequences are near-stationary. Ideally, the knot placement of the spline should be optimized so that more knots are arranged around the high-frequency subsequences than the low-frequency ones. However, such a knot optimization is a highly-nonlinear multimodal problem, which means that the objective function is undifferentiable and the problem typically has multiple local and global solutions. Since this kind of problem cannot be solved using efficient numerical algorithm such as Newton's method, evolutionary computation techniques have gained considerable attention in computer-aided design community.

Dimension reduction technique is also used to eliminate data redundancy of an animation sequence by leveraging a spatial correlation among multiple animation channels [Alexa and Müller 2000]. By applying principal component analysis (PCA), the animation data is approximated by a sequence of the smaller number of latent variables. Existing methods therefore integrate the spline approximation and dimension reduction to further reduce the data redundancy [Karni and Gotsman 2004; Arikan 2006; Liu and McMillan 2006].

This paper presents a new method of animation approximation which leverages both spatial and temporal correlation of animation sequence. While we follow an existing approach of animation approximation using dimension reduction and spline approximation, we tackle a challenging problem of automatic optimization of free knots for nonuniform spline. Our method first reduces the dimen-

*e-mail:tmki@acm.org

sionality of an animation sequence using PCA, and the reduced sequence of latent variables is then approximated by a nonuniform spline whose control points and free knots are optimized using a least square method and a stochastic optimization technique called covariance matrix adaptation evolution strategy (CMA-ES). The nonlinear optimization of free knots improves the approximation accuracy whereas existing techniques use a uniform spline which often fails to capture dynamic movement in the animation.

Contributions We here summarize the major contributions of our technique. To the best of our knowledge, we are the first to introduce a stochastic optimization technique, CMA-ES, for optimizing free knots of nonuniform spline. The most important benefit of CMA-ES is that it uses fewer manual parameters than other optimization techniques. Thanks to the quasi parameter-free algorithm, our system requires that a user specify only a few control parameters which can be physically interpreted. In contrast, previous knot optimization techniques require fine tuning of many manual parameters which are unintuitive for ordinary users. Moreover, the computational cost of our method is practical because CMA-ES is an embarrassingly parallel algorithm. Note that although we only refer to B-spline in the following sections, our CMA-based technique can be applied for any types of spline which has free knots.

2 Related work

Uniform B-spline is most widely used in commercial products because of its simplicity and computational efficiency. The uniform B-spline, however, cannot produce high-frequency vibration of animation curve unless using many control points. To overcome this problem, several algorithms to optimize nonuniform B-splines have been proposed in the field of graphics and computer-aided design. Most of recent methods employ an evolutional computation method such as genetic algorithm [Yoshimoto et al. 2003] and artificial immune system [Ülker and Arslan 2009]. Gálvez and Iglesias [2011] also report that a particle swarm optimization (PSO) is computationally stable, efficient, and works well for arbitrary shapes of curve including discontinuous changes. However, these previous techniques often suffer from manual tuning of unintuitive parameters. For example, PSO-based method requires that three types of manual parameters be appropriately adjusted based on domain-specific knowledge. In contrast, CMA-ES is well-known to be a quasi parameter-free algorithm: even though it has many manual parameters, there is no need to change the most parameters from optimal settings which are suggested in a previous study [Hansen and Kern 2004]. In fact, we confirmed that the suggested settings works for all data set used in our experiments. Moreover, CMA-ES is computationally efficient equivalent to PSO [Hansen 2006].

Our method is also related to [Cashman and Hormann 2012] which approximates a mesh animation sequence by a nonuniform B-spline in a PCA-subspace. However, this method only guarantees a local optimality of the free knots due to its heuristic algorithm. Moreover, it uses time remapping algorithm to improve the convergence of the heuristic optimization, which requires redundant data and extra computation to remap the approximated sequence to the original time domain. In contrast, our method searches a globally optimum approximation without using any time remapping algorithm.

3 Algorithm

3.1 Dimension reduction

Given an animation sequence $\mathbf{y}_n \in \Re^M | n \in \{1, \cdots, N\}$ where M and N respectively denote the number of animation channels and the number of time frames, a low-dimensional animation curve $\mathbf{x}_n \in \Re^L | n \in \{1, \cdots, N\}$ is obtained using truncated PCA [Alexa and Müller 2000]. The reduced number of principal components L is determined according to some criterion such as tolerance of approximation error, lower limit of eigenvalue, and cumulative contribution ratio of eigenvalues. We think the first one is most intuitive for ordinal users because it can be physically interpreted. If the animation $\mathbf{y}$ has a channel of 3D rotations, we use principal geodesics analysis (PGA) [Tournier et al. 2009] instead of the linear PCA.

3.2 CMA-based spline approximation

The reduced sequence $\mathbf{x}$ is approximated by a nonuniform B-spline $\tilde{\mathbf{x}}(\tau), 0 \le \tau \le 1$ which is expressed as follows:

$$\tilde{\mathbf{x}}(\tau) = \sum_{k=0}^{K} B_{k,D}(\tau)\,\mathbf{c}_k, \quad (1)$$

$$B_{k,d}(\tau) = \frac{\tau - t_k}{t_{k+d} - t_k} B_{k,d-1}(\tau) + \frac{t_{k+d+1} - \tau}{t_{k+d+1} - t_{k+1}} B_{k+1,d-1}(\tau), \quad (2)$$

$$B_{k,0}(\tau) = \begin{cases} 1 & t_k \le \tau < t_{k+1} \\ 0 & \text{otherwise} \end{cases}, \quad (3)$$

$$\forall k : t_k \le t_{k+1}$$

$$t_0 = 0, t_{K+D} = 1$$

where $B_{k,d}(\tau)$ is B-spline basis function, $\mathbf{c}_k \in \Re^L | k \in \{0, \cdots, K\}$ are control points, and $\mathbf{t} = \{t_0, \cdots, t_{K+D}\}$ is a knot sequence. The degree of spline D and the number of control points K are manually specified. The first and last knots are commonly repeated with multiplicity equal to the degree of spline D as $t_0 = \cdots = t_D = 0$ and $t_K = \cdots = t_{K+D} = 1$ without loss of generality, and the rest $t_{D+1} \cdots t_{K-1}$ are called internal knots. The goal of the optimization is to find the best placement of the $K - D - 1$ internal knots and the associated control points so as to minimize approximation error. We use root mean squared (RMS) error as the objective function:

$$E_{RMS} = \sqrt{\frac{1}{N} \sum_{n=1}^{N} \|\mathbf{x}_n - \tilde{\mathbf{x}}(\tau_n)\|^2}. \quad (4)$$

$$\tau_n = (n-1)/(N-1)$$

Since E_{RMS} is quadratic with respect to control point $\mathbf{c}$, that optimal $\mathbf{c}$ can be uniquely determined by least square method if knot sequence $\mathbf{t}$ is fixed. However, because E_{RMS} is non-linear and undifferentiable with respect to $\mathbf{t}$, we cannot use a standard minimization algorithm such as Newton's methods.

We use CMA-ES and least square method for optimizing free knots and control points, respectively. CMA-ES is a sampling-based global optimization technique which generally works for nonlinear, high dimensional problem and the objective function not need to be differentiable with respect to free variables, both of which are desirable property for our problem. CMA-ES iterates sampling, evaluation and updating processes until converging to an optimal solution. During the iteration, the mean and variance of the sampling distribution are adaptively updated according to evaluation result at all sampling points using maximum-likelihood estimation.

For our problem, only internal knots are optimized using CMA-ES since the optimal control points $\mathbf{c}$ are uniquely determined when the knot sequence $\mathbf{t}$ is determined. As a result, an iteration of CMA-based spline approximation is composed of four steps: sampling of internal knots, least square estimation of control points, evaluating the approximation error, and updating mean and variance of the sampling distribution. These processes are iterated until either the approximation error or the variance gets smaller than given threshold. Each step of the iteration is briefly explained as follows:

(1) Constrained sampling of free knots The candidate knot sequence $\tilde{\mathbf{t}}_s, s \in \{1, \cdots, S\}$ are sampled according to the normal distribution with the mean and variance, where S is the number of sampling points. To compose a monotonically increasing knot sequence, a positive increment between two adjacent knot $\triangle\tilde{t}_d = \tilde{t}_d - \tilde{t}_{d-1}$ is sampled instead of directly sampling $\tilde{t}_d$. The non-negative condition $\triangle\tilde{t}_d \geq 0$ is enforced by simple rounding out as $\max(\triangle\tilde{t}_d, 0)$. The sampled increments are uniformly scaled by a factor α so as to ensure $t_{K+D} = \sum_{d=1}^{K+D} \alpha\triangle\tilde{t}_d = 1$.

(2) Least-square estimation of control points Optimal control points for each candidate $\tilde{\mathbf{t}}_s$ are uniquely determined using least-square method. Let $\tilde{\mathbf{C}}_s = [\tilde{\mathbf{c}}_{s,0} \cdots \tilde{\mathbf{c}}_{s,K}]^T$ denote a set of control points, the optimal solution is determined by solving a linear system $\mathbf{B}|_{\tilde{\mathbf{t}}_s}\tilde{\mathbf{C}}_s = \mathbf{X}$, where $\mathbf{B}_{ij}|_{\tilde{\mathbf{t}}_s} = B_{j,D}|_{\tilde{\mathbf{t}}_s}(i/(N-1))$ and n-th row of $\mathbf{X}$ is $\mathbf{x}_n$. We use either Moore-Penrose pseudoinverse or QR decomposition to calculate a minimum-error solution of this over-constrained linear problem.

(3) Evaluation Each candidate $\tilde{\mathbf{t}}_s$ and associated control points $\tilde{\mathbf{C}}_s$ is evaluated using the RMS error expressed in Equation 4.

(4) Updating sampling distribution The mean and variance of the sampling distribution are adaptively updated according to the evaluation results using maximum likelihood algorithm. Please refer to [Hansen 2006; Hansen and Kern 2004] for the details.

This optimization procedure can be parallelized because each candidate $\tilde{\mathbf{t}}_s$ is independently evaluated in step (1), (2) and (3). The optimization algorithm is currently implemented on multicore CPU using multithreading technique, and we think that a GPU implementation could also be possible.

3.3 Quasi parameter-free optimization

The manual parameters of our system are the number of principal components L, the number of control points K, the degree of spline D, and the number of sampling points S. The former two parameters are determined according to the tolerance of approximation error or the data capacity. The degree of spline D is generally set to 2 or 3 because C^1 or C^2 continuity of the curve is sufficient to produce a smooth animation. The last parameter, S, should be manually specified according to the target animation. We experimentally confirmed that $100 \leq S \leq 200$ was enough to find approximately global optimum for short animation sequences of less than 500 frames. Although we can use smaller value of S for an efficient computation, as many sampling points as possible should be used for the stable convergence.

4 Experimental results

The approximation capability of our method is evaluated using several functional curves, mesh animation sequences, and motion capture sequences. The approximation accuracy is evaluated using a distortion measure E_D [Karni and Gotsman 2004] expressed as

$$E_D = 100\sqrt{\frac{\sum_n \|\mathbf{y}_n - \tilde{\mathbf{y}}(\tau_n)\|^2}{\sum_n \|\mathbf{y}_n - \bar{\mathbf{y}}\|^2}}, \tag{5}$$

where $\tilde{\mathbf{y}}(\tau_n)$ and $\bar{\mathbf{y}}$ denote the approximated sequence and the mean of original $\mathbf{y}_n$, respectively. All timing were measured on a Dual

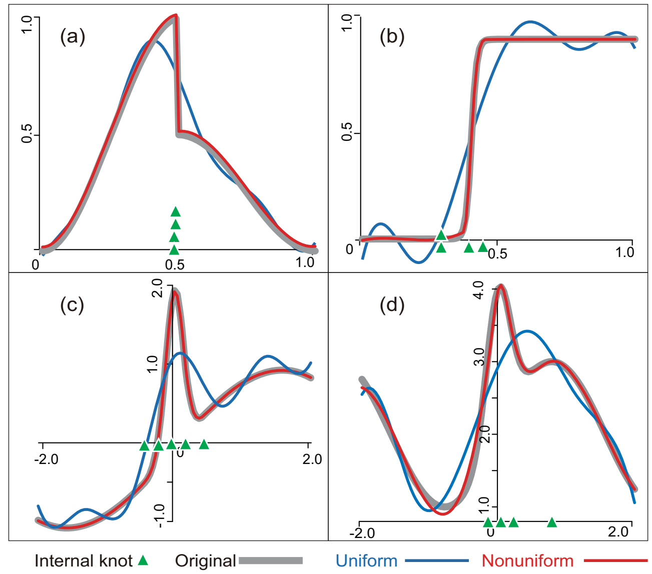

Figure 2: *Approximation of test functions by uniform B-splines and nonuniform ones. Nonuniform B-spline with knot optimization precisely approximates the original curve since the knots are adaptively arranged around where a large change occurs.*

Table 1: *Distortion measures for mesh animations and motion capture sequences. "PCs" and "IKs" stand for "principal components" and "internal knots", respectively.*

	Chicken 9090 DOF 400 frames		Squirrel 11505 DOF 104 frames		Punching 72 DOF 538 frames		Dancing 72 DOF 203 frames	
#PCs	86		35		35		34	
#IKs	30	50	20	40	50	75	20	40
Uniform	24.1	14.9	19.5	10.9	18.8	11.2	22.8	3.7
Nonuniform	11.8	7.3	10.1	1.7	13.4	7.7	12.8	2.6

Xeon W5590 CPU (16 logical processors) at 3.33 GHz with 48 GB RAM.

4.1 1D functional curve

We first demonstrate the approximation capability of our CMA-based algorithm for 1D curve without using dimension reduction. Figure 2 shows four test functions which are used in [Gálvez and Iglesias 2011]. Each curve was discretized by uniform sampling with $N = 101$ frames. We used cubic B-spline ($D = 3$), and $S = 100$ sampling for the optimization. The test functions (a), (b) and (d) were respectively approximated using 4 internal knots where 12 iterations were executed in about 19 milliseconds. For the function (c), approximation with 5 internal knots required 17 iterations executed in about 26 milliseconds. Figure 2 demonstrates the improvement of accuracy by optimizing knot placements. We experimentally confirmed that there was no significant difference in the accuracy with the PSO-based method [Gálvez and Iglesias 2011] because both methods found the same global optimum.

4.2 Mesh animation and motion capture sequence

We compared the approximation accuracy of our nonuniform B-spline and uniform one using mesh animation sequences of chicken and squirrel, and motion capture sequences of dancing and martial arts as shown in the supplemental video. Table 1 summarizes the distortion measures under several experimental conditions. This result clearly shows the nonuniform spline provides better approximation than uniform one. Note that we used so small number of internal knots that causes significant artifacts, in order to visually emphasize the difference of accuracy between two methods.

For chicken sequence, large error occurred in later subsequences when the chicken character gets a surprise and beats its wings. The uniform B-spline caused larger low-frequency error according to the decrease of the number of knots. In contrast, our nonuniform B-spline reduced the noticeable error by arranging many knots around subsequences of dynamic movements. For squirrel animation, our method managed to capture heady arm swings whereas uniform B-spline entirely failed to do it. For two motion capture sequences, the improvement of accuracy was relatively slight as visually significant improvement cannot be observed in the rendered animations. This is because these motion sequences show smooth and constant movement without sudden change, which can be sufficiently approximated using uniform spline. This result indicates that our method is effective especially for dynamic movements which often appear in stylized and exaggerated animations.

5 Discussion

We have developed an animation approximation technique which integrates nonuniform spline approximation and dimension reduction technique. Our main contribution is a stable, efficient and quasi manual parameter-free algorithm of knot optimization which utilizes the capability of CMA-ES. The nonuniform spline is especially effective for approximating complex animation sequence of mixture of stable and dynamic movements.

We here compare our algorithm with some animation compression and approximation techniques. A soft-body animation compression technique which integrates PCA and linear predictive coding (LPC) shows a compression performance superior to our spline-based method [Karni and Gotsman 2004]. If an animation curve can be expressed by a k-th order dynamic equation, LPC accurately approximates the curve with only $k+1$ coefficients. In contrast, our method requires more control points if the curve has many local extrema and saddle points. However, the LPC-based method does not allow a random access to the sequential data due to the sequential decoding algorithm. The spline-based method provides a fast random access in continuous time domain, which is often required for interactive applications such as games. On the other hand, some algorithms that are tailored to compress humanoid animation data [Arikan 2006; Liu and McMillan 2006] also show higher performance for motion capture data. We believe that our method can achieve equivalent performance by introducing domain knowledge about the skeletal structure and adequate post-processing technique like inverse kinematics.

One possible application of our method is a bilinear spatiotemporal basis model [Akhter et al. 2012] which is a generalized model for representing a spatiotemporal data based on dimension reduction and timeseries analysis. Although the previous study reports that orthonormal cosine basis is generally the best choice for a wide range of animation data, we expect the nonuniform spline provides better approximation for complex animation sequence. The more detailed investigation is our future work.

Our future work also includes a development of an automated algorithm to determine the optimal number of control points K. Previous studies introduce information criteria, such as Akaike information criteria and Bayesian information criteria, to determine the best setting that optimizes a tradeoff between approximation accuracy and data size. However, the computational time increases according to the complexity of the problem because this approach uses a brute force algorithm. Moreover, we could allocate more control points to lower eigenmodes than higher ones for efficiently reducing the data size while preserving the perceivable accuracy. We will explore an effective algorithm to determine the optimal number of control points for each eigenmode under the limited data capacity.

Acknowledgement

The chicken character was created by Andrew Glassner, Tom McClure, Scott Benza, and Mark Van Langeveld. This short sequence of connectivity and vertex position data is distributed solely for the purpose of comparison of geometry compression techniques. The squirrel animation is distributed from the Blender Foundation under a Creative Commons Attribution 3.0 License. The motion capture data used in this project was obtained from mocap.cs.cmu.edu.

References

AKHTER, I., SIMON, T., KHAN, S., MATTHEWS, I., AND SHEIKH, Y. 2012. Bilinear spatiotemporal basis models. *ACM Transactions on Graphics 32*, 2, 17.

ALEXA, M., AND MÜLLER, W. 2000. Representing animations by principal components. *Computer Grahics Forum 19*, 3, 411–418.

ARIKAN, O. 2006. Compression of motion capture databases. *ACM Transactions on Graphics 25*, 3, 890–897.

CASHMAN, T., AND HORMANN, K. 2012. A continuous, editable representation for deforming mesh sequences with separate signals for time, pose and shape. *Compuer Graphics Forum 31*, 2, 735–744.

GÁLVEZ, A., AND IGLESIAS, A. 2011. Efficient particle swarm optimization approach for data fitting with free knot b-splines. *Computer-Aided Design 43*, 12, 1683–1692.

GLEICHER, M. 1998. Retargetting motion to new characters. In *Proc. of SIGGRAPH 98*, 33–42.

HANSEN, N., AND KERN, S. 2004. Evaluating the CMA evolution strategy on multimodal test functions. In *Proc. of Eighth International Conference on Parallel Problem Solving from Nature*, 282—291.

HANSEN, N. 2006. The CMA evolution strategy: A comparing review. In *Towards a New Evolutionary Computation. Advances in Estimation of Distribution Algorithms*, J. Lozano, P. Larranaga, I. Inza, and E. Bengoetxea, Eds. Springer, 75–102.

KARNI, Z., AND GOTSMAN, C. 2004. Compression of soft-body animation sequences. *Computers & Graphics 28*, 1, 25–34.

LIU, G., AND MCMILLAN, L. 2006. Segment-based human motion compression. In *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer Animation 2006*, 127–135.

TOURNIER, M., WU, X., COURTY, N., ARNAUD, E., AND REVERET, L. 2009. Motion compression using principal geodesics analysis. *Computer Graphics Forum 28*, 2, 337–346.

ÜLKER, E., AND ARSLAN, A. 2009. Automatic knot adjustment using an artificial immune system for b-spline curve approximation. *Information Sciences 179*, 10, 1483–1494.

YOSHIMOTO, F., HARADA, T., AND YOSHIMOTO, Y. 2003. Data fitting with a spline using a real-coded genetic algorithm. *Computer-Aided Design 35*, 8, 751–760.