

Ladislav Kavan's Physically Based Simulation 2017ed

Edited by wxgopher

May 9, 2019

Contents

1	Classical Mechanics	2
1.1	Basics: a harmonic oscillator	2
2	Time Integration	2
3	Optimization	2
3.1	Implicit Newmark	2
3.2	Optimization Problems	2
3.2.1	Solving unconstrained problems	3
3.2.2	Newton's method	3
3.3	Numerical Linear Algebra	4
4	Elastic Materials and Finite Element Simulation	4
5	Something more...	4
6	Papers and books	4

This lecture note is based on Dr. Ladislav Kavan's Physics-based Animation course CS6660, originally taught at the University of Utah. I also try to include some supplementary materials from other resources.

Warning: Although as I strive to make this material useful, there are certain bugs, use this material at your own risk. I would also be grateful to hear [feedbacks](#).

.....

1 Classical Mechanics

1.1 Basics: a harmonic oscillator

2 Time Integration

3 Optimization

3.1 Implicit Newmark

$$x_{n+1} = x_n + \frac{h}{2}(v_n + v_{n+1}), \quad (1a)$$

$$v_{n+1} = v_n + \frac{h}{2}(f_n + f_{n+1}), \quad (1b)$$

where x and v are positions and velocities, and $f_n \equiv f(x_n)$. Thus we have

$$x_{n+1} - x_n - hv_n = \frac{h^2}{4}(f_n + f_{n+1}). \quad (2)$$

Let $y = x_n + hnv_n + \frac{h^2}{4}f_n$, $x_{n+1} = x$, note that $\nabla E = -f$, then (2) becomes

$$x - y = \frac{h^2}{4}f \Rightarrow x - y + \frac{h^2}{4}f\nabla E = 0. \quad (3)$$

Let $g(x) = \frac{1}{2}\|x - y\|^2 + \frac{h^2}{4}E$, then solving g equals to solve x for

$$\min_x g(x). \quad (4)$$

3.2 Optimization Problems

Problem formulation:

$$\min_x g(x), \quad x \in R^n, \quad g \in R. \quad (5)$$

Optimization problems can be categorized into constrained or unconstrained problems, or convex or non-convex problems.

Theorem 1. For a convex problem (where both objective and feasible set are convex), if the objective is C^2 , then the Hessian $H \succeq 0$, and the local minimum is the global minimum.

Definition 1. A linear programming (LP) is a problem with linear objective and linear equality or inequality constraints. A quadratic programming (QP) is **the same as LP** except with a quadratic objective.

Note 1. Convex QP has polynomial time solver but non-convex QP is NP-hard.

Example 1. A non-convex QP:

$$\min_x \frac{1}{2} \|Ax\|^2, \quad (6a)$$

$$s.t. \quad \|x\|_2 = 1. \quad (6b)$$

Note 2. Software package for solving non-convex problem: IpOPT, KNITRO, or [NEOS-Guide](#)

3.2.1 Solving unconstrained problems

There are two ways to solve an unconstrained problem: descent method or trust-region method.

Descent method refers to pick a descent direction d and do an exact or inexact line search (LS) to determine descent distance αd .

Definition 2. Descent direction: $\forall \alpha \in (0, \alpha_0), \alpha_0 > 0, g(x + \alpha d) < g(x)$.

Definition 3. Exact LS refers to solving the following problem:

$$\arg \min_{\alpha > 0} g(x + \alpha d).$$

Backtracking LS

3.2.2 Newton's method

A usual descent direction is called gradient descent (GD), denoted by $d = -\nabla g$. It doesn't work well sometime, for example:

Example 2.

$$\min_{x_1, x_2} g(x_1, x_2) = \frac{1}{2}(x_1^2 + \gamma x_2^2), \quad \gamma > 0. \quad (7)$$

If we do exact LS, we get

$$x_1^k = \gamma \left(\frac{\gamma - 1}{\gamma + 1} \right)^k, \quad (8a)$$

$$x_2^k = \left(-\frac{\gamma - 1}{\gamma + 1} \right)^k. \quad (8b)$$

If gamma becomes huge, we can tell GD converges slowly (as two GD directions are perpendicular).

Newton's method refers to do a quadratic approximation to problem (5), given by

$$g(x + p) = g(x) + \nabla g^T(x)p + \frac{1}{2}p^T \nabla^2 g(x)p + O(\|p\|^2) \equiv \tilde{g}(x) + O. \quad (9)$$

Thus we have

$$\frac{\partial \tilde{g}}{\partial p} = 0 \iff \nabla g + \nabla^2 g p = 0.$$

The Newton descent direction is given by

$$p = -(\nabla^2 g)^{-1} \nabla g.$$

Note 3. Newton's method is affine invariant. Let $x = Ty$ with T being a nonsingular matrix, so problem (5) is equivalent to problem

$$\min_y \tilde{g}(y). \quad (10)$$

We also have

$$\nabla \tilde{g} = T^T \nabla g, \quad \nabla^2 \tilde{g} = T^T \nabla^2 g T.$$

Newton direction Δy of \tilde{g} and Δx of g is given by

$$\Delta y = -T^{-1} \nabla^2 g \nabla g = T^{-1} \Delta x.$$

3.3 Numerical Linear Algebra

4 Elastic Materials and Finite Element Simulation

5 Something more...

6 Papers and books

1. Nonlinear Continuum Mechanics for Finite Element Analysis by J. Bonet
2. Convex optimization by S. Boyd
3. Numerical optimization by Nocedal