Computer Animation Algorithms and Techniques

Optimization & Constraints

Enforcing Soft and Hard Constraints

Soft constraints - Minimizing energy terms

Hard constraints – constrained optimization

Example: Space-time constraints

Constrained optimization

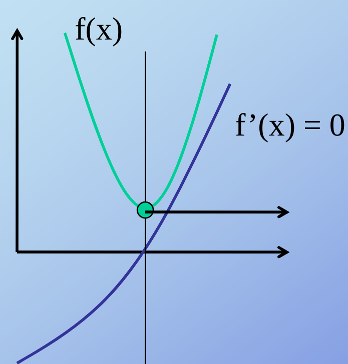
Use whenever the best, shortest, least error, is needed:

Fit surface with least curvature to set of points

Reach for object with minumu torque

Find motion in database whose start pose is closest to end of current motion

Minimum (or maximum) of a function



Ballistic motion

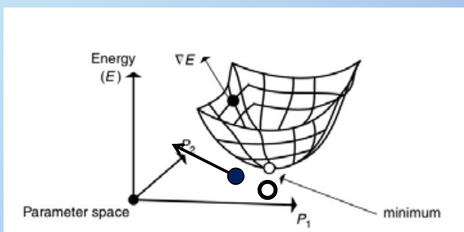
$$f(t) = p_0 + v_0 t - \frac{1}{2} g t^2$$

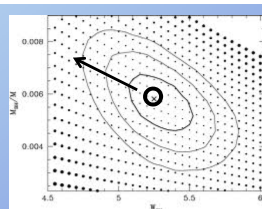
$$f'(t) = v_0 - gt \quad = 0$$

$$t = \frac{v_0}{g}$$

Analytic derivative & solution

Minimum (or maximum) of a function





Multivariate case

$$\nabla f(\vec{x}) = \begin{bmatrix} \frac{\partial f}{x_1} \\ \frac{\partial f}{x_2} \\ \frac{\partial f}{x_3} \\ \ldots \end{bmatrix} = 0$$

Gradient / Jacobian

Energy function

Define energy terms
- in terms of geometric features
- =0 for desirable configuration
- increases for less desirable configurations

Useful geometric features– easy to compute
- Parametric position function P(u,v)
- Surface normal function, N(u,v)
- Implicit function I(x) – distance to surface

Search parameter space
- modify parameters to reduce energy

Useful Constraints

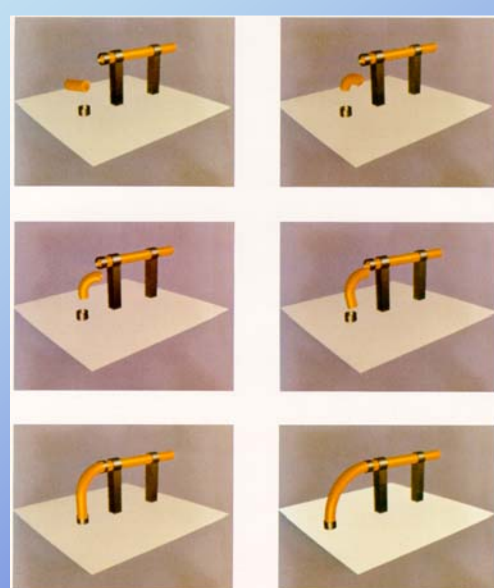
$$E = \left| P(u,v) - Q \right|^2$$

$$E = \left| P^a(u_a, v_a) - P^b(u_b, v_b) \right|^2$$

$$E = \left| P^a(u_a, v_a) - P^b(u_b, v_b) \right|^2 + N^a(u_a, v_a) \bullet N^b(u_b, v_b) - 1.0$$

$$E = (I^b(P^a(u_a, v_a)))^2$$

Pipe assembly



Finding the minimum

minimize $f(\vec{x})$

Analytic derivative/gradient

$\nabla f(\vec{x})$

Numeric approximation to derivative/gradient

$$\nabla f \approx \frac{f(\vec{x}_i) - f(\vec{x}_{i-1})}{\Delta x}$$

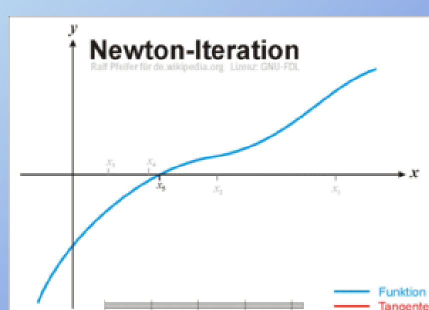
Analytic solution

$\nabla f(\vec{x}) = 0$

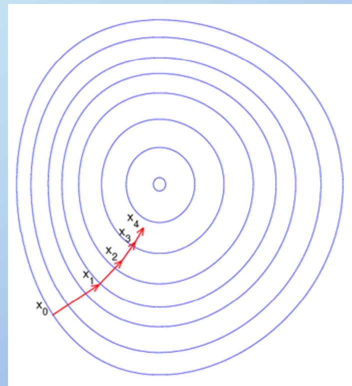
Numeric solution by iterative search
Newton's method
Steepest Descent
Conjugate Gradient Method

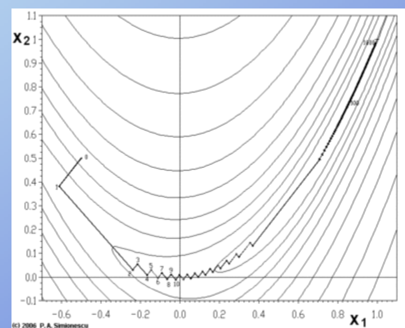
Newton's method

$$f'(\vec{x}) = 0 \qquad x_{i+1} = x_i - \frac{f'(\vec{x}_i)}{f''(\vec{x}_i)}$$

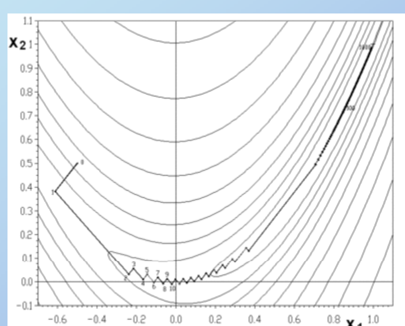


Steepest descent – step in direction of negative of gradient





Conjugate gradient method



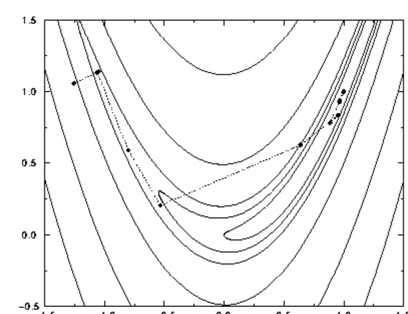


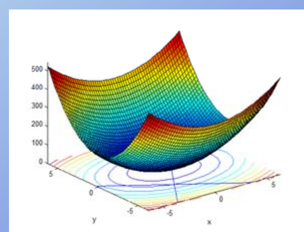
Figure 10: Conjugate gradient minimization path for the two-dimensional Rosenbrock function.

Constrained Optimization

minimize $f(\vec{x})$

$g_i(\vec{x}) = 0$ Equality constraints

$h_i(\vec{x}) \geq 0$ Inequality constraints



Rick Parent Computer Animation

Lagrange multipliers

$$\Lambda(x, y, \lambda) = f(x, y) + \lambda g(x, y)$$

$$\nabla \Lambda(x, y, \lambda) = \nabla f(x, y) + \nabla \lambda g(x, y) = 0$$

$$\nabla_{xy} \Lambda(x, y, \lambda) = \nabla_{xy} f(x, y) + \nabla_{xy} \lambda g(x, y) = 0$$

$$\nabla_{\lambda} \Lambda(x, y, \lambda) = g(x, y) = 0 \quad .$$

Rick Parent Computer Animation

Spacetime constraints

http://www.cs.cmu.edu/~aw/pdf/spacetime.pdf

Constrained optimization problem in time and space

Constraints – e.g.
locating certain points in time and space
Non penetration constraints

Objective function – e.g.
Minimize the amount of force used over time interval
Minimize maximum torque.

Spacetime constraint example

Particle position is function of time, x(t)

Time-varying force function, f(t)

Equation of motion

$$m\ddot{x}(t) - f(t) - mg = 0$$

Given f(t), $x(t_0)$, $\dot{x}(t_0)$ - integrate to get x(t).

Spacetime constraint example

Need to determine f(t)

Subject to constraints $x(t_0) = a$, $x(t_1) = b$

Objective: to minimize total force
Objective function:

$$R = \int_{t_0}^{t_1} \left| f \right|^2 dt$$

Rick Parent Computer Animation

Spacetime constraint example

Use discrete x(t), f(t), R, constraints
Time derivatives approximated by finite differences

$$\dot{x}_i = \frac{x_i - x_{i-1}}{h} \qquad \ddot{x}_i = \frac{x_{i+1} - 2x_i + x_{i-1}}{h^2}$$

Constraints:

$$\begin{cases} p_i = m\dfrac{x_{i+1} - 2x_i + x_{i-1}}{h^2} - f_i - mg = 0 \\ c_a = x_1 - a = 0 \\ c_b = x_n - b = 0 \end{cases}$$

R – minimize discrete version subject to constraints.

$$R = h \sum_i \left| f \right|^2$$

Rick Parent Computer Animation

Spacetime constraint canonical form

Numerical Solution – canonical form

S_j – collection of scalar independent variables

x, y, z components of the x_i’s and f_i’s

$R(S_j)$ – objective function to be minimized

sum of forces squared used at each time step

$C_i (S_j)$ – collection of scalar constraint functions => 0.

components of p_i’s, c_a, and c_b .

Spacetime constraint - canonical

Numerical problem statement

Find S_j that minimizes $R(S_j)$ subject to $C_i(S_j) = 0$

Numerical solution method:
- Request values of R and C_i for given S_j
- Access to derivatives of R and C_i with respect to S_j
- Iteratively provides updated values for solution vector S_i .

Rick Parent Computer Animation

Spacetime constraint

Sequential Quadratic Programming (SQP)

Computes second-order Newton-Raphson step in R

Computes first-order Newton-Raphson step in the C_i's

Projects the first onto the null space of the second
to the hyperplane for which all the C_i's are constant to the first order.

Spacetime constraint using SQP

Sequential Quadratic Programming (SQP)

Computes first-order Newton-Raphson step in the C_i's

Jacobian: $$J_{ij} = \frac{\partial C_i}{\partial S_j}$$

Computes second-order Newton-Raphson step in R

Hessian $$H_{ij} = \frac{\partial^2 R}{\partial S_i \partial S_j}$$

Plus the first derivative vector: $$\frac{\partial R}{\partial S_j}$$

Spacetime constraint example

$$p_i = m\frac{x_{i+1} - 2x_i + x_{i-1}}{h^2} - f_i - mg = 0$$

Matrices

$$\frac{\partial p_i}{\partial x_j} = 2m/h^2 \qquad \text{i=j}$$

$$= -m/h^2 \qquad \text{i=j-1, j+1}$$

$$= 0 \qquad \text{otherwise}$$

$$\frac{\partial p_i}{\partial f_j} = 1 \qquad \text{i=j}$$

$$= 0 \qquad \text{Otherwise.}$$

Spacetime constraint example

Matrices

$$\frac{\partial R}{\partial f_i} = 2 f_i$$

$$\frac{\partial^2 R}{\partial f_i \partial f_j} = 2 \qquad \text{i=j}$$

$$= 0 \qquad \text{Otherwise.}$$

Spacetime constraint example

SQP step

Solve 2 linear systems in sequence

1

$$-\frac{\partial R}{\partial S_i} = \sum_j H_{ij} \hat{S}_j$$

Yields a step that minimizes a 2nd order approx. to R

2

$$-C_i = \sum_j J_{ij} (\tilde{S}_j + \hat{S}_j)$$

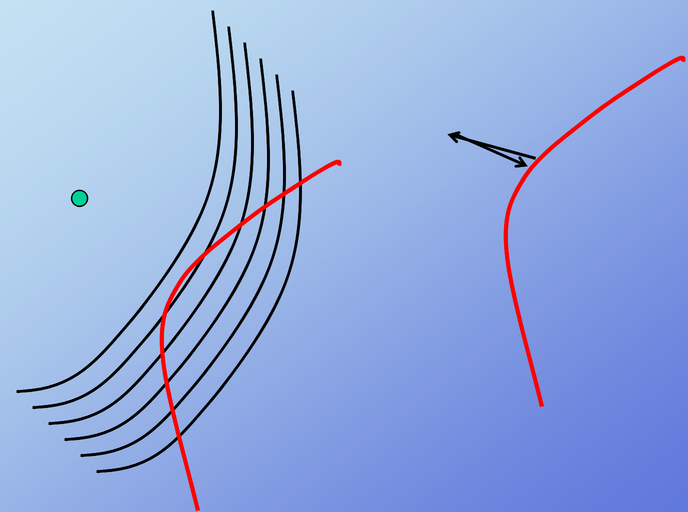
Yields a step that drives linear approx. to C_i's to zero
And projects optimization step S_j to null space of constraint Jacobian.

Spacetime constraint example

Final update: $\tilde{S}_j + \hat{S}_j$

Reaches fixed point:
- When C_i's = 0
- Any further decrease in R violates constraints.

Constrained optimization



Rick Parent Computer Animation

Spacetime constraint example

Note about Linear system solving

Large matrices often with spacetime problems

Inverting is $O(n^3)$

Spacetime problems almost always sparse

Over and under constrainted systems easily arise

Under constrained: pseudo-inverse

Pseudo-inverse for sparse matrix
  sparse conjugate gradient algorithm: $O(n^2)$.

Rick Parent Computer Animation