Theory of Functional Connections applied to Nonlinear Programming subject to Equality Constraints

Tina Mai[1,*], Daniele Mortari[2]

[1]Duy Tan University (DTU), Da Nang, Viet Nam

[2]Texas A&M University (TAMU), College Station, Texas, USA

The Fourth German–Russian Workshop on Numerical Methods and Mathematical Modelling in Geophysical and Biomedical Sciences

Far Eastern Federal University campus

Island Russky, Vladivostok

October 7-11, 2019

Outline

- Motivations and background on the TFC
  - Univariate and multivariate
  - Applications on ODEs
- QP subject to equality constraints
  - Approach #1
  - Equivalent reduced equality system
  - Approach #2
  - Accuracy and speed tests
- NLP subject to equality constraints
  - The 2nd order approach
  - Convergence analysis
    - Quadratic convergence
    - Bounded number of iterations
- Conclusions

Motivations: Optimal control

Indirect Method: Apply Pontryagin Minimum Principle (PMP) to derive the necessary conditions and solve a TPBVP (generally not well-posed)

Direct Method: Transform a continuous problem into a finite NLP problems and find the minimum (Convergence to a global minimum generally non-guaranteed)

Theory of functional connections

Formal constrained expression

$$\boxed{y(x) = g(x) + \sum_{k=1}^{n} \eta_k \, p_k(x)}$$

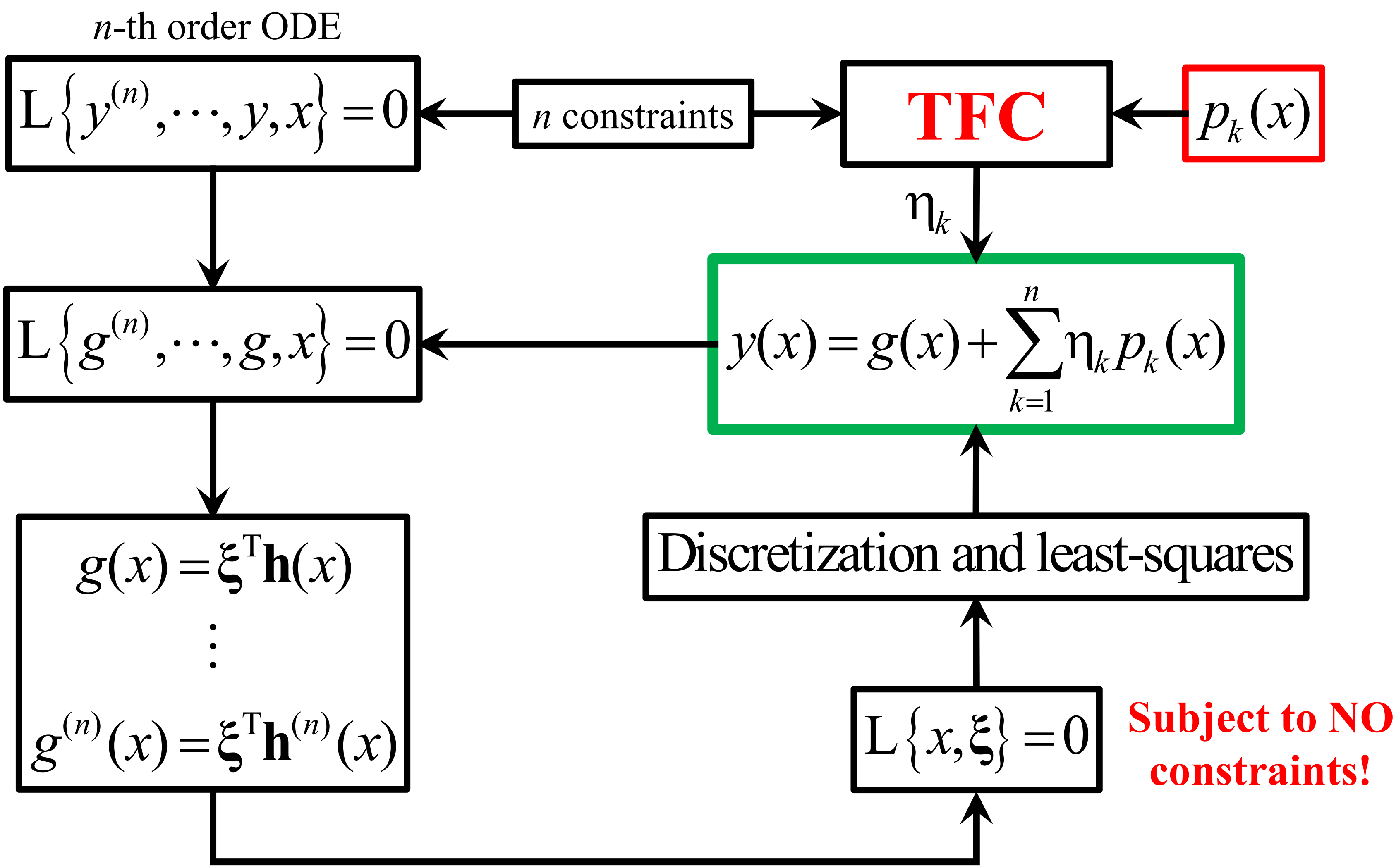
$$\begin{cases} p_k(x) & \text{are } n \text{ assigned functions} \\ \eta_k & \text{are coefficient functions} \\ g(x) & \text{is a } \textbf{free} \text{ function} \end{cases}$$

Four constraints example:

$$\left.\frac{\mathrm{d}^2 y}{\mathrm{d}t^2}\right|_{-1} = \ddot{y}_{-1}, \qquad y(0) = y_0, \qquad y(2) = y_2, \qquad \text{and} \qquad \left.\frac{\mathrm{d}y}{\mathrm{d}t}\right|_{2} = \dot{y}_2$$

$$y(x) = g(x) + \frac{-4x + 4x^2 - x^3}{14}(\ddot{y}_{-1} - \ddot{g}_{-1}) + \frac{28 - 24x + 3x^2 + x^3}{28}(y_0 - g_0) +$$
$$+ \frac{24x - 3x^2 - x^3}{28}(y_2 - g_2) + \frac{-10x + 3x^2 + x^3}{14}(\dot{y}_2 - \dot{g}_2)$$

How to use TFC to solve ODEs



ODEs: features summary

1. Approximate analytical solution → Analysis (e.g., derivative, integral, etc.)
2. Unification → IVP, BVP, MVP
3. Robustness → Very low condition number
4. Speed →~ msec → real-time applications
5. Accuracy → machine error level
6. Constraints
   1. Constraint range and integration range are completely independent.
   2. Constraint types: absolute, relative, component, linear, and integral. (Coming: **infinite and inequality**)

TFC in n-dimensions

$$f(\mathbf{x}) = \underbrace{\mathrm{M}\,(c(\mathbf{x}))_{i_1 i_2 \ldots i_n}\,\mathbf{v}_{i_1}\mathbf{v}_{i_2}\ldots\mathbf{v}_{i_n}}_{A(\mathbf{x})} + \underbrace{g(\mathbf{x}) - \mathrm{M}\,(g(\mathbf{x}))_{i_1 i_2 \ldots i_n}\,\mathbf{v}_{i_1}\mathbf{v}_{i_2}\ldots\mathbf{v}_{i_n}}_{B(\mathbf{x})}$$

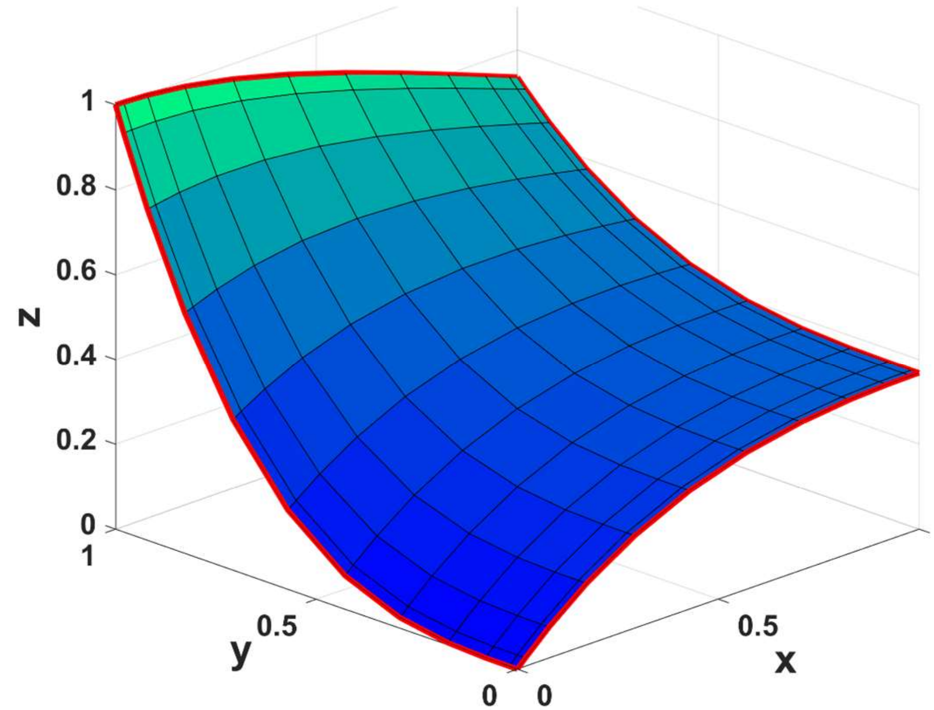
$$\mathbf{v}_k = \left\{1,\ \ \sum_{i=1}^{\ell_k}\alpha_{i1}h_i(x_k),\ \ \sum_{i=1}^{\ell_k}\alpha_{i2}h_i(x_k),\ \ \ldots,\ \ \sum_{i=1}^{\ell_k}\alpha_{i\ell_k}h_i(x_k)\right\}$$

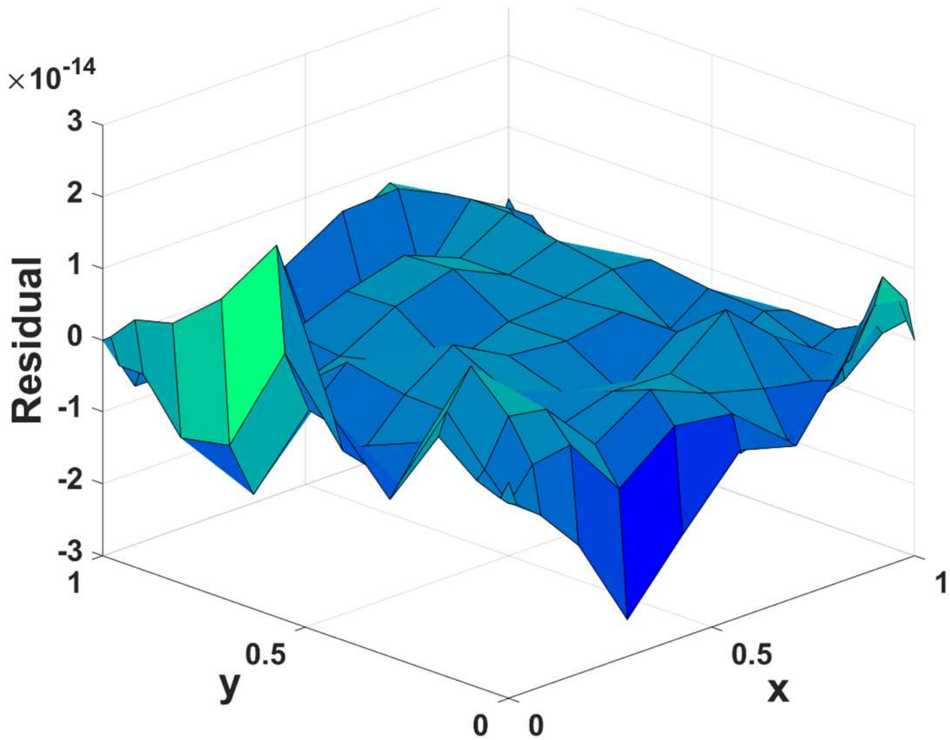
where the ℓ_k functions $h_i(x_k)$ must be linearly independent

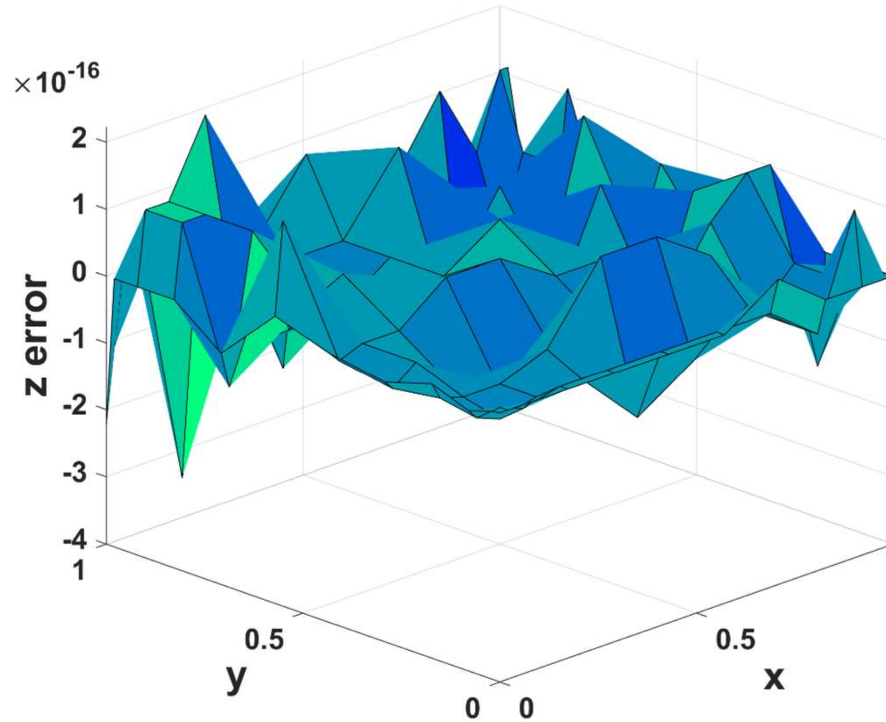
$$\begin{bmatrix} {}^k b_{p_1}^{d_1}[h_1] & {}^k b_{p_1}^{d_1}[h_2] & \cdots & {}^k b_{p_1}^{d_1}[h_{\ell_k}] \\ {}^k b_{p_2}^{d_2}[h_1] & {}^k b_{p_2}^{d_2}[h_2] & \cdots & {}^k b_{p_2}^{d_2}[h_{\ell_k}] \\ \vdots & \vdots & \ddots & \vdots \\ {}^k b_{p_{\ell_k}}^{d_{\ell_k}}[h_1] & {}^k b_{p_{\ell_k}}^{d_{\ell_k}}[h_2] & \cdots & {}^k b_{p_{\ell_k}}^{d_{\ell_k}}[h_{\ell_k}] \end{bmatrix} \begin{bmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1\ell_k} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2\ell_k} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{\ell_k 1} & \alpha_{\ell_k 2} & \cdots & \alpha_{\ell_k \ell_k} \end{bmatrix} = I_{\ell_k \times \ell_k}$$

$$\nabla^2 z(x,y) = e^{-x}(x - 2 + y^3 + 6y)$$

$$\text{subject to:} \begin{cases} z(x,0) = xe^{-x} \\ z(0,y) = y^3 \\ z(x,1) = e^{-x}(x+1) \\ z(1,y) = (1+y^3)e^{-1} \end{cases}$$







QP using (classic TFC approach)

$$\max_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^{\mathrm{T}}Q\mathbf{x} + \mathbf{c}^{\mathrm{T}}\mathbf{x} \quad : \quad \underset{m\times n}{A}\,\mathbf{x} = \mathbf{b}$$

Classic TFC approach: $\mathbf{x} = \mathbf{g} + \underset{n\times m}{H}\,\boldsymbol{\eta}$

$$A(\mathbf{g} + H\boldsymbol{\eta}) = \mathbf{b} \quad \rightarrow \quad \boldsymbol{\eta} = (AH)^{-1}(\mathbf{b} - A\mathbf{g})$$

$$\mathbf{x} = \underbrace{H(AH)^{-1}\mathbf{b}}_{\mathbf{x}_0} + \underbrace{[I_{n\times n} - H(AH)^{-1}A]}_{D}\,\mathbf{g} = \mathbf{x}_0 + D\mathbf{g}$$

$$f(\mathbf{g}) = \frac{1}{2}(\mathbf{x}_0 + D\mathbf{g})^{\mathrm{T}}Q(\mathbf{x}_0 + D\mathbf{g}) + \mathbf{c}^{\mathrm{T}}(\mathbf{x}_0 + D\mathbf{g})$$

$$\frac{\mathrm{d}f(\mathbf{g})}{\mathrm{d}\mathbf{g}} = \mathbf{0} \quad \rightarrow \quad \mathcal{A}\mathbf{g} + \mathbf{d} = \mathbf{0} \quad \rightarrow \quad \begin{cases} \mathcal{A} = D^{\mathrm{T}}QD \\ \mathbf{d} = D^{\mathrm{T}}(Q\mathbf{x}_0 + \mathbf{c}) \end{cases}$$

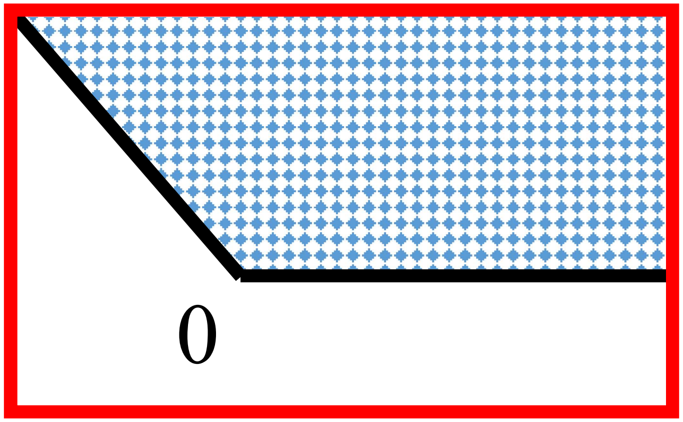
$$\mathcal{A} = U\Sigma V^{\mathrm{T}} \quad \rightarrow \mathcal{A}^{+} = U\Sigma^{+}V^{\mathrm{T}} \quad \rightarrow \quad \boxed{\mathbf{x} = \mathbf{x}_0 - DV\Sigma^{+}U^{\mathrm{T}}\mathbf{d}}$$

Equivalent equality constraints

(… what if $\text{rank}(A) = p < m$)

$$\underbrace{A}_{m\times n}\mathbf{x} = \mathbf{b} \quad \rightarrow \quad AP = \underbrace{Q}_{m\times m}\ \underbrace{R}_{m\times n} \quad \rightarrow \quad Q^{\mathrm{T}} A\mathbf{x} = RP^{\mathrm{T}}\mathbf{x} = Q^{\mathrm{T}}\mathbf{b}$$

(Rank Revealing QR decomposition)



$QQ^{\mathrm{T}} = I_{m\times m}$ and $R \equiv$ upper trapezoidal

$\mathbf{e} = \text{diag}(R)$

$|\mathbf{e}_1| \geq |\mathbf{e}_2| \geq \ \dots,$

$\text{rank}(A) = p$ where $|\mathbf{e}_p| > \varepsilon \max\{m, n\}|\mathbf{e}_1|$

$$RP^{\mathrm{T}}\mathbf{x} = Q^{\mathrm{T}}\mathbf{b} \quad \rightarrow \quad \underbrace{\tilde{A}}_{p\times n}\mathbf{x} = \tilde{\mathbf{b}} \quad \rightarrow \quad \boxed{\mathbf{x}_0 = \widetilde{H}(\tilde{A}\widetilde{H})^{-1}\tilde{\mathbf{b}}}$$

QP using (approach #2)

$$\max_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^{\mathrm{T}}Q\mathbf{x} + \mathbf{c}^{\mathrm{T}}\mathbf{x}, \quad \text{subject to} \quad \underset{m\times n}{A}\,\mathbf{x} = \mathbf{b}$$

$$AN = 0 \quad \rightarrow \quad \mathbf{x} = \mathbf{x}_0 + \underset{n\times r}{N}\,\mathbf{g} \quad \text{where} \quad r = n - m$$

$$\mathbf{x}_0 = H(AH)^{-1}\mathbf{b} \quad \text{or} \quad \mathbf{x}_0 = A^{\mathrm{T}}\left(AA^{\mathrm{T}}\right)^{-1}\mathbf{b}$$

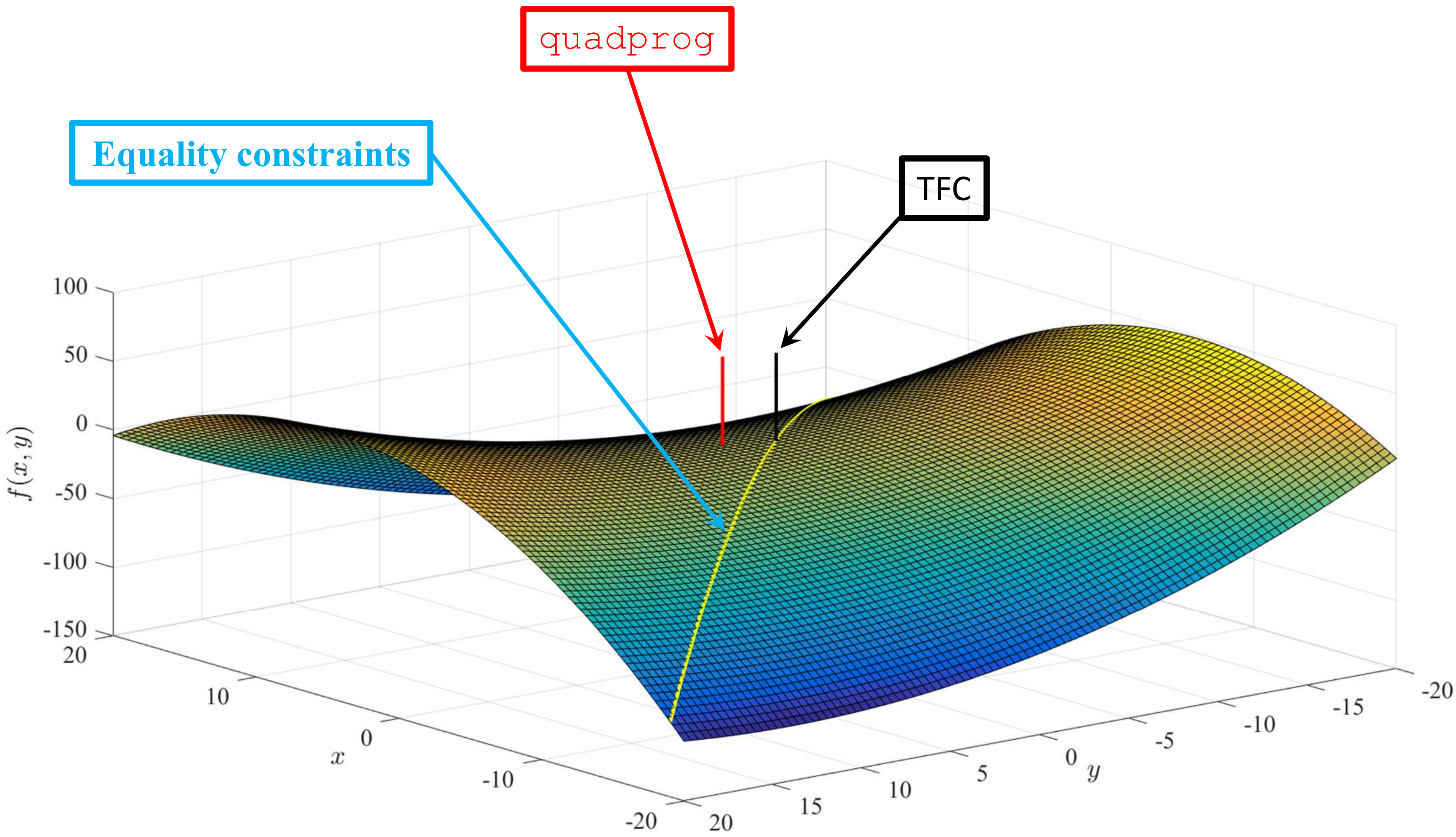
$$f(\mathbf{g}) = \frac{1}{2}(\mathbf{x}_0 + N\mathbf{g})^{\mathrm{T}}Q(\mathbf{x}_0 + N\mathbf{g}) + \mathbf{c}^{\mathrm{T}}(\mathbf{x}_0 + N\mathbf{g})$$

$$\frac{\mathrm{d}f(\mathbf{g})}{\mathrm{d}\mathbf{g}} = \mathbf{0} \quad \rightarrow \quad \mathcal{B}\mathbf{g} + \mathbf{e} = \mathbf{0} \quad \rightarrow \quad \begin{cases} \mathcal{B} = N^{\mathrm{T}}QN \\ \mathbf{e} = N^{\mathrm{T}}(Q\mathbf{x}_0 + \mathbf{c}) \end{cases}$$

$$\boxed{\mathbf{x} = \mathbf{x}_0 - N\mathcal{B}^{-1}\mathbf{e}}$$

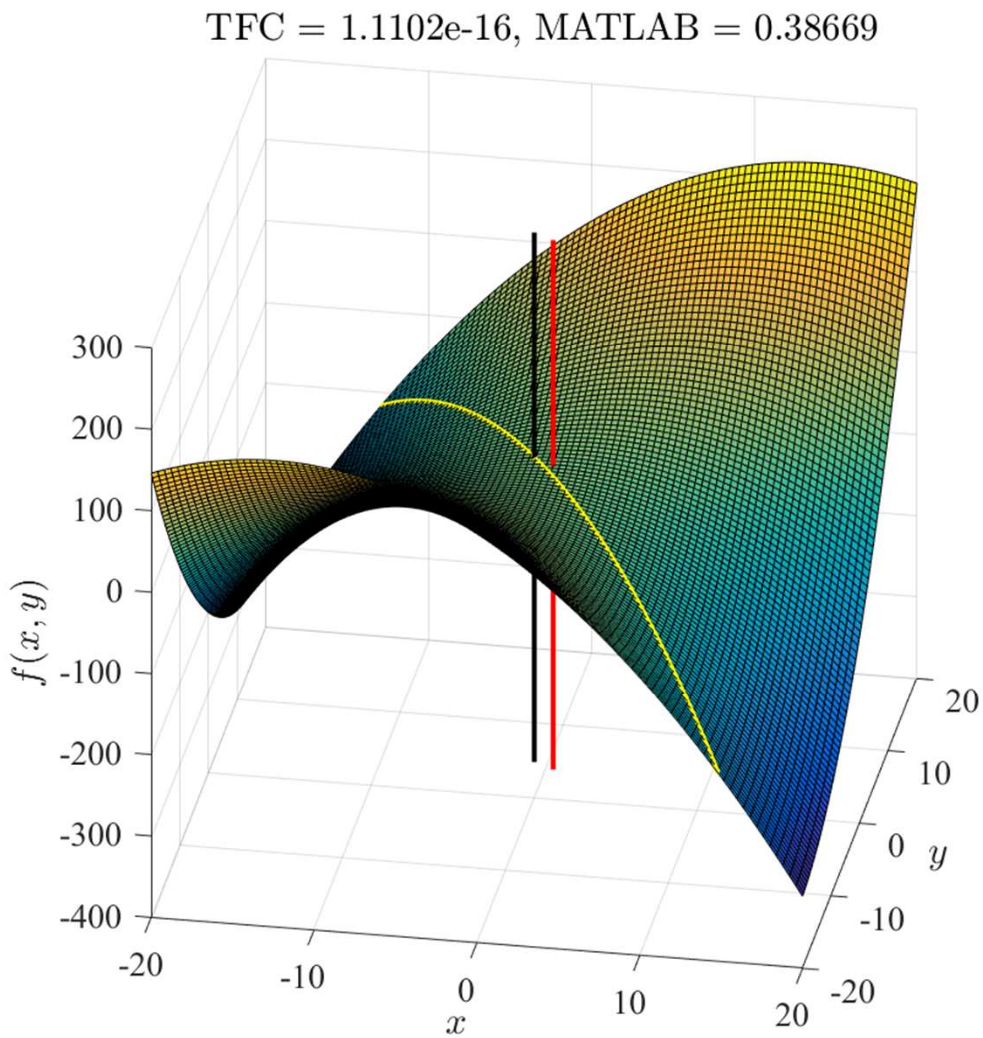
R2016b `quadprog` requires

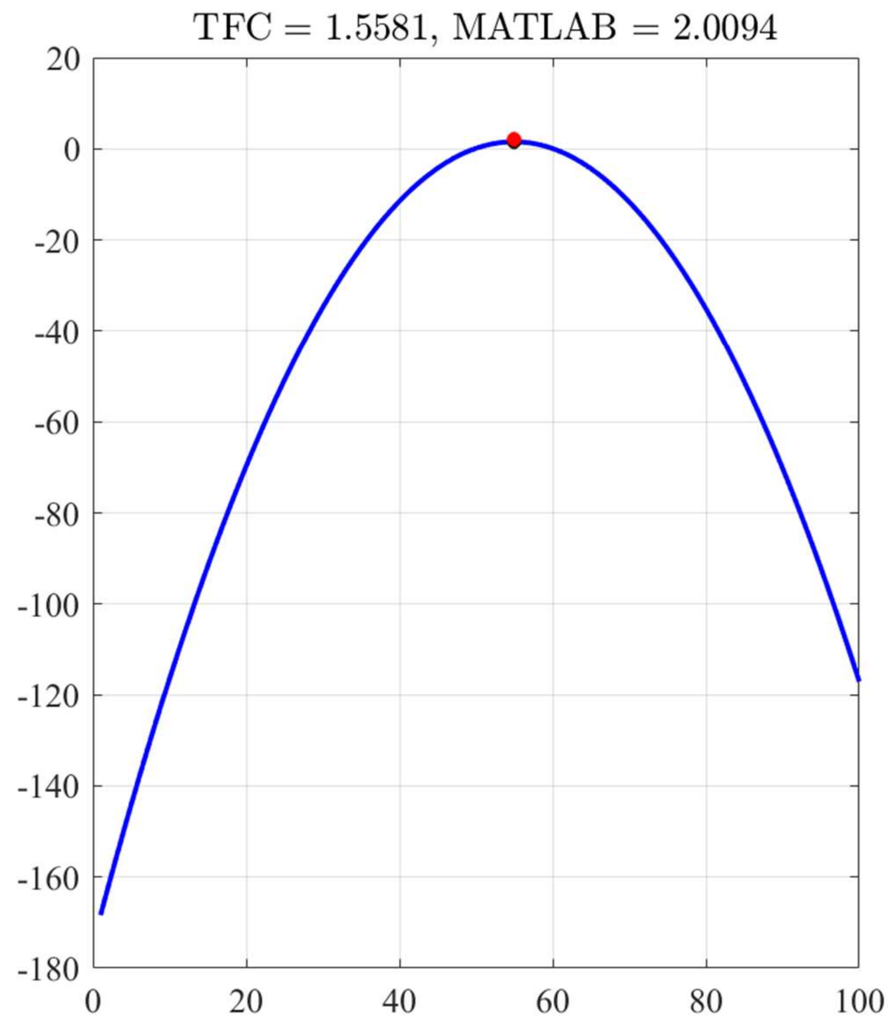
the reduced Hessian $N^{\mathrm{T}}QN$ must be positive definite.



Accuracy tests with R2016b `quadprog`

$$\left|A\mathbf{x}-\mathbf{b}\right|_2 \qquad f(\mathbf{x})=\frac{1}{2}\mathbf{x}^{\mathrm{T}}Q\mathbf{x}+\mathbf{c}^{\mathrm{T}}\mathbf{x}$$





Speed tests with R2019a `quadprog`

n	m	TFC	`quadprog`	time ratio
10	2	0.027746	0.87057	31.3759
10	4	0.029558	0.92745	31.3768
10	8	0.032127	0.92885	28.9121
20	4	0.044136	0.77945	17.6601
20	8	0.051081	0.94080	18.418
20	16	0.085394	0.94296	11.0425
40	8	0.088137	0.84634	9.6026
40	16	0.131900	0.81007	6.1415
40	32	0.198200	0.82851	4.1802
80	16	0.273250	0.96279	3.5235
80	32	0.394500	1.11990	2.8388
80	64	0.677070	1.32050	1.9502

NLP using TFC

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) \quad : \quad A\mathbf{x} = \mathbf{b}$$

$$\left\{f(\mathbf{x}) \in \mathbb{R}^n \to \mathbb{R}^1, \quad A \in \mathbb{R}^{m\times n}, \text{rank}(A) = m < n, \quad \mathbf{b} \in \mathbb{R}^m\right\}$$

$$\mathbf{x} = \mathbf{x}_0 + N\mathbf{g} \quad \text{where} \quad \left\{r = n - m, \quad N \in \mathbb{R}^{n\times r}, \quad \mathbf{g} \in \mathbb{R}^r\right\}$$

$$f(\mathbf{x}) = f(\mathbf{x}_0) + \mathbf{J}^{\mathrm{T}}(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) + \frac{1}{2}(\mathbf{x} - \mathbf{x}_0)^{\mathrm{T}} H(\mathbf{x}_0) H(\mathbf{x} - \mathbf{x}_0) + \mathrm{O}(\mathbf{x}^3)$$

$$h(\mathbf{g}) = \hat{h}(\mathbf{g}) + \text{HOT} = f(\mathbf{x}_0) + \mathbf{J}_0^{\mathrm{T}} N\mathbf{g} + \frac{1}{2}\mathbf{g}^{\mathrm{T}} N^{\mathrm{T}} H_0 N\mathbf{g} + \mathrm{O}(\mathbf{g}^3)$$

$$\mathbf{J}_k = \nabla f\left(\mathbf{x}^{(k)}\right) = \begin{Bmatrix} \dfrac{\partial f}{\partial x_1} \\ \vdots \\ \dfrac{\partial f}{\partial x_n} \end{Bmatrix}_{\mathbf{x}^{(k)}} \quad \text{and} \quad H_k = \nabla^2 f\left(\mathbf{x}^{(k)}\right) = \begin{bmatrix} \dfrac{\partial^2 f}{\partial x_1^2} & \cdots & \dfrac{\partial^2 f}{\partial x_1 x_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial^2 f}{\partial x_n x_1} & \cdots & \dfrac{\partial^2 f}{\partial x_n^2} \end{bmatrix}_{\mathbf{x}^{(k)}}$$

NLP using TFC

$$\hat{h}(\mathbf{g}) \approx f(\mathbf{x}_0) + \mathbf{J}_0^{\mathrm{T}} N \mathbf{g} + \frac{1}{2}\mathbf{g}^{\mathrm{T}} N^{\mathrm{T}} H_0 N \mathbf{g}$$

$$\frac{\partial \hat{h}(\mathbf{g})}{\partial \mathbf{g}} = \mathbf{0} \quad \rightarrow \quad \mathbf{g}^{(1)} = -\left(N^{\mathrm{T}} H_0 N\right)^{-1} N^{\mathrm{T}} \mathbf{J}_0$$

$$\mathbf{x}^{(1)} = \mathbf{x}_0 + N\mathbf{g}^{(1)} = \mathbf{x}_0 - N\left(N^{\mathrm{T}} H_0 N\right)^{-1} N^{\mathrm{T}} \mathbf{J}_0$$

and the iterative process is

$$\mathbf{x}^{(k+1)} = \mathbf{x}_0 + N\mathbf{g}^{(k+1)}$$

$$= \mathbf{x}_0 - \sum_{j=0}^{k} N\left(N^{\mathrm{T}} H_j N\right)^{-1} N^{\mathrm{T}} \mathbf{J}_j$$

NLP using TFC (Full nonlinear)

$$\mathcal{L}(\mathbf{g}) := \frac{\partial \hat{h}(\mathbf{g})}{\partial \mathbf{g}} = \mathbf{0}$$

and the iterative process is

$$\mathbf{g}^{(k+1)} = \mathbf{g}^{(k)} - (F_k)^{-1}\mathbf{E}_k = -\sum_{j=0}^{k} \left(F_j\right)^{-1}\mathbf{E}_j$$

where $\begin{cases} \mathbf{E}_j = \nabla h(\mathbf{g})\Big|_{\mathbf{g}^{(j)}} = N^{\mathrm{T}}\mathbf{J}_j \\ F_j = \nabla^2 h(\mathbf{g})\Big|_{\mathbf{g}^{(j)}} = N^{\mathrm{T}} H_j N \end{cases}$

Convergence occurs when

$$\left\|\mathbf{g}^{(k+1)} - \mathbf{g}^{(k)}\right\|_2 < \varepsilon_{\mathbf{g}} \quad \text{or} \quad \left\|\mathcal{L}\left(\mathbf{g}^{(k)}\right)\right\|_2 < \varepsilon_{\mathcal{L}}$$

$$\mathbf{x}^{(k+1)} = \mathbf{x}_0 + N\mathbf{g}^{(k+1)}$$

Convergence Analysis

1) quadratic convergence rate

$$\|\mathbf{E}_{k+1}\|_2 \le \left(\frac{L\ \|N\|_2^3}{2m^2}\right)\ \|\mathbf{E}_k\|_2^2$$

where
$$\begin{cases} \mathbf{E}_k = \nabla h(\mathbf{g})\Big|_{\mathbf{g}^{(k)}} = N^{\mathrm{T}}\mathbf{J}_k \\ \|\nabla^2 f(\mathbf{x}) - \nabla^2 f(\mathbf{y})\|_2 \le L\|\mathbf{x}-\mathbf{y}\|_2 \end{cases}$$

2) bounded number of iterations

$$N\ \frac{M^2 L^2\ \|N\|_2^6}{\alpha\beta\ m^5\ \min\{1,\ 9(1-2\alpha)^2\}}\left[h\left(\mathbf{g}^{(0)}\right)-q^*\right]_{\max}$$

$\alpha \in (0,0.5)$, $\beta \in (0,1)$, and

$MI \succcurlyeq \nabla^2 h(\mathbf{g}) \succcurlyeq mI$

Conclusions

- Motivation and background on the TFC
  - Univariate and multivariate
  - Applications on ODEs
- QP subject to equality constraints
  - Approach #1
  - Equivalent reduced equality system
  - Approach #2
  - Accuracy and speed tests
- NLP subject to equality constraints
  - The 2nd order approach
  - Convergence analysis
    - Quadratic convergence
    - Bounded number of iterations
- **Inequality constraints**

MATLAB iterative approaches

- **Interior-point-convex**. This algorithm attempts to follow a path that is strictly inside the constraints.
- **Trust-region-reflective**. This algorithm is a subspace trust-region method based on the interior-reflective Newton method described in quadprog.

R2016b `quadprog` requires

the reduced Hessian $N^{\mathrm{T}}QN$ must be positive definite.