Nonlinear Parameter Continuation with COCO

Lecture given during
Advanced Summer School on
Continuation Methods for Nonlinear Problems

Harry Dankowicz

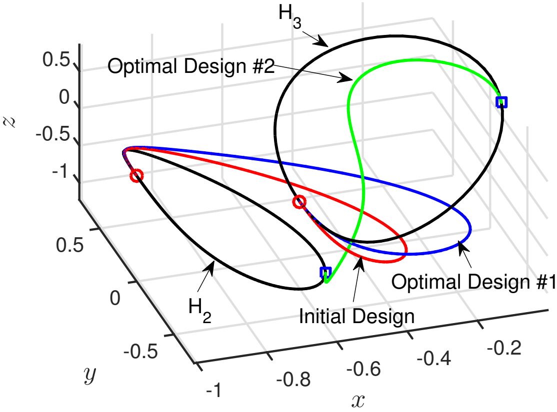
Department of Mechanical Science and Engineering
University of Illinois at Urbana-Champaign

August 13-24, 2018

Outline

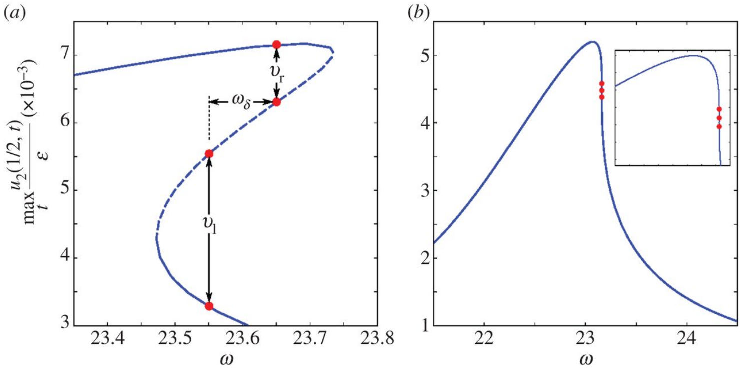
Motivation

For given time of travel, find a spacecraft transfer orbit between two halo orbits of known period around a libration point of the circular restricted three-body problem that minimizes fuel cost.



Motivation

For given total material volume, find a pair of values for the width and thickness of second layer of a microresonator that maximizes the dynamic range.



M. Saghafi, H. Dankowicz, W. Lacarbonara, Proc. R. Soc. A, 2015

Problem decomposition

- In each case, constraints restrict attention to an implicitly defined manifold along which optimization is performed.
- The constraint problem supports a natural decomposition into individually defined subproblems that are glued together during problem construction.
- Is the same true for the variational conditions that must hold at a local extremum?
- Given the use of parameter continuation in mapping out the implicitly defined submanifold, does this method play a role in the analysis of the optimization problem?

Problem decomposition

- In each case, constraints restrict attention to an implicitly defined manifold along which optimization is performed.
- The constraint problem supports a natural decomposition into individually defined subproblems that are glued together during problem construction.
- Is the same true for the variational conditions that must hold at a local extremum?
- Given the use of parameter continuation in mapping out the implicitly defined submanifold, does this method play a role in the analysis of the optimization problem?

Problem decomposition

- In each case, constraints restrict attention to an implicitly defined manifold along which optimization is performed.
- The constraint problem supports a natural decomposition into individually defined subproblems that are glued together during problem construction.
- Is the same true for the variational conditions that must hold at a local extremum?
- Given the use of parameter continuation in mapping out the implicitly defined submanifold, does this method play a role in the analysis of the optimization problem?

Problem decomposition

- In each case, constraints restrict attention to an implicitly defined manifold along which optimization is performed.
- The constraint problem supports a natural decomposition into individually defined subproblems that are glued together during problem construction.
- Is the same true for the variational conditions that must hold at a local extremum?
- Given the use of parameter continuation in mapping out the implicitly defined submanifold, does this method play a role in the analysis of the optimization problem?

Constrained optimization

Fundamental problem:

Find a locally *optimal solution* $(\hat{u}, \hat{\mu})$ that minimizes the function $(u, \mu) \mapsto \mu_1$ under the constraints imposed by the *extended continuation problem*

$$\begin{pmatrix} \Phi(u) \\ \Psi(u) - \mu \end{pmatrix} = 0$$

The *zero function* $\Phi : U \to Y$ and *monitor function* $\Psi : U \to \mathbb{R}^l$ are continuously Fréchet differentiable mappings; the *continuation variable space* U is a real Banach space; the image space Y is a real Banach space; and the components of the vector $\mu \in \mathbb{R}^l$ are the *continuation parameters*.

Constrained optimization

Define the scalar-valued *Lagrangian*

$$L(u, \mu, \lambda, \eta) := \mu_1 + \langle \lambda, \Phi(u) \rangle_Y + \eta^\top \cdot (\Psi(u) - \mu)$$

where $\langle \cdot, \cdot \rangle_Y : Y^* \times Y \to \mathbb{R}$ is the pairing of Y with its dual, and the vectors $\lambda \in Y^*$ and $\eta \in \mathbb{R}^l$ are the corresponding *Lagrange multipliers*.

Suppose that $(\hat{u}, \hat{\mu} = \Psi(\hat{u}))$ is an optimal solution, and that the range of $D\Phi(\hat{u})$ equals Y and its nullspace is of finite dimension $d \leq l$.

Constrained optimization

Then, there exist unique $\hat{\lambda}$ and $\hat{\eta}$ such that $\delta L = 0$, i.e.,

$$
\begin{aligned}
0 = \delta\mu_1 + \left\langle \delta\lambda, \Phi(\hat{u}) \right\rangle_Y + \delta\eta^\top \cdot (\Psi(\hat{u}) - \hat{\mu}) \\
+ \left\langle \hat{\lambda}, D\Phi(\hat{u})\delta u \right\rangle_Y + \hat{\eta}^\top \cdot (D\Psi(\hat{u})\delta u - \delta\mu)
\end{aligned}
$$

or, equivalently,

$$
\begin{aligned}
0 = \delta\mu_1 + \left\langle \delta\lambda, \Phi(\hat{u}) \right\rangle_Y + \delta\eta^\top \cdot (\Psi(\hat{u}) - \hat{\mu}) \\
+ \left\langle (D\Phi(\hat{u}))^* \hat{\lambda} + (D\Psi(\hat{u}))^* \hat{\eta}, \delta u \right\rangle_U - \hat{\eta}^\top \cdot \delta\mu
\end{aligned}
$$

for arbitrary variations δu, $\delta\mu$, $\delta\lambda$, and $\delta\eta$.

Constrained optimization

From the independence of the individual variations, we conclude that $\hat{\eta}_1 = 1$, $\hat{\eta}_{\{2,\ldots,l\}} = 0$, and

$$\Phi(\hat{u}) = 0,\ \Psi(\hat{u}) - \hat{\mu} = 0,\ (D\Phi(\hat{u}))^* \hat{\lambda} + (D\Psi(\hat{u}))^* \hat{\eta} = 0$$

and, consequently, that

$$\Phi(\hat{u}) = 0,\ \Psi(\hat{u}) - \hat{\mu} = 0,\ (D\Phi(\hat{u}))^* \hat{\lambda} + (D\Psi_1(\hat{u}))^* = 0$$

is a necessary condition for a local extremum at $(\hat{u}, \hat{\mu})$ along the constraint manifold defined implicitly by Φ.

Fundamental insights

A local optimum may be found using a method of successive continuation (Kernévez & Doedel, 1987) along manifolds defined by restrictions of the extended continuation problem

$$\begin{pmatrix} \Phi(u) \\ (D\Phi(u))^*\lambda + (D\Psi(u))^*\eta \\ \Psi(u) - \mu \\ \eta - \nu \end{pmatrix} = 0$$

obtained by fixing subsets of the continuation parameters μ and ν, until all components of μ are free and all components of ν equal 0, except ν_1 which equals 1.

Fundamental insights

Step 1: Detect a fold in μ_1 along a one-dimensional solution branch, with only trivial solution to adjoint problem away from fold, obtained by fixing $d - 1$ other components of μ.

Step 2: Continue along a solution path in λ and η until $\eta_1 = 1$, for the same choice of fixed μ's.

Step 3: Perform one or several runs of continuation, each time releasing one or several of the previously fixed components of μ, until all remaining elements of ν are fixed at 0.

Staged construction

Suppose that $Y = Y_1 \times \cdots \times Y_N$, $U = U_1 \times \cdots \times U_N$, and there is a sequence $0 \leq l_1 \leq \cdots \leq l_N = l$, such that

$$\Phi_n(u) = \phi_n(u_{\mathbb{K}_n^o}, u_n)$$

and

$$\Psi_{\{l_{n-1}+1,\ldots,l_n\}}(u) = \psi_n(u_{\mathbb{K}_n^o}, u_n)$$

in terms of the *n-th stage realizations* ϕ_n and ψ_n, and a sequence of subsets $\mathbb{K}_n^o \subseteq \{1, \ldots, n-1\}$, such that $\mathbb{K}_1^o = \emptyset$.

Accordingly, in the n-th stage of construction, we define the realizations ϕ_n and ψ_n, the index set $\mathbb{K}_n^o$, the continuation variables u_n, and the continuation parameters $\mu_{\{l_{n-1}+1,\ldots,l_n\}}$.

Staged construction

If $\mathbb{K}_n^o = \emptyset$ for some $n > 1$, then the n-th stage realizations are independent of the continuation variables introduced in previous stages. In this case, the problem added at the n-th stage is *embedded* in the overall continuation problem.

Continuation problems that arise in practice often naturally decompose into distinct embedded problems whose mutual dependence is defined at later stages of construction in terms of *gluing conditions* that depend only on a small subset of the continuation variables associated with each embedded problem.

Staged construction

At each stage of construction, variations of the partial Lagrangian

$$\langle \lambda_n, \phi_n \rangle_{Y_n} + \eta_{\{l_{n-1}+1,\ldots,l_n\}}^{\top} \cdot \left(\psi_n - \mu_{\{l_{n-1}+1,\ldots,l_n\}} \right)$$

with respect to $u_{\mathbb{K}_n^o}$ and u_n yield

$$\left\langle (D\phi_n)^* \, \lambda_n + (D\psi_n)^* \, \eta_{\{l_{n-1}+1,\ldots,l_n\}}, (\delta u_{\mathbb{K}_n^o}, \delta u_n) \right\rangle_{U_{\mathbb{K}_n^o} \times U_n}$$

Each stage of construction contributes terms to partially populated adjoint equations associated with variations in $u_{\mathbb{K}_n^o}$, as well as new equations associated with variations in u_n. Only once all elements of Φ and Ψ have been constructed are the adjoint equations fully formed.

Staged construction

We visualize the details of this process by considering an upper triangular operator matrix representation of the adjoint $D(\Phi, \Psi)$ with rows associated with variations in the continuation variables and columns associated with the Lagrange multipliers.

At each stage of construction, the matrix representation grows by the addition of columns for the new Lagrange multipliers λ_n and $\eta_{\{l_{n-1}+1,\ldots,l_n\}}$ and rows for variations in the new continuation variables u_n.

Staged construction

$$\begin{array}{rl} \text{minimize:} & e(x(0), x(T), T, p) + \int_0^T g(x,p)dt \\ \text{subject to:} & \dot{x} - f(x,p) = 0, \quad b(x(0), x(T), T, p) = 0, \\ & \int_0^T h(x,p)dt = 0 \end{array}$$

$$J = \begin{pmatrix} -\dot{(\cdot)} - Tf_x & 0 & h_x & Tg_x & 0 \\ -(\cdot)|_{\tau=0} & b_{x(0)} & 0 & e_{x(0)} & 0 \\ (\cdot)|_{\tau=1} & b_{x(1)} & 0 & e_{x(1)} & 0 \\ -\langle(\cdot), f\rangle & b_T & 0 & e_T + \int_0^1 g \, \mathrm{d}\tau & 0 \\ -T\langle(\cdot), f_p\rangle & b_p & \int_0^1 h_p \, \mathrm{d}\tau & e_p + T\int_0^1 g_p \, \mathrm{d}\tau & I_q \end{pmatrix}$$

Staged construction

As the new equations do not depend on $\lambda_{\{1,\ldots,n-1\}}$ or $\eta_{\{1,\ldots,l_{n-1}\}}$, the corresponding entries are 0, giving the matrix an upper triangular/rectangular form.

$$\begin{pmatrix} J_1 & & & & X_1 & Y_1 \\ & J_2 & & & X_2 & Y_2 \\ & & \ddots & & \vdots & \vdots \\ & & & J_N & X_N & Y_N \end{pmatrix}$$

Accordingly, in the n-th stage of construction, we define the contributions to the matrix representation in terms of the realizations $\left(D\phi_n(u_{\mathbb{K}_n^\circ}, u_n)\right)^*$ and $\left(D\psi_n(u_{\mathbb{K}_n^\circ}, u_n)\right)^*$.

Code implementation

The MATLAB-based package COCO implements staged construction for the extended continuation problem, including the adjoint equations. As an example, to find the inflection point along the frequency-response curve of a linear oscillator:

Problem initialization:

```
>> prob = coco_prob;
>> prob = coco_set(prob, 'ode', 'autonomous', false);
>> prob = coco_set(prob, 'cont', 'NAdapt', 1);
>> prob = coco_set(prob, 'coll', 'NTST', 100);
```

Trajectory initialization:

```
>> [t1, x1]  = ode45(@(t,x) linode(t, x, 0.98), ...
     1.56164+[0 2*pi/0.98], [1.01958; 0]);
>> [t2, x2]  = ode45(@(t,x) linode(t, x, 0.88), ...
     1.49982+[0 2*pi/0.88], [1.10077; 0]);
```

Code implementation

During continuation of solutions to trajectory problems, functions defining first derivatives are optional. During continuation of solutions to trajectory problems and associated adjoint equations, functions defining first derivatives are required, while functions defining second derivatives are optional.

Trajectory problems:

```
>> coll_funcs = {@linode, @linode_dx, @linode_dp, ...
     @linode_dt, @linode_dxdx, @linode_dxdp, ...
     @linode_dpdp, @linode_dtdx, @linode_dtdp, ...
     @linode_dtdt};
>> prob = ode_isol2coll(prob, 'orb1', coll_funcs{:}, ...
     t1, x1, 'om', 0.98);
>> prob = ode_isol2coll(prob, 'orb2', coll_funcs{:}, ...
     t2, x2, 0.88);
```

Code implementation

Boundary conditions:

```
>> bc_funcs = {@linode_bc, @linode_bc_du, ...
     @linode_bc_dudu};
>> [data1, uidx1] = coco_get_func_data(prob, ...
     'orb1.coll', 'data', 'uidx');
>> maps1 = data1.coll_seg.maps;
>> [data2, uidx2] = coco_get_func_data(prob, ...
     'orb2.coll', 'data', 'uidx');
>> maps2 = data2.coll_seg.maps;
>> prob = coco_add_func(prob, 'po1', bc_funcs{:}, ...
     data1, 'zero', 'uidx', uidx1([maps1.x0_idx; ...
        maps1.x1_idx; maps1.T_idx; maps1.p_idx]));
>> prob = coco_add_func(prob, 'po2', bc_funcs{:}, ...
     data2, 'zero', 'uidx', uidx2([maps2.x0_idx; ...
       maps2.x1_idx; maps2.T_idx; maps2.p_idx]));
```

Code implementation

Gluing conditions and fitness function:

```
prob = coco_add_glue(prob, 'glue', uidx1(maps1.p_idx), ...
  uidx2(maps2.p_idx), -0.1);
prob = coco_add_glue(prob, 'amp', uidx1(maps1.x0_idx(1)), ...
  uidx2(maps2.x0_idx(1)), 'ampdiff', 'inactive');
```

Adjoints:

```
prob = adjt_isol2coll(prob, 'orb1');
prob = adjt_isol2coll(prob, 'orb2');
[data1, axidx1] = coco_get_adjt_data(prob, 'orb1.coll', 'data', 'axidx');
opt1 = data1.coll_opt;
[data2, axidx2] = coco_get_adjt_data(prob, 'orb2.coll', 'data', 'axidx');
opt2 = data2.coll_opt;
prob = coco_add_adjt(prob, 'po1', 'aidx', ...
  axidx1([opt1.x0_idx; opt1.x1_idx; opt1.T_idx; opt1.p_idx]));
prob = coco_add_adjt(prob, 'po2', 'aidx', ...
  axidx2([opt2.x0_idx; opt2.x1_idx; opt2.T_idx; opt2.p_idx]));
prob = coco_add_adjt(prob, 'glue', 'aidx', ...
  [axidx1(opt1.p_idx); axidx2(opt2.p_idx)]);
prob = coco_add_adjt(prob, 'amp', 'd.ampdiff', 'aidx', ...
  [axidx1(opt1.x0_idx(1)); axidx2(opt2.x0_idx(1))]);
```

Final reflections

- The COCO toolbox library supports construction of zero problems and adjoints associated with equilibrium points, autonomous and non-autonomous trajectory segments, multi-point boundary-value problems, and periodic-orbit problems, as well as autonomous multi-segment periodic orbit problems in hybrid dynamical systems.
- Homotopy can be used for initializing a solution guess for the zero problem. The solution guess for the Lagrange multipliers is the trivial zero solution. Nonzero solutions are obtained after branch-switching at fold points.
- Ongoing work aims to provide optimization support for periodic orbits in delay-differential equations with multiple constant delays, as well as to explore the topology of the extended solution manifold in $(u, \mu, \lambda, \eta, \nu)$.