Some Constrained Simulations

· constraints for cloth representation

Large steps in cloth simulation (1998)

· force-based implicit solver

A versatile and Robust Model for Geometrically Complex Deformable Solids (2004)

· generalized constrained dynamics

Verlet's integration ('60)

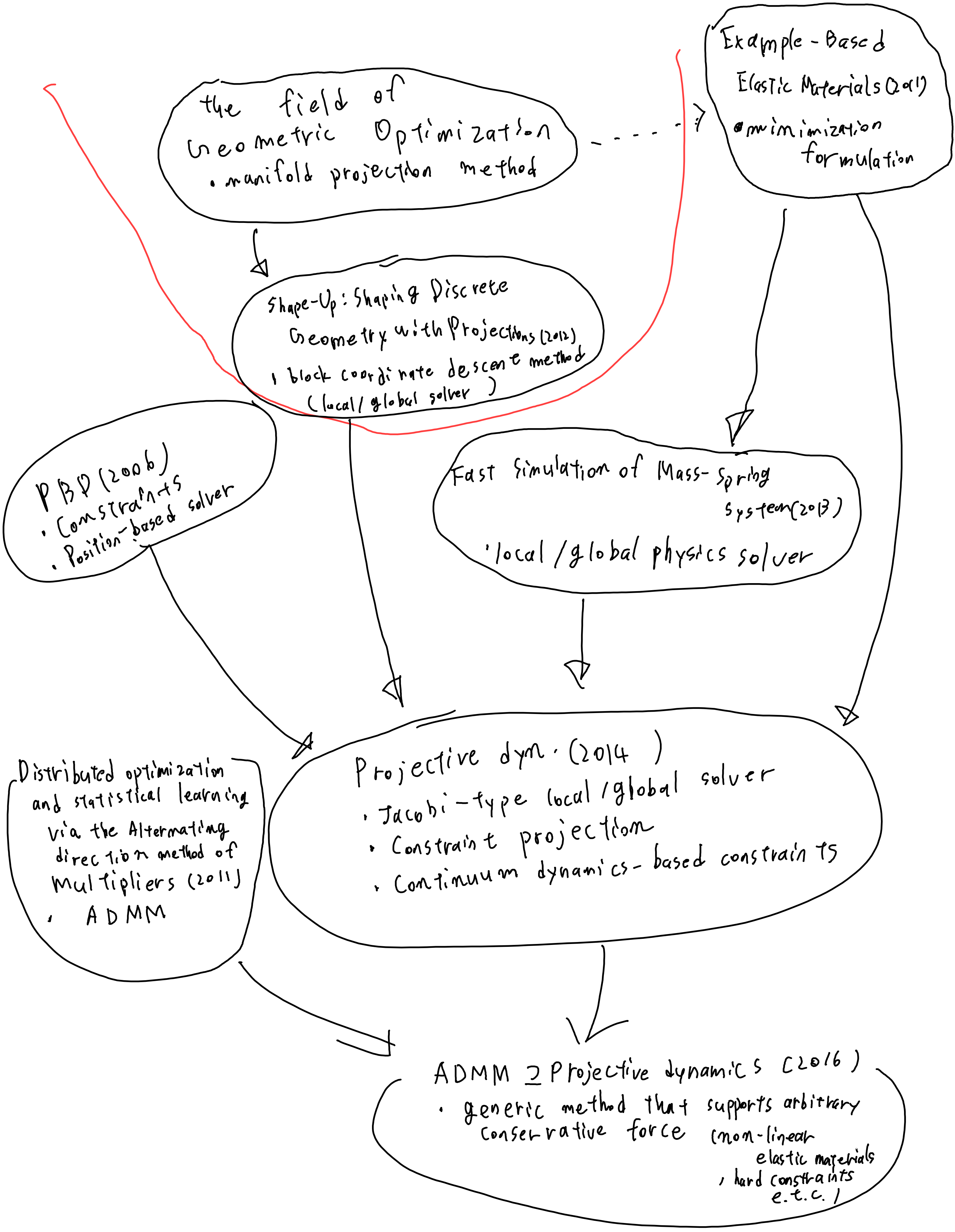
Advanced Character Phisics (2001)

· Position-based method

← force-based

position-based →

Position Based Dynamics (2006)


the field of
Geometric Optimization
・manifold projection method
Example-Based
Elastic Materials (2011)
・minimization
formulation
Shape-Up: Shaping Discrete
Geometry with Projections (2012)
・block coordinate descent method
(local/global solver)
PBD (2006)
・Constraints
・Position-based solver
Fast Simulation of Mass-Spring
System (2013)
・local/global physics solver
Distributed optimization
and statistical learning
via the Alternating
direction method of
multipliers (2011)
・ADMM
Projective dyn. (2014)
・Jacobi-type local/global solver
・Constraint projection
・Continuum dynamics-based constraints
ADMM ⊇ Projective dynamics (2016)
・generic method that supports arbitrary
conservative force (non-linear
elastic materials
, hard constraints
e.t.c.)

PBD
- position based method

Efficient Simulation of Inextensible Cloth (2007)
- fast projection

Interactive Simulation of Elastic Deformable Materials (2006)
- constraints with the phisically-based parameters
- compliant constrahe formulation

Strain based Dymamics (2014)
- The constraints from <u>strain tensor</u>
- Decoupling stretch from shear resistance

XPBD has an advantage against to Strain based Dym.

X PBD (2017)
- Physicall-based constraints
- total Lagrange multiplier ← This formulation resolves the iteration dependency

- Physics
  - Newtonian mechanics
  - Lagrangian mechanics
  - continuum Mechanics
    - what is Tensor
    - force, strain, and stress
    - some tensors
      - kirchhoff stress tensor
      - Cauchy-Green deformation tensor
      - St. Venant strain tensor

      from dynamic deformables

- Numerical Integration
  - The linear solvers
  - Newton's method
  - Iterative solver and what is global solver
    - Itrative
      - Jacobi type
      - Gauss seidel type
    - global
  - misc. topics
    - Lagrange Multiplier
    - LU decomposition
    - Schur decomposition

- Numerical physics
  - mass-spring system
  - Verlet's integration
  - Explicit/Implicit euler scheme
  - constrained Dynamics
  - shape matching

column

Eulerian/Lagrangian

Newton, Euler, Lagrangian,
Lagrange, Hamiltonian, Hessian
Jacobi, Newton-Raphthon, Newton-Euler equation
Jacobian

-
- Interpret the papers
  - Position Based Dynamics
    - Large steps in Cloth Simulation (1998)
    - Advanced Character Physics (2001)
    - A versetile and Robust Model for Geometrically complex Deformable solids (2004)
    - Position Based Dynamics (2006)
  - XPBD
    - Geometric, Variational Integrators for Computer Animation (2006)
    - Interactice simulation of Elastic Deformable Materials (2006)
    - Efficient simulation of Inextensible Cloth (2007)
    - XPBD (2017)
    - Small steps in Physics simulation (2018)
  - Projective Dynamics

⋮

⋮

- Projective Dynamics
  - Example-Based Elastic Materials (2011)
  - Fast simulation of Mass-Spring System (2013)
  - Projective Dynamics
  - ADMM $\supseteq$ Projective Dynamics
- Post XPBD

XPBD P.3.

$$M(x^{n+1}-\tilde{x})-\nabla C(x^{n+1})^T\lambda^{n+1}=0 \quad \cdots ①$$

$$C(x^{n+1})+\tilde{\alpha}\lambda^{n+1}=0 \quad \cdots ②$$

Let $x^{n+1}=x^n+\Delta x,\ \lambda^{n+1}=\lambda^n+\Delta\lambda \quad \cdots ③$

and from ① $M(x^n+\Delta x-\tilde{x})-\nabla C(x^{n+1})^T(\lambda^n+\Delta\lambda)=0,$

$$M\Delta x-\nabla C(x^{n+1})^T\lambda^n+\nabla C(x^{n+1})^T\Delta\lambda=-M(x^n-\tilde{x})$$

$$\to M\Delta x-\{\nabla C(x^{n+1})-\nabla C(x^n)\}\lambda^n-\nabla C(x^n)\lambda^n$$

$$-\{\nabla C(x^{n+1})-\nabla C(x^n)\}\Delta\lambda-\nabla C(x^n)\Delta\lambda^n=-M(x^n-\tilde{x})$$

$$M\Delta x-\nabla C(x^n)\Delta\lambda^n-(\nabla C(x^{n+1})-\nabla C(x^n))(\lambda^n+\Delta\lambda)=-g(x,\lambda)$$

$$M\Delta x-\nabla C(x^n)\Delta\lambda^n-\frac{\partial C(x)}{\partial x}(\lambda^n+\Delta\lambda)^T\Delta x=-g(x,\lambda)$$

$$\left(M-\frac{\partial C(x)}{\partial x}\right)\Delta x-\nabla C(x^n)\Delta\lambda^n-\underbrace{\frac{\partial C(x)}{\partial x}^T\Delta\lambda\Delta x}_{\text{drop } (\because \Delta\lambda\Delta x \ll 1)}=-g(x,\lambda)$$

Hessian

$$\left(M-\frac{\partial C(x)}{\partial x}\right)\Delta x-\nabla C(x^n\Delta\lambda^n)=-g(x,\lambda) \quad \cdots ⑤$$

$K=\frac{\partial g}{\partial x}$ and from ③ into ②,

$$C(x^n+\Delta x)+\tilde{\alpha}(\lambda^n+\Delta\lambda)$$

$$\underbrace{C(x^n+\Delta x)-C(x_n)}_{\nabla(x_i)\Delta x}+C(x_n)+\tilde{\alpha}\Delta\lambda=-\tilde{\alpha}\Delta\lambda \quad \cdots ⑥$$

finally,

$$\begin{bmatrix} K & -\nabla C^T(x_i) \\ \nabla C(x_i) & \tilde{\alpha} \end{bmatrix}\begin{bmatrix} \Delta x \\ \Delta\lambda \end{bmatrix}=-\begin{bmatrix} g(x_i,\lambda_i) \\ h(x_i,\lambda_i) \end{bmatrix}$$