

Optimization Algorithm Design via Electric Circuits

Tetiana Parshakova
Center for Computational Mathematics
Flatiron Institute

Alg-ML Seminar, Princeton

10/14/2025

Motivation

- classical optimization algorithm design: based on worst-case guarantees
 - provably convergent but often conservative and slow in practice
- modern ML optimizers: tuned for fast empirical performance
 - often lack convergence guarantees and may not converge under convexity

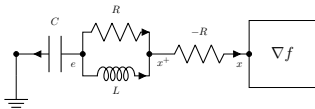
goal

- principled way to design algorithms that are fast and provably convergent

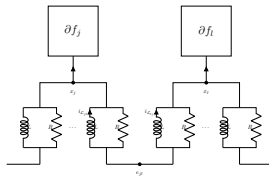
Principled optimization algorithm design

- reframe optimization algorithms as circuit diagrams: RLC circuits composed with nonlinear resistor ∇f
- replace heavy differential relations with modular diagrams governed by local rules
 - similar to how Feynman diagrams make computation in QFT easier
- convergent by construction due to energy dissipation
- equilibrium state representing the primal-dual solution

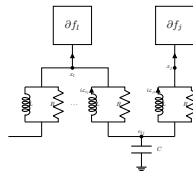
Nesterov Acceleration



Decentralized ADMM



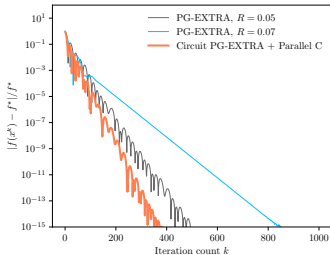
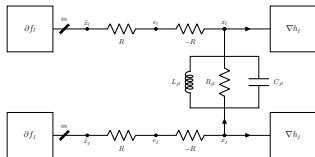
DADMM+C (new!)



Principled optimization algorithm design

- vary RLC circuits to create new algorithms with convergence guarantees
- use proof preserving automatic discretization to get implementable algorithm

PG-EXTRA + Parallel C (new!)



S. Boyd, T. Parshakova, E. Ryu, and J. Suh. Optimization algorithm design via electric circuits. *NeurIPS Spotlight*, 2024.

Outline

Continuous-time optimization with circuits

Circuits for classical algorithms

Automatic discretization

Distributed convex optimization problem

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in \mathcal{R}(E^\top) \quad (\text{i.e., } x = E^\top z \text{ for some } z) \end{array}$$

- $f: \mathbf{R}^m \rightarrow \mathbf{R} \cup \{\infty\}$ is closed, convex, and proper
- n nets N_1, \dots, N_n forming a partition of $\{1, \dots, m\}$
- $E \in \mathbf{R}^{n \times m}$ is a selection matrix

$$E_{ij} = \begin{cases} +1 & \text{if } j \in N_i \\ 0 & \text{otherwise} \end{cases}$$

$x \in \mathcal{R}(E^\top)$ ensures consistency among components: $x_j = x_{j'}$ if $j, j' \in N_i$

Simple example

$$\begin{array}{ll}
 \text{minimize} & f(x) \\
 \text{subject to} & x_1 = x_3 \\
 & x_2 = x_4
 \end{array}
 \Leftrightarrow
 \begin{array}{ll}
 \text{minimize} & f(x) \\
 \text{subject to} & \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_1 \\ z_2 \\ z_3 \end{bmatrix}
 \end{array}
 \Leftrightarrow
 \begin{array}{ll}
 \text{minimize} & f(x) \\
 \text{subject to} & x \in \mathcal{R}(E^\top)
 \end{array}$$

- $N_1 = \{1, 3\}$, $N_2 = \{2, 4\}$, $N_3 = \{5\}$, and $E^\top = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Example: Unconstrained problem

$$\begin{array}{lll} \underset{x \in \mathbf{R}^m}{\text{minimize}} & f(x) & \Leftrightarrow \underset{x, z \in \mathbf{R}^m}{\text{minimize}} \quad f(x) \\ & & \text{subject to} \quad x = z \end{array} \quad \Leftrightarrow \quad \begin{array}{ll} \underset{x \in \mathbf{R}^m}{\text{minimize}} & f(x) \\ \text{subject to} & x \in \mathcal{R}(E^\top) \end{array}$$

- $x \in \mathbf{R}^m$ is the decision variable
- $N_i = \{i\}$ for all $i = 1, \dots, m$
- $E^\top = I \in \mathbf{R}^{m \times m}$

Example: Consensus problem

$$\begin{array}{ll} \underset{x_1, \dots, x_N \in \mathbf{R}^{m/N}}{\text{minimize}} & \sum_{i=1}^N f_i(x_i) \\ \text{subject to} & x_1 = \dots = x_N \end{array} \quad \Leftrightarrow \quad \begin{array}{ll} \underset{x_1, \dots, x_N \in \mathbf{R}^{m/N}, z \in \mathbf{R}^{m/N}}{\text{minimize}} & \sum_{i=1}^N f_i(x_i) \\ \text{subject to} & \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} z \\ \vdots \\ z \end{bmatrix} \end{array}$$

$$\Leftrightarrow \begin{array}{ll} \underset{x \in \mathbf{R}^m}{\text{minimize}} & f(x) \\ \text{subject to} & x \in \mathcal{R}(E^\top) \end{array}$$

- $f(x) = f_1(x_1) + \dots + f_N(x_N)$ is block-separable, and $E^\top = \begin{bmatrix} I \\ \vdots \\ I \end{bmatrix} \in \mathbf{R}^{m \times m/N}$

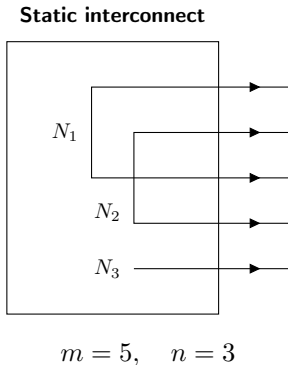
Our goal

$$\begin{array}{ll}\text{minimize} & f(x) \\ \text{subject to} & x \in \mathcal{R}(E^\top)\end{array}$$

our goal is to find a primal-dual solution satisfying the KKT conditions

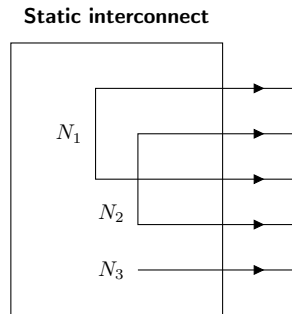
$$y \in \partial f(x), \quad x \in \mathcal{R}(E^\top), \quad y \in \mathcal{N}(E)$$

Static interconnect



- static interconnect is a set of (ideal) wires connecting m terminals and forming n nets
- $x \in \mathbf{R}^m$ is terminal potentials, and $y \in \mathbf{R}^m$ is vector of currents leaving the terminals
- Kirchhoff's voltage law (KVL) is $x \in \mathcal{R}(E^\top)$
- Kirchhoff's current law (KCL) is $y \in \mathcal{N}(E)$
- static interconnect enforces the V-I relationship $(x, y) \in \mathcal{R}(E^\top) \times \mathcal{N}(E)$

Static interconnect for simple example



$$N_1 = \{1, 3\}, N_2 = \{2, 4\}, \\ N_3 = \{5\}$$

$$E^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

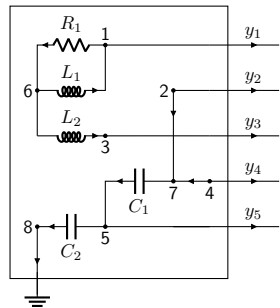
$$x \in \mathcal{R}(E^T) \Leftrightarrow x_1 = x_3, x_2 = x_4$$

$$y \in \mathcal{N}(E) \Leftrightarrow y_1 + y_3 = 0, y_2 + y_4 = 0, y_5 = 0$$

Dynamic interconnect

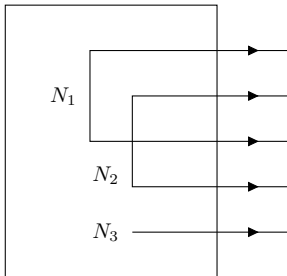
- dynamic interconnect is an RLC circuit with m terminals and 1 ground node

Dynamic interconnect

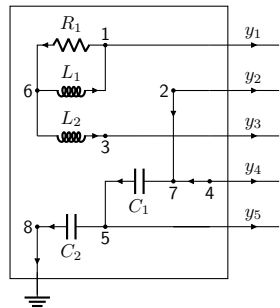


Admissible dynamic interconnect

Static interconnect



Dynamic interconnect



- dynamic interconnect is admissible if it relaxes to the static interconnect at equilibrium

$$\left\{ (x, y) \mid Ai = \begin{bmatrix} -y \\ 0 \end{bmatrix}, v = A^T \begin{bmatrix} x \\ e \end{bmatrix}, v_{\mathcal{R}} = D_{\mathcal{R}} i_{\mathcal{R}}, v_{\mathcal{L}} = 0, i_{\mathcal{C}} = 0 \right\} = \mathcal{R}(E^T) \times \mathcal{N}(E)$$

Composing static interconnect with ∂f

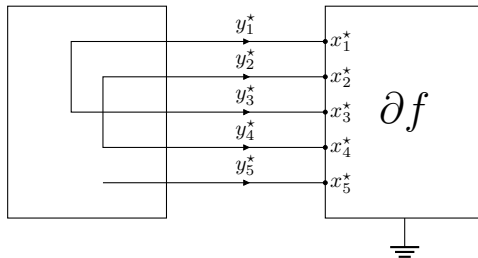
- ∂f is interpreted as m -terminal grounded electric device
- ∂f enforces the V-I relation
 $y \in \partial f(x)$

$$y \in \partial f(x) \quad (\text{nonlinear resistor})$$

$$x \in \mathcal{R}(E^\top) \quad (\text{KVL})$$

$$y \in \mathcal{N}(E) \quad (\text{KCL})$$

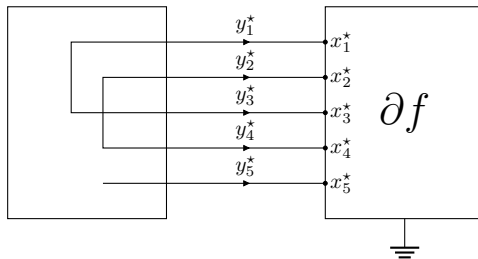
Static interconnect



Composing static interconnect with ∂f

$$\begin{aligned} y &\in \partial f(x) && \text{(stationarity)} \\ x &\in \mathcal{R}(E^\top) && \text{(primal feasibility)} \\ y &\in \mathcal{N}(E) && \text{(dual feasibility)} \end{aligned}$$

Static interconnect



Composing dynamic interconnect with ∂f

$$y(t) \in \partial f(x(t)) \quad (\text{nonlinear resistor})$$

$$v(t) = A^T \begin{bmatrix} x(t) \\ e(t) \end{bmatrix} \quad (\text{KVL})$$

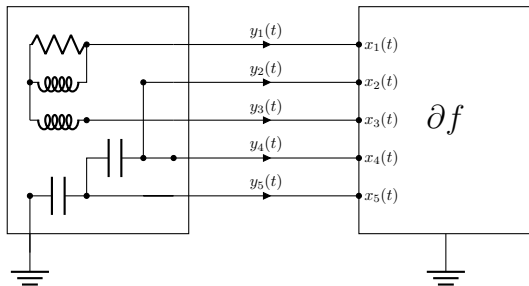
$$Ai(t) = \begin{bmatrix} -y(t) \\ 0 \end{bmatrix} \quad (\text{KCL})$$

$$v_{\mathcal{R}}(t) = D_{\mathcal{R}} i_{\mathcal{R}}(t) \quad (\text{resistor})$$

$$v_{\mathcal{L}}(t) = D_{\mathcal{L}} \frac{d}{dt} i_{\mathcal{L}}(t) \quad (\text{inductor})$$

$$i_{\mathcal{C}}(t) = D_{\mathcal{C}} \frac{d}{dt} v_{\mathcal{C}}(t) \quad (\text{capacitor})$$

Dynamic interconnect



Energy dissipation

- (v^*, i^*, x^*, y^*) is equilibrium of admissible dynamic interconnect composed with ∂f
- define the energy of the circuit at time t as

$$\mathcal{E}(t) = \frac{1}{2} \|v_{\mathcal{C}}(t) - v_{\mathcal{C}}^*\|_{D_{\mathcal{C}}}^2 + \frac{1}{2} \|i_{\mathcal{L}}(t) - i_{\mathcal{L}}^*\|_{D_{\mathcal{L}}}^2$$

- convexity of f is incremental passivity of ∂f

$$\langle x - x', y - y' \rangle \geq 0, \quad y \in \partial f(x), \quad y' \in \partial f(x')$$

- energy is a dissipative (non-increasing) quantity

$$\frac{d}{dt} \mathcal{E} = \langle v_{\mathcal{C}} - v_{\mathcal{C}}^*, i_{\mathcal{C}} - i_{\mathcal{C}}^* \rangle + \langle i_{\mathcal{L}} - i_{\mathcal{L}}^*, v_{\mathcal{L}} - v_{\mathcal{L}}^* \rangle = -\|i_{\mathcal{R}}\|_{D_{\mathcal{R}}}^2 - \underbrace{\langle x - x^*, y - y^* \rangle}_{\geq 0} \leq 0$$

Energy dissipation

Theorem

Assume $f: \mathbf{R}^m \rightarrow \mathbf{R}$ is strongly convex and smooth. Assume the dynamic interconnect is admissible, and let (x^, y^*) be a primal-dual solution pair. Let $(v(t), i(t), x(t), y(t))$ be a curve satisfying dynamic interconnect differential algebraic inclusion. Then,*

$$\lim_{t \rightarrow \infty} (x(t), y(t)) = (x^*, y^*).$$

Outline

Continuous-time optimization with circuits

Circuits for classical algorithms

Automatic discretization

Moreau envelope

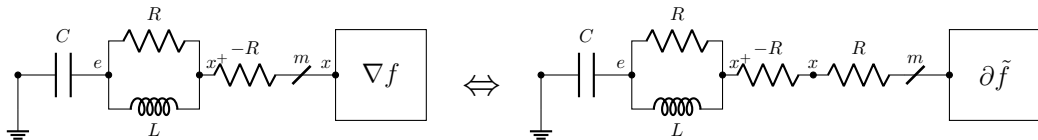
- for $R > 0$, the Moreau envelope of $f: \mathbf{R}^m \rightarrow \mathbf{R} \cup \{\infty\}$ with parameter R is

$${}^R f(x) = \inf_{z \in \mathbf{R}^m} \left(f(z) + \frac{1}{2R} \|z - x\|_2^2 \right)$$

- ∂f composed with a resistor is equivalent to $\nabla {}^R f(x)$, i.e., same V-I relation on the m pins of x



Nesterov acceleration



- $f: \mathbf{R}^m \rightarrow \mathbf{R}$ is a $1/R$ -smooth convex function

- V-I relations

$$y = \nabla f(x), \quad \frac{x^+ - x}{-R} = \nabla f(x), \quad v_C = e,$$

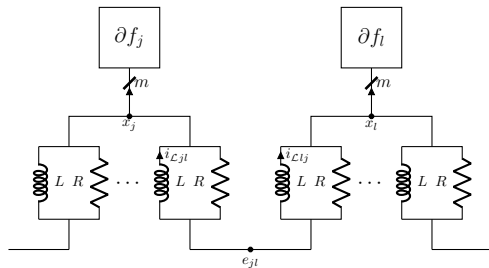
$$\frac{d}{dt} i_L = \frac{1}{L} (v_C - x^+), \quad \frac{d}{dt} v_C = -\frac{1}{C} \nabla f(x)$$

- high-resolution ODE of Nesterov acceleration

$$\frac{d^2}{dt^2} x + \frac{R}{L} \frac{d}{dt} x + \left(\frac{1}{C} - \frac{R^2}{L} \right) \frac{d}{dt} \nabla f(x) + \frac{R}{LC} \nabla f(x) = 0$$

Decentralized ADMM

- $f_1, \dots, f_N: \mathbf{R}^m \rightarrow \mathbf{R} \cup \{\infty\}$ be CCP functions
- communication graph G over computation nodes
- for every edge (j, l) in graph G the circuit is on the right



Decentralized ADMM

- V-I relations for $j \in \Gamma_l$

$$x_j = \mathbf{prox}_{(R/|\Gamma_j|)f_j} \left(\frac{1}{|\Gamma_j|} \sum_{l \in \Gamma_j} (R i_{\mathcal{L}_{jl}} + e_{jl}) \right)$$

$$e_{jl} = \frac{1}{2}(x_j + x_l)$$

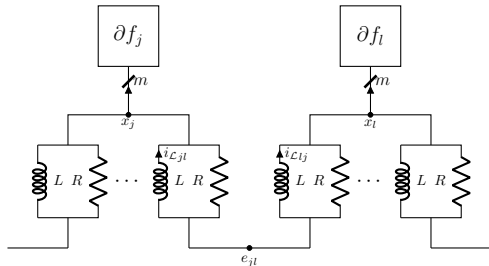
$$\frac{d}{dt} i_{\mathcal{L}_{jl}} = \frac{1}{L}(e_{jl} - x_j)$$

- L/R stepsize Euler discretization

$$x_j^{k+1} = \mathbf{prox}_{(R/|\Gamma_j|)f_j} \left(\frac{1}{|\Gamma_j|} \sum_{l \in \Gamma_j} (R i_{\mathcal{L}_{jl}}^k + e_{jl}^k) \right)$$

$$e_{jl}^{k+1} = \frac{1}{2}(x_j^{k+1} + x_l^{k+1})$$

$$i_{\mathcal{L}_{jl}}^{k+1} = i_{\mathcal{L}_{jl}}^k + \frac{1}{R}(e_{jl}^{k+1} - x_j^{k+1})$$

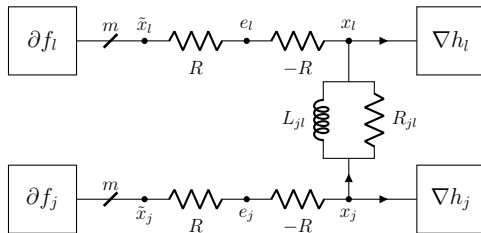


PG-EXTRA

- $f_1, \dots, f_N: \mathbf{R}^m \rightarrow \mathbf{R} \cup \{\infty\}$ are CCP functions and $h_1, \dots, h_N: \mathbf{R}^m \rightarrow \mathbf{R}$ are convex M -smooth functions

- define mixing matrix $W \in \mathbf{R}^{N \times N}$

$$W_{jl} = \begin{cases} 1 - \sum_{l \in \Gamma_j} \frac{R}{R_{jl}} & \text{if } j = l \\ \frac{R}{R_{jl}} & \text{if } j \neq l, \\ 0 & \text{otherwise} \end{cases} \quad l \in \Gamma_j$$



PG-EXTRA

- V-I relations for $j \in \Gamma_l$

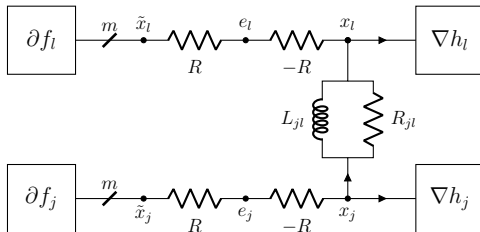
$$x_j = \mathbf{prox}_{Rf_j} \left(\sum_{l=1}^N W_{jl} x_l - R \nabla h_j(x_j) - w_j \right)$$

$$\frac{d}{dt} w_j = x_j - \sum_{l=1}^N W_{jl} x_l$$

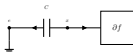
- 1/2 stepsize Euler discretization

$$x^{k+1} = \mathbf{prox}_{Rf} \left(W x^k - R \nabla h(x^k) - w^k \right)$$

$$w^{k+1} = w^k + \frac{1}{2} (I - W) x^k$$



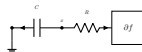
Gradient Descent



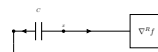
Nesterov Acceleration



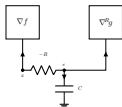
Proximal Point Method



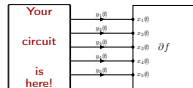
Proximal (Moreau)



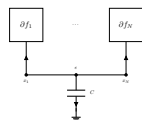
Prox-Grad Method



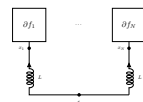
Your Algorithm



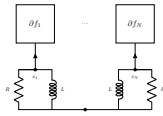
Primal Decomposition



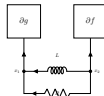
Dual Decomposition



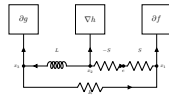
Prox Decomposition



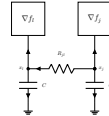
Douglas-Rachford Splitting



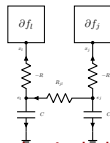
Davis-Yin Splitting



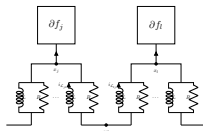
Decentralized GD



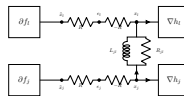
Adapt-then-Combine



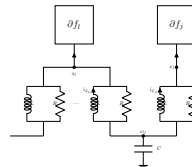
Decentralized ADMM



PG-EXTRA



DADMM+C (new!)



Circuits for classical algorithms

Outline

Continuous-time optimization with circuits

Circuits for classical algorithms

Automatic discretization

Challenges

not every dynamics discretization leads to a convergent algorithm

- in numerical analysis we focus on convergence of x^k to the solution trajectory $x(kh)$ as $h \rightarrow 0$
- in optimization we are interested in convergence

$$\lim_{k \rightarrow \infty} x^k = x^*, \quad \lim_{k \rightarrow \infty} f(x^k) = f(x^*)$$

we provide a novel discretization methodology

- that *preserves the proof structure*
- is automatic, based on computer-assisted approach

Sufficiently dissipative discretization

- discretize the dynamics of an admissible dynamic interconnect using a two-stage Runge–Kutta method with parameters α, β and stepsize $h > 0$
- $\{(v^k, i^k, x^k, y^k)\}_{k=1}^{\infty}$ the iterates of discretized algorithm

$$(v^{k+1}, i^{k+1}, x^{k+1}, y^{k+1}) = T_{f, \alpha, \beta, h}(v^k, i^k, x^k, y^k)$$

- energy stored in the circuit at time $t = kh$ is

$$\mathcal{E}_k = \frac{1}{2} \|v_{\mathcal{C}}^k - v_{\mathcal{C}}^{\star}\|_{D_{\mathcal{C}}}^2 + \frac{1}{2} \|i_{\mathcal{L}}^k - i_{\mathcal{L}}^{\star}\|_{D_{\mathcal{L}}}^2$$

- discretization is *sufficiently dissipative* if there is an $\eta > 0$ such that for all $k = 1, 2, \dots$

$$\mathcal{E}_{k+1} - \mathcal{E}_k + \underbrace{\eta \langle x^k - x^{\star}, y^k - y^{\star} \rangle}_{\geq 0} \leq 0$$

Sufficiently dissipative discretization

Lemma

Assume $f: \mathbf{R}^m \rightarrow \mathbf{R} \cup \{\infty\}$ is a strictly convex function and the dynamic interconnect is admissible. If the two-stage Runge–Kutta discretization generates a discrete-time sequence $\{(v^k, i^k, x^k, y^k)\}_{k=1}^{\infty}$ that is sufficiently dissipative, then

$$\lim_{k \rightarrow \infty} x^k = x^*.$$

Automatic discretization

- discretization, characterized by (α, β, h) , is dissipative for a given $\eta > 0$ if the objective of the worst-case optimization problem is non-positive

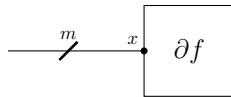
$$\begin{array}{ll}\text{maximize} & \mathcal{E}_2 - \mathcal{E}_1 + \eta \langle x^1 - x^*, y^1 - y^* \rangle \\ \text{subject to} & \mathcal{E}_1 = \frac{1}{2} \|v_{\mathcal{C}}^1 - v_{\mathcal{C}}^*\|_{D_{\mathcal{C}}}^2 + \frac{1}{2} \|i_{\mathcal{L}}^1 - i_{\mathcal{L}}^*\|_{D_{\mathcal{L}}}^2 \\ & \mathcal{E}_2 = \frac{1}{2} \|v_{\mathcal{C}}^2 - v_{\mathcal{C}}^*\|_{D_{\mathcal{C}}}^2 + \frac{1}{2} \|i_{\mathcal{L}}^2 - i_{\mathcal{L}}^*\|_{D_{\mathcal{L}}}^2 \\ & (v^2, i^2, x^2, y^2) = T_{f, \alpha, \beta, h}(v^1, i^1, x^1, y^1) \\ & (v^1, i^1, x^1, y^1) \text{ feasible initial point} \\ & (v^*, i^*, x^*, y^*) \text{ values at equilibrium} \\ & f \in \mathcal{F}\end{array}$$

- $f, v^1, i^1, x^1, y^1, v^*, i^*, x^*, y^*$ are the decision variables
 - \mathcal{F} is a family of functions (e.g., L -smooth convex)
- can be solved as finite dimensional semidefinite program (SDP) using ideas from performance estimation problem (PEP)

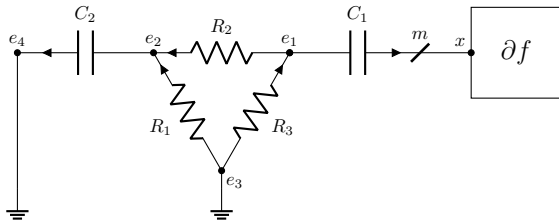
Package ciropt

step 1: create the static interconnect representing the optimality conditions of your problem

minimize $f(x)$



step 2: design your algorithm: design RLC circuit that relaxes to the static interconnect in equilibrium



Package ciropt

step 3: write the V-I relations: a convergent dynamics by the construction

$$\left. \begin{aligned} x &= \mathbf{prox}_{(R/2)f}(z) \\ y &= \frac{2}{R}(z - x) \\ \frac{d}{dt}e_2 &= -\frac{1}{2CR}(Ry + 3e_2) \\ \frac{d}{dt}z &= -\frac{1}{4CR}(5Ry + 3e_2) \end{aligned} \right| \begin{aligned} x^k &= \mathbf{prox}_{(R/2)f}(z^k) \\ y^k &= \frac{2}{R}(z^k - x^k) \\ e_2^{k+1} &= e_2^k - \frac{h}{2CR}(Ry^k + 3e_2^k) \\ z^{k+1} &= z^k - \frac{h}{4CR}(5Ry^k + 3e_2^k). \end{aligned}$$

Package ciropt

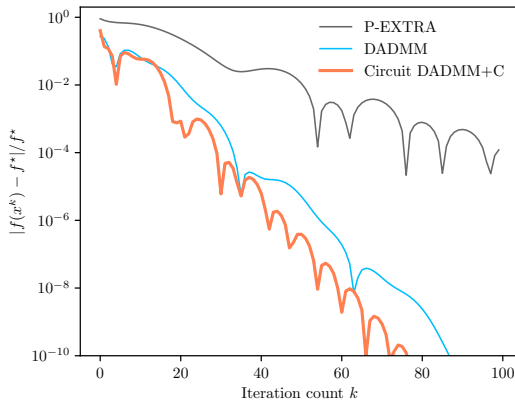
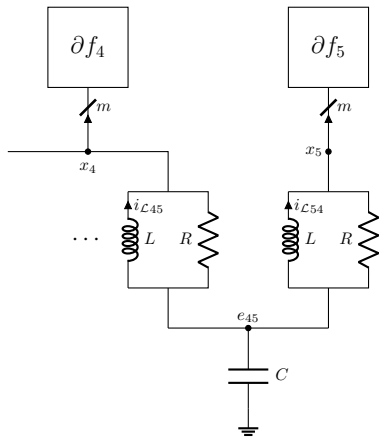
step 4: leverage our PEP-based automatic discretization package ciropt and obtain algorithm

```
import ciropt as co
problem = co.CircuitOpt()
f = co.def_function(problem, mu=0, M=np.inf)
...
In [2]: params["eta"], params["h"]
Out[2]: (6.66, 6.66)
```

step 5: your algorithm is ready to use!

$$\begin{aligned}x^k &= \mathbf{prox}_{(1/2)f}(z^k) \\ y^k &= 2(z^k - x^k) \\ w^{k+1} &= w^k - 0.33(y^k + 3w^k) \\ z^{k+1} &= z^k - 0.16(5y^k + 3w^k)\end{aligned}$$

Numerical results: DADMM+C



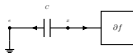
Conclusions

- introduce a framework for designing optimization algorithms via RLC circuits
 - convergent due to energy dissipation
- electric circuits for standard methods
 - Nesterov acceleration, proximal point method, prox-gradient, primal decomposition, dual decomposition, DYS, DRS, decentralized gradient descent, diffusion, DADMM and PG-EXTRA
- PEP-based automated discretization that preserves proof structure
 - https://github.com/cvxgrp/optimization_via_circuits

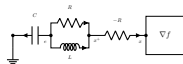
Future directions

- extending the framework for stochastic programming
- extracting convergence rates
 - energy dissipation over multiple steps
- include methods with time dependent step sizes
 - using time dependent electric components
- key question: how to automate the development of accelerated algorithms?
 - search over admissible circuit designs
 - find circuits which result in fast relaxation, e.g., critical damping
 - guidance on how to design fast optimization method from circuit architecture

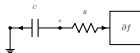
Gradient Descent



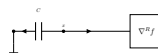
Nesterov Acceleration



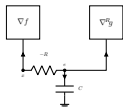
Proximal Point Method



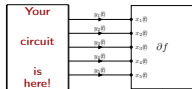
Proximal (Moreau)



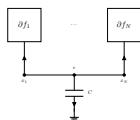
Prox-Grad Method



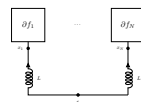
Your Algorithm



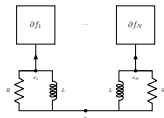
Primal Decomposition



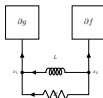
Dual Decomposition



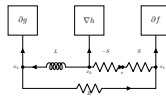
Prox Decomposition



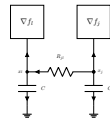
Douglas–Rachford Splitting



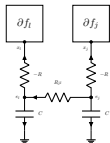
Davis–Yin Splitting



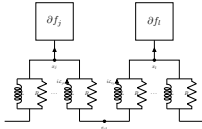
Decentralized GD



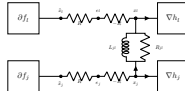
Adapt-then-Combine



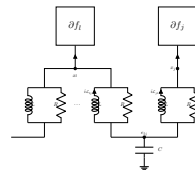
Decentralized ADMM



PG-EXTRA



DADMM+C (new!)



$O(1/K)$ convergence rate

- Lagrangian $L(x, z, y) = f(x) + y^T(x - Ez)$
- by dissipativity,

$$\begin{aligned} 0 \leq L(x^k, z^*, y^*) - f(x^*) &\leq \underbrace{-\langle y^k, x^* - x^k \rangle}_{\geq f(x^k) - f(x^*)} + \langle y^*, x^k - \underbrace{Ez^*}_{=x^*} \rangle \\ &= \langle y^k - y^*, x^k - x^* \rangle \leq \frac{1}{\eta}(\mathcal{E}_k - \mathcal{E}_{k+1}) \end{aligned}$$

- by summability,

$$\min_{k \in \{0, \dots, K\}} L(x^k, z^*, y^*) - f(x^*) \leq \frac{1}{\eta(K+1)} \mathcal{E}_0 = O\left(\frac{1}{K}\right)$$

Automatic discretization

- capacitor and inductor ODEs are of the form

$$\frac{d}{dt}x(t) = F(x(t))$$

- two-stage Runge–Kutta method for discretization, with coefficients α , β , and stepsize h

$$\begin{aligned}x^{k+1/2} &= x^k + \alpha h F(x^k) \\x^{k+1} &= x^k + \beta h F(x^k) + (1 - \beta) h F(x^{k+1/2})\end{aligned}$$

- interpolation lemma states $f_l \in \mathcal{F}_{\mu_l, M_l}(\mathbf{R}^n)$ if and only if

$$\begin{aligned}I_K &= \{1, 1.5, 2, \star\} \\0 &\geq f_l^j - f_l^i + \langle g_l^j, x^i - x^j \rangle + \frac{1}{2M_l} \|g_l^i - g_l^j\|_2^2 \\&\quad + \frac{\mu_l}{2(1 - \mu_l/M_l)} \|x^i - x^j - 1/M_l(g_l^i - g_l^j)\|_2^2, \quad i, j \in I_K\end{aligned}$$

Automatic discretization

- decentralized optimization problem

$$\begin{array}{ll}\text{minimize} & f_1(x_1) + \cdots + f_N(x_N) \\ \text{subject to} & x_1, \dots, x_N \in \mathbf{R}^{m/N} \\ & x_j = x_l, \quad j = 1, \dots, N, \quad l \in \Gamma_j\end{array}$$

- computer-assisted discretization

$$\begin{array}{ll}\text{maximize} & \eta \\ \text{subject to} & \mathcal{E}_2 - \mathcal{E}_1 + \eta \langle x^1 - x^*, y^1 - y^* \rangle \leq 0 \\ & \mathcal{E}_1 = \frac{1}{2} \|v_C^1 - v_C^*\|_{D_C}^2 + \frac{1}{2} \|i_{\mathcal{L}}^1 - i_{\mathcal{L}}^*\|_{D_{\mathcal{L}}}^2 \\ & \mathcal{E}_2 = \frac{1}{2} \|v_C^2 - v_C^*\|_{D_C}^2 + \frac{1}{2} \|i_{\mathcal{L}}^2 - i_{\mathcal{L}}^*\|_{D_{\mathcal{L}}}^2 \\ & (v^2, i^2, x^2, y^2) = T_{f, \alpha, \beta, h}(v^1, i^1, x^1, y^1) \\ & (v^1, i^1, x^1, y^1) \text{ feasible initial point} \\ & (v^*, i^*, x^*, y^*) \text{ values at equilibrium} \\ & f_l \in \mathcal{F}_{\mu_l, M_l}(\mathbf{R}^n) \\ & \alpha, \beta \geq 0, \quad h, \eta > 0\end{array}$$