Distributed Optimization of Continuous-time Multi-agent Networks

Yiguang Hong

Academy of Mathematics & Systems Science
Chinese Academy of Sciences
Outline

1. Background
2. Formulation
3. Continuous-time optimization
   1. Fundamental problem
   2. Optimization for physical systems
4. Conclusions
1. Background

**Convex optimization:** \( \min f(y), \quad y \in \mathbb{R}^m \)

where \( f \) is convex

If \( f(y) \) is differentiable, the discrete-time dynamics:

\[
y(t+1) = y(t) - k \nabla f(y(t)), \quad k > 0
\]

or continuous-time dynamics:

\[
\dot{y}(t) = -k \nabla f(y(t)), \quad k > 0
\]
Distributed design

Agent → Multi-agent systems (MAS): distributed design in a network without a center

- Network topology and Information flow
- Design of protocols and algorithms
- Complexity (unbalanced, uncertain, asynchronous, heterogeneous, ...)
Consensus: the basic problem

Agent dynamics: \( \frac{dx_i}{dt} = u_i \), \( i = 1, \ldots, 2 \)
Leader (or desired position): \( x_0 \)

Neighbor-based communication (\( N_i \): the neighbor set of agent \( i \))

Distributed control: \( u_i = \sum_{j \in N_i} (x_j - x_i) \)

Multi-agent consensus (agreement, synchronization):
- Leader-following: \( x_i - x_0 \rightarrow 0 \)
- Leaderless: \( x_i - x_j \rightarrow 0 \)
Distributed optimization

- **Distributed optimization**: Optimization (task) + distributed design (consensus)
- Distributed **convex** optimization $\rightarrow$ distributed matrix optimization, distributed MPC and dynamic programming, ...

- **Applications**: industry and energy (smart grids, sensor network, manufacture), economics and society (social networks, marketing, traffic), biology and ecology ......
2. Formulation

Convex optimization: \( \min_{z \in R^n} f(z) \)

with \( f(z) \) convex

→ Distributed version: \( f(z) = \sum_{i=1}^{n} f_i(z) \)

• each agent \( i \) knows its own cost function \( f_i \) or its gradient \( \nabla f_i \)

• Local cost function \( f_i \) may not have the same optimal solution of \( f \)
Distribution formulation

Convex optimization: \( \min_{z \in \mathbb{R}^m} f(z) \)

Constraints: \( g(z) \leq 0; \ z \in \Omega, \) with \( g(z), \ \Omega: \) convex

\( \rightarrow \) Distributed version: \( f(z) = \sum_{i=1}^{n} f_i(z) \)

Constraints:

- **Global**: known by every agent \( \rightarrow \) conventional one
- **Local**: for agent \( i \): \( g_i(z_i) \leq 0 \) and/or \( z_i \in \Omega_i \) (with \( \Omega = \) nonempty intersection of all local constraint sets \( \Omega_i \))
- **Coupled**: \( g(z_1, z_2 \ldots z_n) \leq 0 \)
Preliminaries: convex analysis

- A function $f(x)$ is **convex** if
  
  $$f(cx + (1 - c)y) \leq cf(x) + (1 - c)f(y)$$

  for any $x, y$ and $0 < c < 1$.

- It is **strictly convex** if it is convex and “=” holds iff $x = y$.

- It is **strongly convex** if it is strictly convex and there is $\sigma$ such that
  
  $$f(cx + (1 - c)y) \leq cf(x) + (1 - c)f(y)$$
  
  $$-\frac{1}{2}\sigma c(1-c)\|x-y\|_2^2$$
Convex set

- $K$ is a convex set if, for $0 < \lambda < 1$
  
  $$(1 - \lambda)x + \lambda y \in K \quad x \in K, y \in K$$

- $d(x,K) :$ distance between set $K$ and $x$

  $$\|x\|_K \triangleq \inf \{\|x - y\| | y \in K\}$$
Preliminaries: graph

Graph for the interaction between agents $\rightarrow$ Laplacian or stochastic matrices

- Undirected or directed graph (balanced)
- Fixed or switched graph

Node (agent for computation)

Link (information flow)
Switching $\rightarrow$ joint connection

Joint connection: union graph in $[t, \infty)$ is connected for any $t$: a necessary condition

Uniform joint connection: $\exists T$, union graph on $[t, t+T]$ is connected
Many extensions …

• Non-convex optimization $\rightarrow$ constrained convex problem …
• Online or robust optimization: regret analysis …
• Zero-sum game (saddle point): $\min \max f(x, y)$
• Aggregative game
• Coverage: search/rescue, evasion/pursuit
• Machine learning …
Discrete-time optimization

Joint work with students (Y. Lou, G. Shi, Y. Zhang, P. Yi) and professors (Profs. Xie, Jonhasson, and Liu, et al)

• **Convex intersection** computation with approximate projection (IEEE TAC 2014, full paper): accurate projection $\rightarrow$ approximate projection set; the critical approximate angle.

• **Non-convex intersection** computation (IEEE Trans Wireless Communications 2015, full paper): ring intersection with application to localization even when the intersection set is empty.

• **Random sleep** algorithms (SCL 2013, CTT 2015): update with random sleep procedure, due to random failure, or sleep to save energy, or stochastic disturbance, etc.
Discrete time optimization

- **Zero-sum game** *(IEEE TAC 2016, full paper)*: the parties against each other to solve the saddle point problem; adaptive heterogeneous stepsizes for unbalanced graphs

- **Optimization with quantization** *(IEEE TCNS 2014, full paper)*: exact optimization can be achieved with one bit when the graph is fixed, with at most 3 bits when it is switching

- **Optimization with constraints** *(SIAM Control & Optimization, 2016, IEEE TAC, under review)*: convergence rate for stochastic algorithm, nonsmooth optimization with equality constraints
Recent Attention: continuous-time

Conventional optimization algorithm: discrete time

**Recent years:** analysis and design of continuous-time algorithm

- Few works done for continuous-time approximation or constrained optimization in the past: Arrow et al (1958), Ljung (1977), Brockett (1988), ...

- A way to connect discrete-time decision and continuous-time control
3. Continuous-time optimization

**Fundamental Problems:**
- Connectivity: time-varying graphs, balanced weights
- Uncertainty: communication, measurement, environment
- Constraints: local, coupled, ...

**Cyber-physical (hybrid) problems:**
- Communication cost: random sleep, event-based, quantization ...
- Disturbance rejection hybrid/hierarchical computation
- Complicated dynamics: nonlinear physical agents
Why continuous-time model?

New era $\rightarrow$ new problems:

- Optimization solved not with digital computers, but by physical systems
- Cyber-physical system: hybrid model with discrete-time communication and continuous-time physical systems
- New design viewpoint from continuous-time dynamics
- Maybe quantum computation?
# Comparison

<table>
<thead>
<tr>
<th></th>
<th><strong>Discrete-time</strong></th>
<th><strong>Continuous-time</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Tools</strong></td>
<td>Variational inequality, monotone property, fixed point …</td>
<td>Lyapunov function, passivity, input-output stability …</td>
</tr>
<tr>
<td><strong>Design</strong></td>
<td>Time-varying stepsize, ADMM, dual variable, …</td>
<td>Dynamic compensation, autonomous equation, singular perturbation …</td>
</tr>
<tr>
<td><strong>Theory</strong></td>
<td>Convex optimization, saddle-point dynamics, …</td>
<td>Nonlinear control, differential inclusion, robust control …</td>
</tr>
</tbody>
</table>
Continuous-time optimization

Joint work with G. Shi, Y. Lou, P. Yi, X. Wang, Z. Deng, Y. Zhang, X. Zeng, et al

3.1 Distributed convex intersection

- Basic formulation: Agent $\frac{dx_i}{dt} = u_i$ only knows the information of its own closed convex set $X_i$ and its neighbor $x_j \rightarrow$ the agents achieve consensus within $X_0 (= \cap X_i)$, which is not empty

- **Aim**: distributed algorithm with switching interaction topologies
Find a point in the intersection set of a group of convex set


- Projected consensus algorithm (PCA: a decentralized version of APC) with time-varying directed interconnection, or its randomized version; Nedic et al 2010, Shi et al, 2012…
Projected consensus algorithm (PCA)

PCA for continuous-time system: accurate projection for optimization + neighbor-based rule for consensus

- Centralized design $\rightarrow$ Distributed design: neighbor-based rule, not completely connected
- Conventional analysis $\rightarrow$ set analysis (non-smoothness)
- Switching interaction topology (non-smoothness): common Lyapunov function
Main Results (TAC 2013)

PCA: local projection for intersection + neighbor-based rule for consensus

Result 1: Global convex intersection of MAS uniformly jointly strongly connected Consistent with the existing discrete time results

Result 2: In the bidirectional case, MAS achieves global convex intersection = \([t, \infty)\) joint connection
Numerical simulation

$t = 0$ to $t = 1000$
Approximate PCA

PCA $\rightarrow$ APCA: approximate projection with $0 \leq \theta \leq \theta^* < \pi/2$ (APCA)

In practice, it is hard or expensive to get accurate projection $\rightarrow$ Approximately projected consensus algorithm (APCA) for unknown projection:

IEEE TAC 2014
Critical approximate angle

Suppose $\alpha_{ik} = 1$, $\theta_k = \theta \ \forall i, k$

$n \geq 1$ nodes

- $0 < \theta < \pi/4$ implies
  $$\sup_{x(0)} \lim_{k \to \infty} \sup_{x_i(k)} |x_i(k)|_{X_0} < \infty, \ i = 1, \ldots, n$$

$n = 1$ node

- $\theta = \pi/4$ implies
  $$\lim_{k \to \infty} \sup_{x_*(k)} |x_*(k)|_{X_*} < \infty$$

- $\pi/4 < \theta < \pi/2$ implies
  $$\lim_{k \to \infty} \sup_{x_*(k)} |x_*(k)|_{X_*} = \infty,$$
  $$|x_*(0)|_{X_*} > \sup_{y_1, y_2 \in X_*} |y_1 - y_2| / (\tan \theta - 1)$$
Approximate projection performs better than the accurate one! (IEEE TAC 2014)

\[ h(k) = \max_{1 \leq i \leq 3} |x_i(k)| \]
Continuous-time case

Accurate projection: hard to obtain in practice
Approximate projection is a cheap choice

Approximate angle: $0 \leq \theta \leq \theta^* < \pi/2$; modified projection sets

![Diagram showing approximate projection and modified projection sets](image)

$v \quad \theta \quad b(v, K) \quad P_K(v) \quad C_K(v, \theta)$
Main Results

- Connectivity: uniformly jointly-connected for balanced graph in continuous-time case
- Approximate projection + consensus rule + stepsize condition $\rightarrow$ optimal consensus (Automatica 2016)
- **Difference** between continuous and discrete time cases:
  - Definition of approximate projection $\rightarrow$ virtual stepsize based on finite curvature,
  - The intersection set $X_0 = \cap X_i$ may be empty but the aim can be achieved by one algorithm
  - No critical approximate angle (essential difference between continuous-time and discrete-time cases)
3.2 Optimization with constraints

- Constraints: from multiple objectives and condition limitations → analysis and design of distributed optimization algorithms
- Application: smart grids, sensor network, social systems, wireless communication, …

Given constraints: global, local, coupled …
Active constraints: invariance, bounds …
Well known constraints

Constraints for

\[
\min_{x \in \Omega} f(x), \quad f(x) = \sum_{i=1}^{n} f^i(x_i)
\]

1. Local inequality constraint:
   \[g^i_j(x_i) \leq 0,\]

2. Resource allocation:
   \[\sum_{i=1}^{n} x_i = d_0\]

3. Local equality constraints:
   \[A^i x = b^i_{i=1}\]

4. Constraint sets:
   \[x_i \in \Omega_i\]

- Various combinations of constraints: 2+4 & 3+4 with some conditions …
- Applications to sensor networks or smart grids
Remarks

Start with simple cases: Lipschitz of the gradient + undirected graph +

- Strict convexity $\rightarrow$ asymptotical stability
- Strong convexity $\rightarrow$ exponential stability

Extensions with many challenges:

- Convexity $\rightarrow$ non-unique solution, multiple equilibria
- Nonsmooth functions $\rightarrow$ nonsmooth analysis
- Directed time-varying graph $\rightarrow$ auxillary? dynamics to estimate unbalanced weights, analysis based on common bound
- Communication cost $\rightarrow$ quantization, random sleep, event-based
Local inequality constraints

Problem: \( \min f(x), \quad f(x) = \sum_{i=1}^{N} f_i(x) \)
\( g_j^i(x) \leq 0, \quad j = 1, ..., J^i, \quad i = 1, ..., N \)

- Distributed control:

\[
\begin{align*}
\dot{x}_i &= -\nabla f_i(x_i) - \sum_{j \in \mathcal{N}_i} a_{ij} (x_i - x_j) \\
&\quad - \sum_{j \in \mathcal{N}_i} a_{ij} (v_i - v_j) - \sum_{j=1}^{J^i} \lambda_{ij} \nabla g_j^i(x_i) \\
\dot{v}_i &= \sum_{j \in \mathcal{N}_i} a_{ij} (x_i - x_j); \\
\dot{\lambda}_{ij} &= [g_j^i(x_i)]^+_\lambda_{ij}, \quad j = 1, ..., J^i.
\end{align*}
\]

- Convergence based on a hybrid LaSalle invariance principle (SCL 2015).
Resource allocation

Problem:
\[
\min_{x_i \in \mathbb{R}^m, i \in N} \sum_{i \in N} f_i(x_i),
\]
subject to \[
\sum_{i \in N} x_i = \sum_{i \in N} d_i.
\]

- Distributed control:
\[
\begin{align*}
\dot{x}_i &= -\nabla f_i(x_i) + \lambda_i \\
\dot{\lambda}_i &= -\sum_{j \in N_i} (\lambda_i - \lambda_j) - \sum_{j \in N_i} (z_i - z_j) + (d_i - x_i) \\
\dot{z}_i &= \sum_{j \in N_i} (\lambda_i - \lambda_j)
\end{align*}
\]

- Results: convergence; exponential convergence; additional constraint sets (CCC 2015, Automatica 2016)
3.3 Optimization with disturbance

- Stochastic disturbance (noise, package loss ...) discussed in some optimization results
- Modeled deterministic disturbance may be considered when the agents are physical (UAV, robots) and moving in practical environment
- Exact optimization with disturbance rejection: agent dynamics + optimization goal + exogenous disturbance
Basic Distributed Algorithm

Design: optimization + consensus + internal-model-based disturbance rejection

\[ \dot{v}_i = \alpha \beta \sum_{j=1}^{N} a_{ij} (x_i - x_j) \]

\[ \dot{\eta}_i = (I_n \otimes F) \eta_i + (I_n \otimes G) u_i \]

\[ u_i = -\alpha \nabla f_i(x_i) - v_i - \beta \sum_{j=1}^{N} a_{ij} (x_i - x_j) \]

\[ - (I_n \otimes \Psi) \eta_i \]

optimal term

consensus term

internal model term
Main Results

- The exact optimization can be achieved with known disturbance frequency by internal model (Control Theory & Technology, 2014)
- It is also achieved semi-globally with unknown frequency by adaptive internal model (CCC 2014)
- The agent dynamics can be extended to a nonlinear case (IEEE T-Cybernetics, 2015)
- Event-triggered design for both communication and gradient measurement (IET CTA, 2016)
Simulation (5 agents)

Topology and error trajectories

\[
\begin{align*}
    f_1(x) &= (x + 2)^2, \\
    f_2(x) &= (x - 5)^2 \\
    f_3(x) &= x^2 \ln(1 + x^2) + x^2 \\
    f_4(x) &= \frac{x^2}{\sqrt{x^2 + 1}} + x^2, \\
    f_5(x) &= \frac{x^2}{\ln(2 + x^2)}
\end{align*}
\]
3.4 Nonlinear/High-order Agent

Motivation:
- Integration of control and optimization
- Cyber optimization solved by physical systems

Results:
1. Euler–Lagrangian (EL) systems: nonlinear second order systems
2. High order linear systems → special nonlinear systems
Optimization of EL systems

Mechanical systems in the Euler-Lagrange form:

\[ M_i (q_i) \ddot{q}_i + C_i (q_i, \dot{q}_i) \dot{q}_i = \tau_i. \]

\( N \) heterogeneous agents with uncertain parameters

Distributed optimization control design for EL systems:

\[ \tau_i = - kq_i - \alpha \nabla f_i (q_i) - k \sum_{j \in N_i} a_{ij} (q_i - q_j) - kv_i \]

\[ \dot{v}_i = \sum_{j \in N_i} a_{ij} (q_i - q_j) \]

Task: tracking, formation, coverage

Constraint: obstacle, energy, resource ...
Results for EL systems

- Basic assumptions: Strong convexity & Undirected graph
- **Result 1 (Unmanned Systems 2016):** Lipschitz of gradient $\rightarrow$ semi global convergence (exponential).
- **Result 2 (Automatica 2017):** Global Lipschitz of gradient $\rightarrow$ global convergence (exponential) (optimization of double integrator + tracking control of EL systems)
- **Result 3 (Kybernetika 2017):** Event-triggered optimization design for EL systems
- **Result 4 (IFAC conference 2016):** Optimization design with kinematic constraints (saturation of velocity and acceleration)
For agents in the form of $n$-th integrator: $x_i^{(n-1)} = v_i$.

The algorithm for each agent:

$$v_i = - \sum_{i=1}^{n-1} k_{n-i} x_i^{(i)} - \beta \sum_{j \in N_i} a_{ij} (x_i - x_j) - \alpha \nabla f_i(x_i) - w_i$$

$$\dot{w}_i = \alpha \beta \sum_{j \in N_i} a_{ij} \left( (x_i - x_j) + \sum_{i=1}^{n-1} (x_i^{(i)} - x_j^{(i)}) \right),$$

Results: strong convexity + undirected graph + global Lipschitz $\Rightarrow$ exponential convergence

$\Rightarrow$ minimum phase nonlinear systems and observer-based output feedback design
4. Conclusions

- **Distributed optimization**: optimization algorithms based on local information $\rightarrow$ scalability, reliability, and maybe security …
- **Challenges**: operations research + control systems + complex network + computational complexity + …
- **Applications**: estimation (sensor), simultaneous routing & resource allocation (wireless communication), opinion dynamics (social networks), intersection computation (computer), ……
## Research framework

<table>
<thead>
<tr>
<th>Physical</th>
<th>constraint</th>
<th>uncertainty</th>
<th>dynamics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>Non-holonomic, saturation, event-based ...</td>
<td>Identification, adaptive con., robust con.</td>
<td>Stochastic, time-varying, nonlinear ...</td>
</tr>
<tr>
<td>Optimiz.</td>
<td>Inequality, bounded set, equality ...</td>
<td>Data-based, online regret, robust opti.</td>
<td>High order, multi-scale, ...</td>
</tr>
<tr>
<td>Network</td>
<td>Communication, environment, energy ...</td>
<td>Survivability, security, failure ...</td>
<td>Link dynamics, split/merge, switching ...</td>
</tr>
</tbody>
</table>
Some recent results for MAS

- Containment control & multiple leaders (Automatica 2014)
- Distributed output regulation (IEEE TAC 2013, 2014, 2016; Automatica 2015; IJRNC 2013): Internal model based design
- Attitude synchronization and formation (Automatica 2014)
- Coverage: cooperative sweeping (Automatica 2013)
- Distributed Kalman filter (IEEE TAC 2013)
- Quantization in control and optimization (IEEE TCNS 2014, IEEE TAC 2016)
- Target surrounding (IEEE TAC 2015)
- Opinion dynamics (Physica A 2013, Automatica 2016)

... ...
Thank you!