Motion Reparametrization

Fernando Wagner da Silva

Luiz Velho

Jonas Gomes

IMPA - Instituto de Matemática Pura e Aplicada

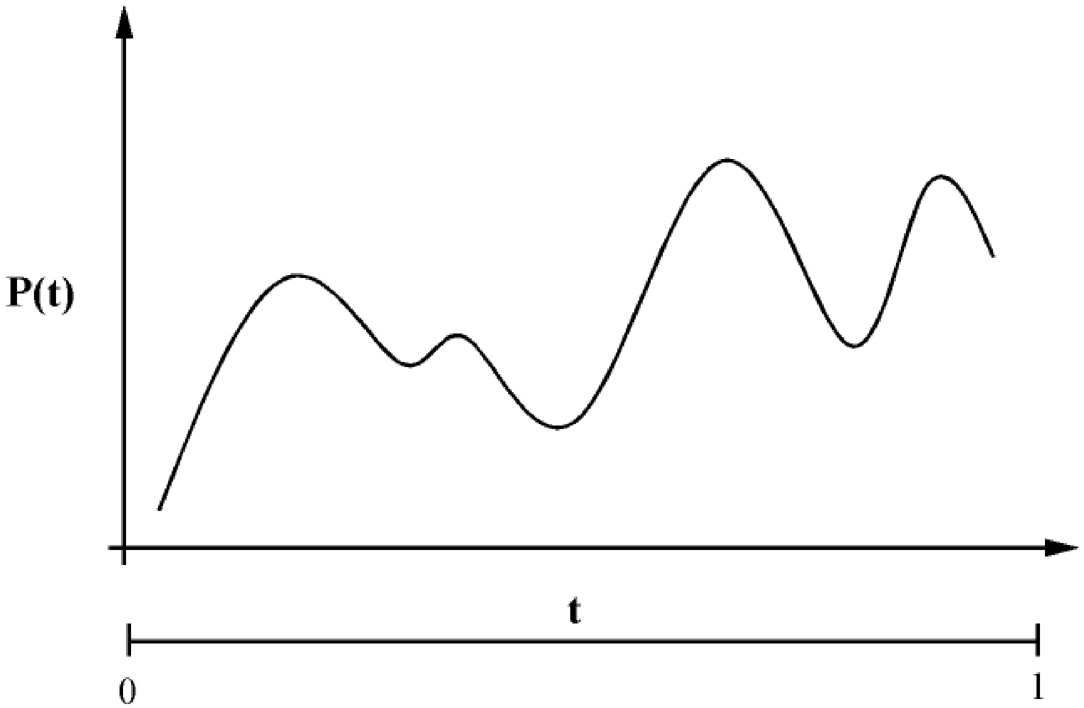
LCG - COPPE/SISTEMAS - UFRJ

Rio de Janeiro - Brazil

Motivation

- Reparametrization of motion captured data
  - – generation of slow-down and acceleration effects.
  - – synchronization of mocap animation with audio.
- Other applications

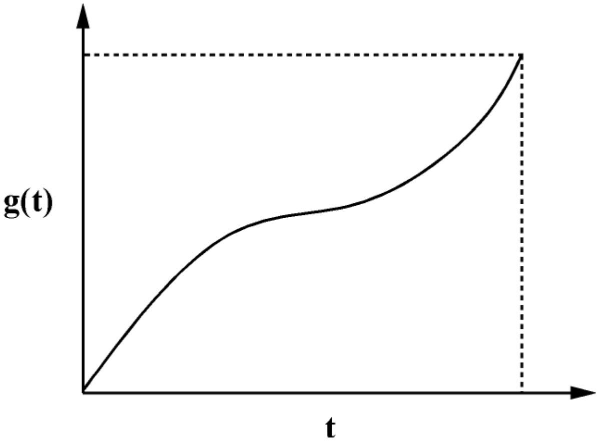
Continuous Objects in CG



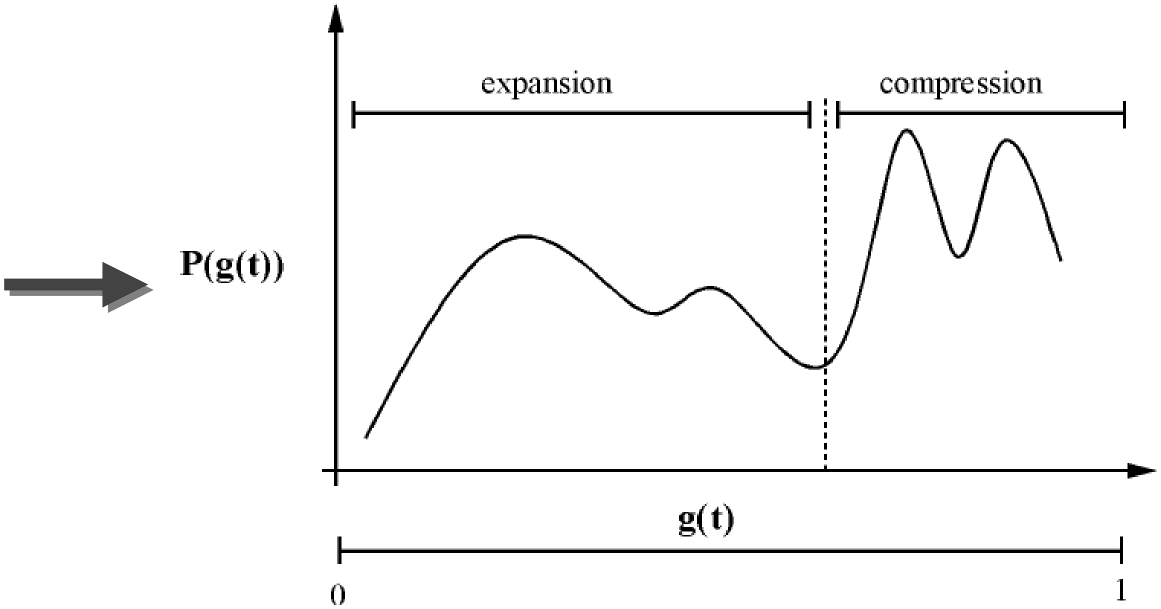
continuous interval

- usually described as parametric functions
- example: splines

Reparametrization

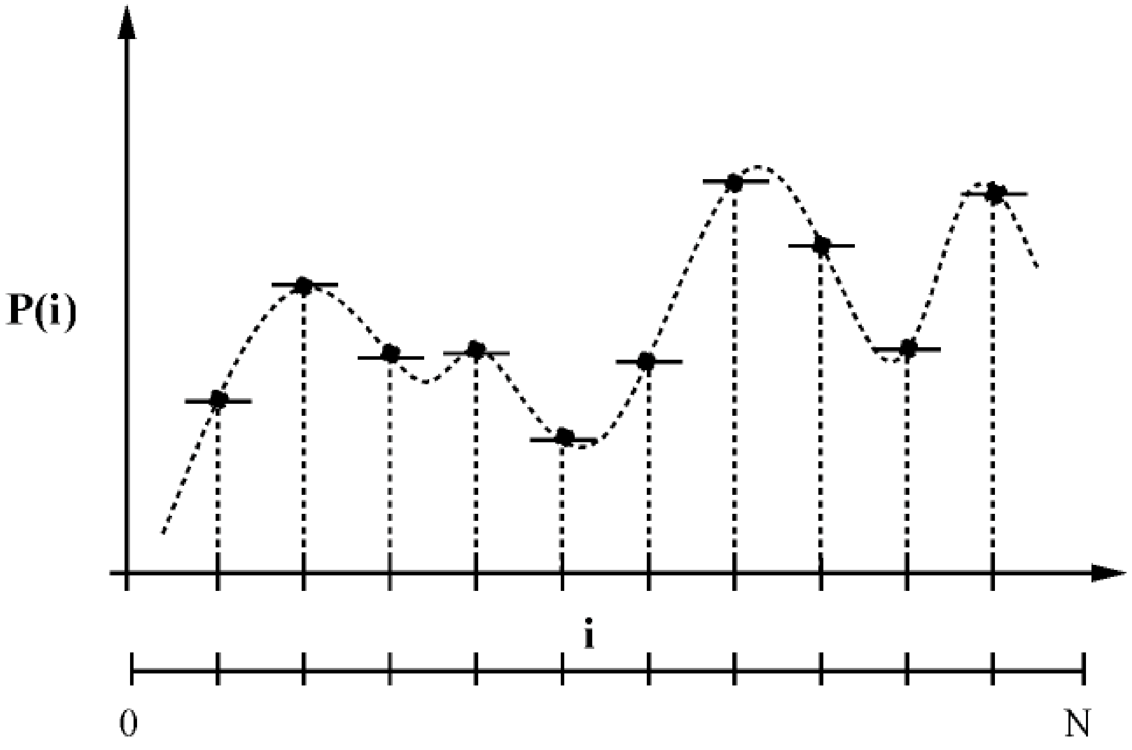


reparametrization function



- change of parametric function
- warping (compression / expansion)

Discretization

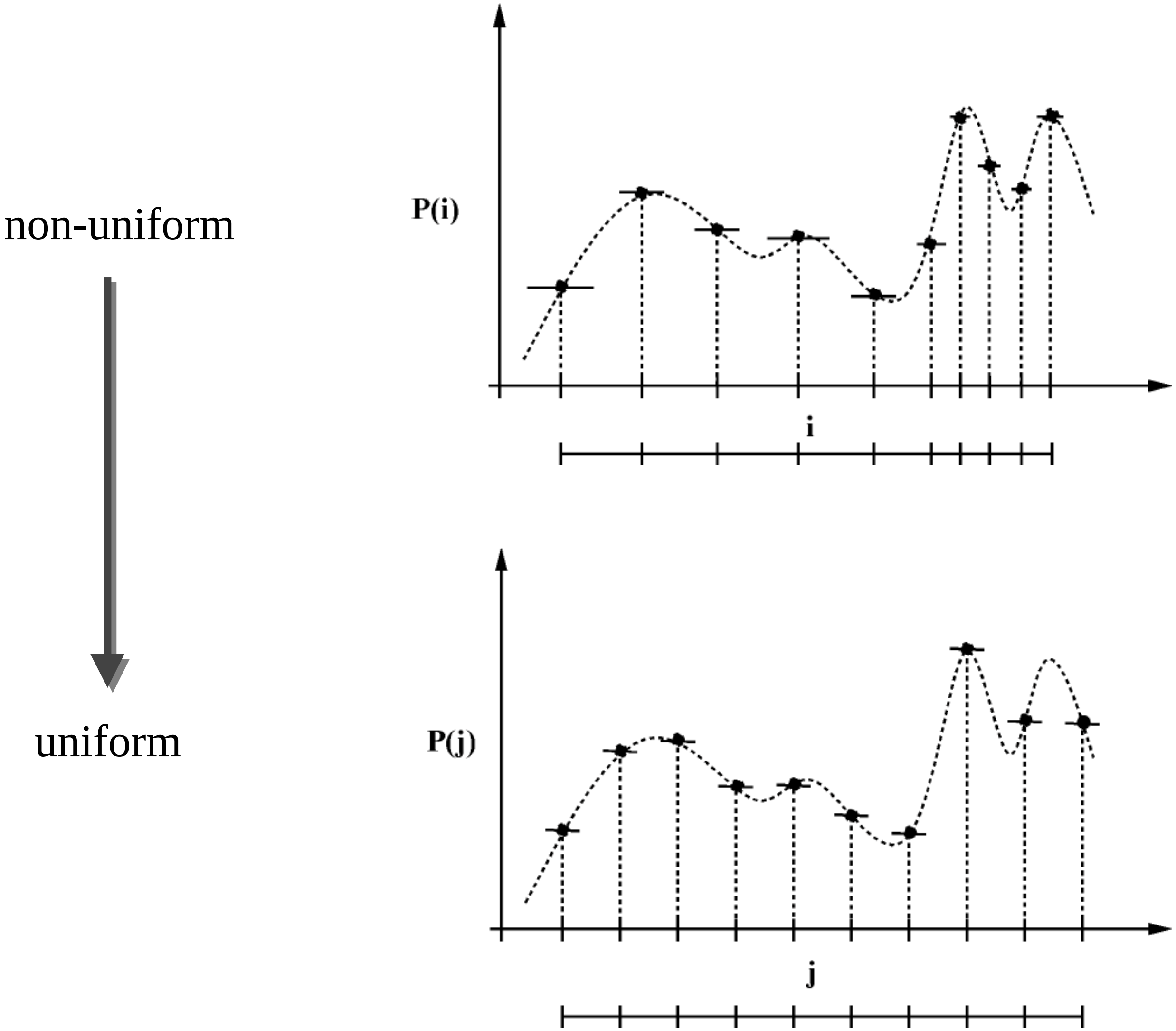


discrete interval

- samples of a continuous function
- example: motion capture data

Resampling

- Need uniform sampling rate

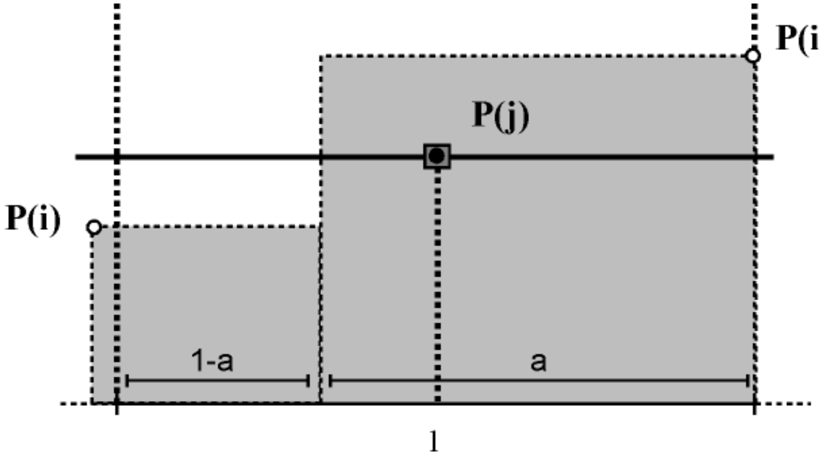


non-uniform

uniform

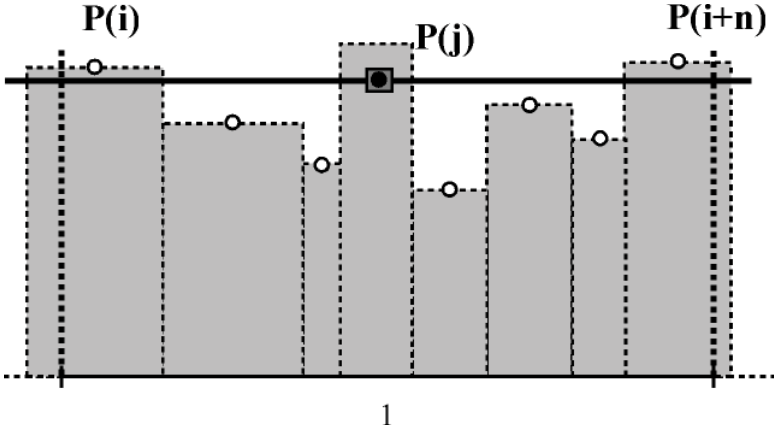
Antialiasing in Resampling

- Reconstruction and resampling



expansion

$P(j) = (1-a) . P(i) + a . P(i+1)$



compression

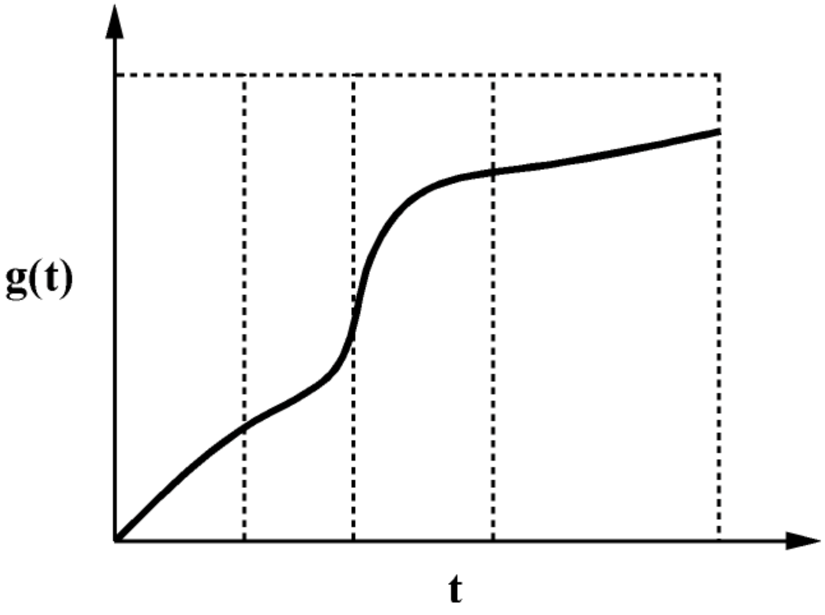
$$P(j) = \sum_{i=1}^{n} \omega_i . P(i)$$

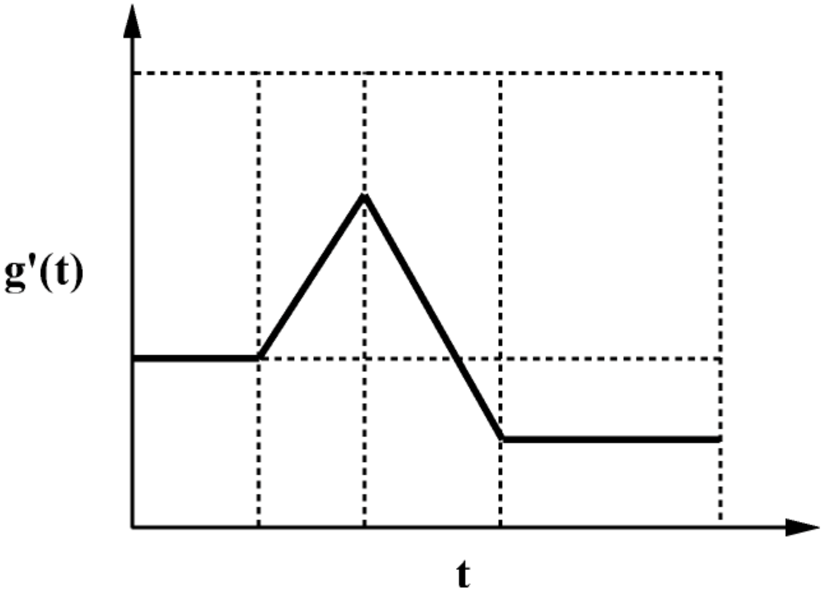
Time-dependent Applications

- Reparametrization changes “timing”
- Sound processing
  - “pitch” adjustment.
- Computer animation
  - velocity/acceleration adjustment.
- Video processing
  - slow-motion and accelerated-time effects.

Specification

- Derivative of reparametrization function (g’(t))
  - – more suitable for time-dependent applications.





Discrete Reparametrization

- Possible solution
  - reconstruct the motion curve from samples
  - reparametrize the continuous curve and sample it.

  ☒ very time and memory consuming.

- Our solution
  - reparametrization in the discrete domain.
  - local resampling according to a velocity curve.

  ☑ more efficient and natural.

Overview of Computation

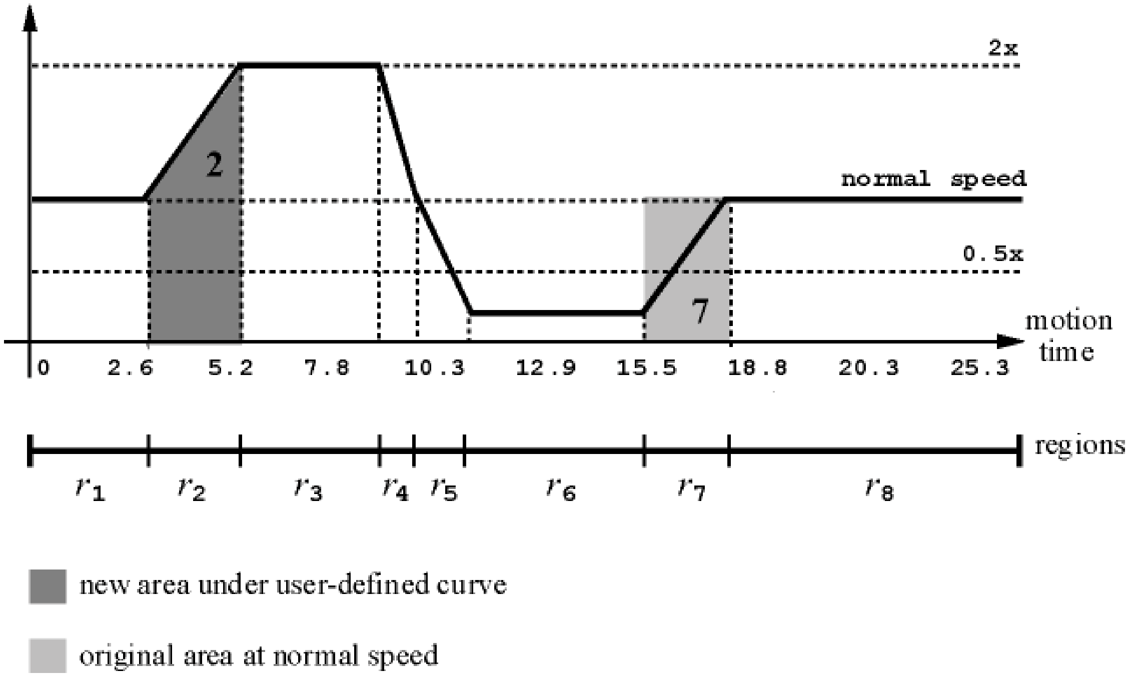
- Input
  - original sampled data.
  - user-defined velocity.

- Processing
  - discrete reparametrization.

- Output
  - new set of samples.

Identifying Regions of Change

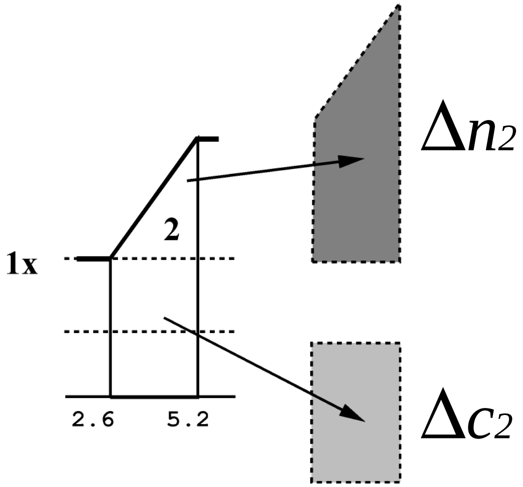
- Velocity function
  - defined over the temporal description of the original signal.
  - comprises regions r_i of monotonic increase and/or decrease.



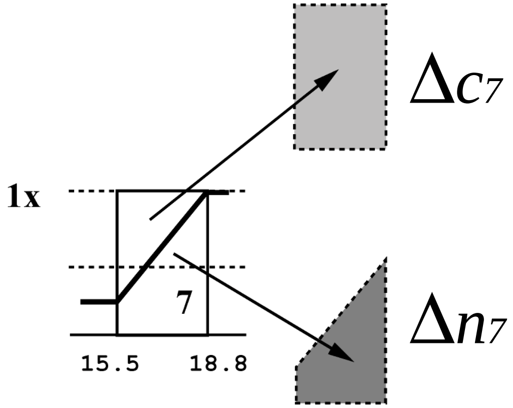
Computing Warp Factors

- Expansion/Compression regions
  - detected by using ratio $\Delta r_i = \Delta c_i / \Delta n_i$.
  - Δr_i is used to calculate new number of samples.

- expansion ($\Delta r_i > 1$)



- compression ($\Delta r_i < 1$)



Algorithm Description

- Identification of regions r_i

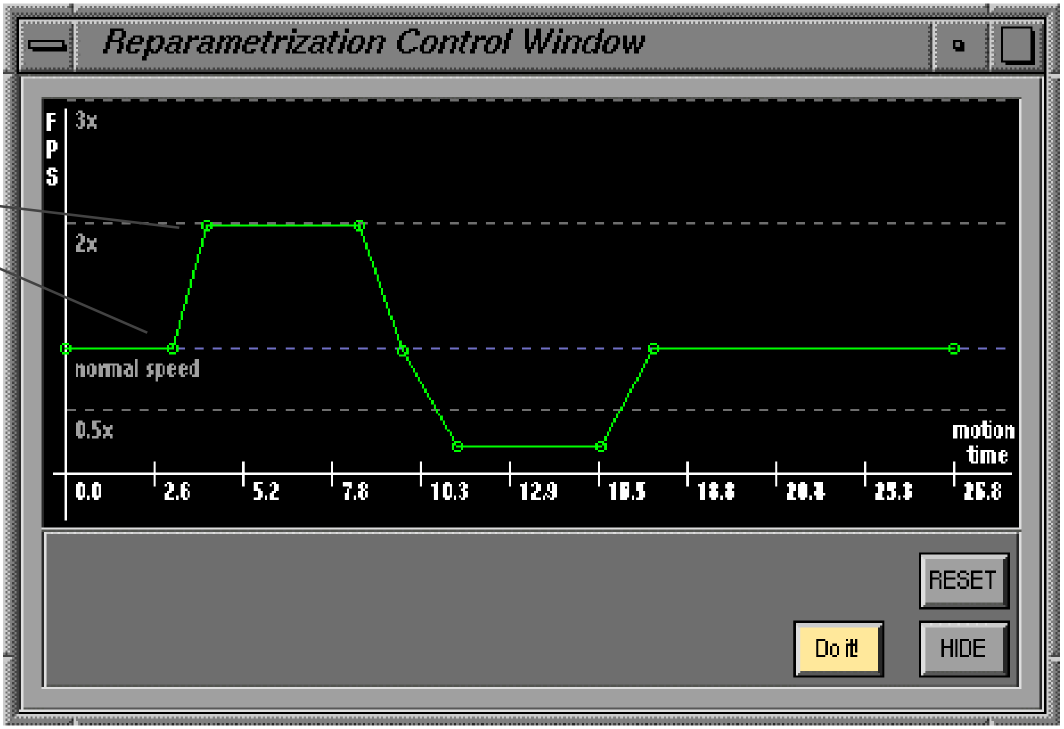
- Computation of warp factors Δr_i

- Calculation of new number of samples
  - for each region r_i : $NS_{r_i} = \Delta r_i \,.\, NS_{\mathrm{orig}}$
  - total number of samples: $\sum_{i=1}^{n} NS_{r_i}$

- Antialiasing where necessary

User Interface

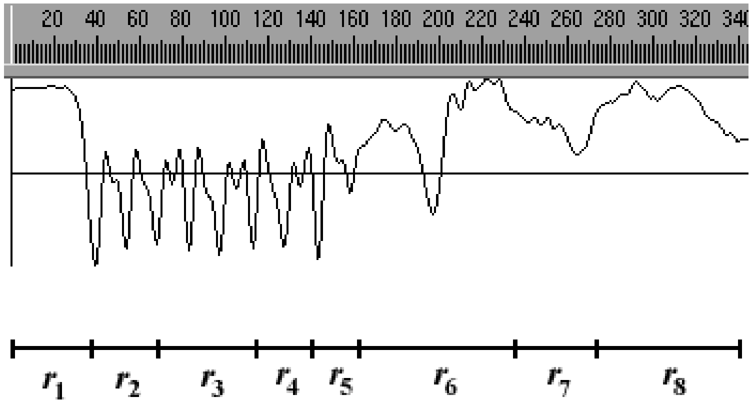
- Velocity function



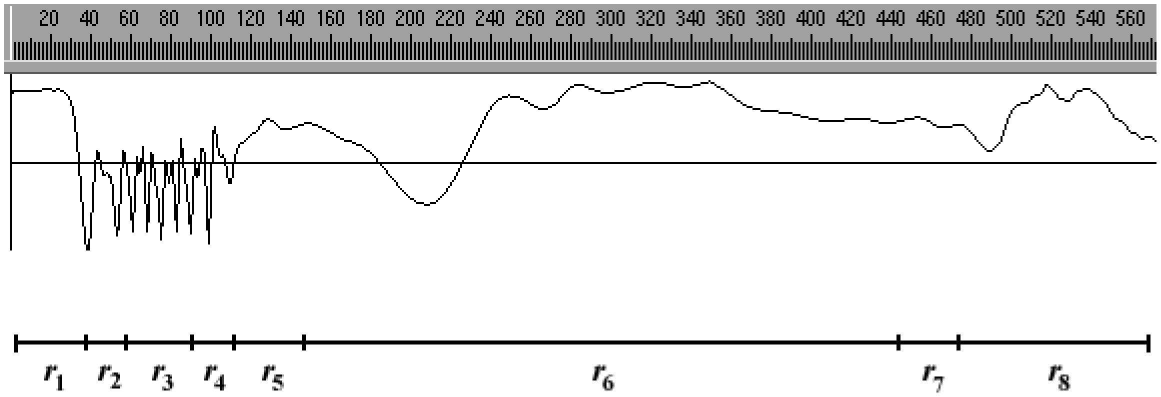
control points

Example

- original data



- reparametrized data



Video

- Original motion captured data

- Application of the algorithm using in-house animation system.

- New motion data.

Conclusions / Future Work

- Conclusion
  - velocity function: natural interface for time-dependent applications.
  - discrete reparametrization: fast and efficient (low memory usage).
- Future work
  - applications in sound and image processing.
  - applications in modeling.

Additional Info

http://www.visgraf.impa.br/mocap