Current Status of Homotopy-based Initialization in OpenModelica

10^{th} OpenModelica Annual Workshop

Patrick Täuber
Bernhard Bachmann

FH Bielefeld
University of
Applied Sciences

Department of Engineering and Mathematics
University of Applied Sciences Bielefeld, Germany

February 5, 2018

Contents

Homotopy Method

Why homotopy?

- Some nonlinear algebraic systems are hard to solve with standard methods
- Especially during initialization:
  - Large systems with lots of unknowns (algebraic variables, discrete variables, states, ...)
  - Good start values are needed for convergence

Homotopy Method

Why homotopy?

- Some nonlinear algebraic systems are hard to solve with standard methods
- Especially during initialization:
  - Large systems with lots of unknowns (algebraic variables, discrete variables, states, ...)
  - Good start values are needed for convergence

Idea of Homotopy

- Starting from solving a simplified problem the problem is transformed to the original one successively

→ Continuation method

→ Convergence is possible even without accurate initial guess values

Homotopy Method

General Approach

Nonlinear Problem

Solve non-linear equation system

$$\underline{F}(\underline{x}^*) = \underline{0}$$

with given start vector $\underline{x}_0 \in \mathbb{R}^n$.

Simple Example

$$f(x) = 2x - 4 + sin(2\pi x),$$

$$x_0 = 0.5, \quad x^* = 2.$$

Possible Homotopy Functions

FH Bielefeld University of Applied Sciences

Nonlinear Problem

Solve non-linear equation system

$$\underline{F}(\underline{x}^*) = \underline{0}$$

with given start vector $\underline{x}_0 \in \mathbb{R}^n$.

Possible Homotopy Functions

- Fixpoint-Homotopy:

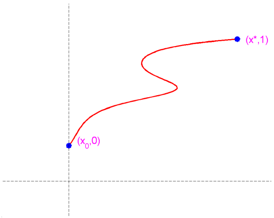
$$\underline{H}(\underline{x}, \lambda) = \lambda \underline{F}(\underline{x}) + (1 - \lambda)(\underline{x} - \underline{x}_0) = \underline{0}$$

Simple Example

$$f(x) = 2x - 4 + sin(2\pi x),$$

$$x_0 = 0.5, \quad x^* = 2.$$

Homotopy Path (Fixpoint)



Nonlinear Problem

Solve non-linear equation system

$$\underline{F}(\underline{x}^*) = \underline{0}$$

with given start vector $\underline{x}_0 \in \mathbb{R}^n$.

Possible Homotopy Functions

- Fixpoint-Homotopy:

$$\underline{H}(\underline{x}, \lambda) = \lambda \underline{F}(\underline{x}) + (1-\lambda)(\underline{x} - \underline{x}_0) = \underline{0}$$

- Newton-Homotopy:

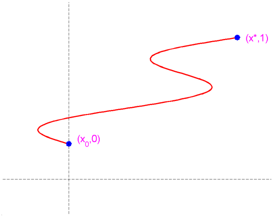
$$\underline{H}(\underline{x}, \lambda) = \underline{F}(\underline{x}) - (1-\lambda)\underline{F}(\underline{x}_0) = \underline{0}$$

Simple Example

$$f(x) = 2x - 4 + sin(2\pi x),$$

$$x_0 = 0.5, \quad x^* = 2.$$

Homotopy Path (Newton)



Homotopy Method

General Approach

- Modelica offers a homotopy operator for the homotopy-based initialization[1]

  → The homotopy function is defined by the user

Possible Homotopy Functions

- Fixpoint-Homotopy:

$$\underline{H}(\underline{x},\lambda) = \lambda\underline{F}(\underline{x}) + (1-\lambda)(\underline{x}-\underline{x}_0) = \underline{0}$$

- Newton-Homotopy:

$$\underline{H}(\underline{x},\lambda) = \underline{F}(\underline{x}) - (1-\lambda)\underline{F}(\underline{x}_0) = \underline{0}$$

- Homotopy operator for initialization:

$$homotopy(actual = actual, simplified = simplified) = \lambda actual + (1-\lambda) simplified$$

[1] Modelica Language Specification 3.4, p. 23

- Modelica offers a homotopy operator for the homotopy-based initialization[1]
  → The homotopy function is defined by the user

Possible Homotopy Functions

- Fixpoint-Homotopy:

$$\underline{H}(\underline{x}, \lambda) = \lambda \underline{F}(\underline{x}) + (1 - \lambda)(\underline{x} - \underline{x}_0) = \underline{0}$$

- Newton-Homotopy:

$$\underline{H}(\underline{x}, \lambda) = \underline{F}(\underline{x}) - (1 - \lambda)\underline{F}(\underline{x}_0) = \underline{0}$$

- Homotopy operator for initialization:

$$homotopy(actual = actual, simplified = simplified) = \lambda actual + (1 - \lambda) simplified$$

[1] Modelica Language Specification 3.4, p. 23

Homotopy Iteration

Start with $(\underline{x}_0, \underbrace{\lambda_0}_{=0})$ and $\underline{H}(\underline{x}_0, \lambda_0) = \underline{0}$.

Determine $(\underline{x}_{i+1}, \lambda_{i+1})$ with $\underline{H}(\underline{x}_{i+1}, \lambda_{i+1}) = \underline{0}$.

Stop, when $\lambda_m = 1$ yields.

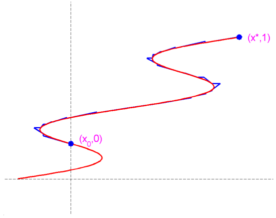
$$\Rightarrow \underline{H}(\underline{x}_m, \underbrace{\lambda_m}_{=1}) = \underline{F}(\underline{x}_m) = \underline{0} \Rightarrow \underline{x}^* = \underline{x}_m.$$

Simple Example

$$f(x) = 2x - 4 + sin(2\pi x),$$

$$x_0 = 0.5, \quad x^* = 2.$$

Homotopy Path (Iteration)



Homotopy Iteration

Start with $(\underline{x}_0, \underbrace{\lambda_0}_{=0})$ and $\underline{H}(\underline{x}_0, \lambda_0) = \underline{0}$.

Determine $(\underline{x}_{i+1}, \lambda_{i+1})$ with $\underline{H}(\underline{x}_{i+1}, \lambda_{i+1}) = \underline{0}$.

Stop, when $\lambda_m = 1$ yields.

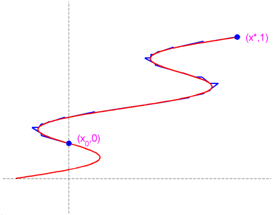
$$\Rightarrow \underline{H}(\underline{x}_m, \underbrace{\lambda_m}_{=1}) = \underline{F}(\underline{x}_m) = \underline{0} \Rightarrow \underline{x}^* = \underline{x}_m.$$

Simple Example

$$f(x) = 2x - 4 + sin(2\pi x),$$

$$x_0 = 0.5, \quad x^* = 2.$$

Homotopy Path (Iteration)



Homotopy Iteration

Start with $(\underline{x}_0, \underbrace{\lambda_0}_{=0})$ and $\underline{H}(\underline{x}_0, \lambda_0) = \underline{0}$.

Determine $(\underline{x}_{i+1}, \lambda_{i+1})$ with $\underline{H}(\underline{x}_{i+1}, \lambda_{i+1}) = \underline{0}$.

Stop, when $\lambda_m = 1$ yields.

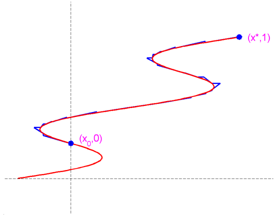
$$\Rightarrow \underline{H}(\underline{x}_m, \underbrace{\lambda_m}_{=1}) = \underline{F}(\underline{x}_m) = \underline{0} \Rightarrow \underline{x}^* = \underline{x}_m.$$

Simple Example

$$f(x) = 2x - 4 + sin(2\pi x),$$

$$x_0 = 0.5, \quad x^* = 2.$$

Homotopy Path (Iteration)



Homotopy Approaches in OM

(OMCompiler v1.13.0-dev.305+g3f877ad)

- Several different homotopy approaches are available in OpenModelica:
  - Homotopy solver as nonlinear system solver
    - Introduces the homotopy function by itself (fixpoint, newton)
    - Default fallback method if newton algorithm fails
  - Homotopy approaches for solving the initial systems
    - Usage of homotopy operator
    - Local/global homotopy with equidistant λ-steps
    - Local/global homotopy with the use of the homotopy solver (adaptive step size)

Homotopy Approaches in OM
(OMCompiler v1.13.0-dev.305+g3f877ad)

- Several different homotopy approaches are available in OpenModelica:
  - Homotopy solver as nonlinear system solver
    - Introduces the homotopy function by itself (fixpoint, newton)
    - Default fallback method if newton algorithm fails
  - Homotopy approaches for solving the initial systems
    - Usage of homotopy operator
    - Local/global homotopy with equidistant λ-steps
    - Local/global homotopy with the use of the homotopy solver (adaptive step size)

- Each nonlinear system with homotopy operator is solved separately with its own homotopy iteration

Modelica Model

```
model homotopy4 "testsuite/simulation/modelica/initialization/homotopy4.mos"
  Real x,y,z,x1,y1,z1,x2,y2,z2,x3,y3,z3,a,b,c,d,e,f;
  parameter Real p1 = 10;
equation
  a = sin(p1* time + 3.1415/2);

  x + 5 = z + a;
  -y + 10 = z + p1;
  x - y + 9 = z;

  b = sin(a*x);

  // (Beginning of homotopy iteration loop)
  c = homotopy(b^2, b);
  // (End of homotopy iteration loop)

  x1 + 5 = z1 + c;
  -y1 + 10 = z1 + p1;
  x1 - y1 + 9 = z1;

  d = x1 + y1 + z1;

  // Beginning of homotopy iteration loop
  homotopy(x2^2, x2) + 5 = z2 + 5;
  homotopy(y2^2, -d) + 10 = z2 + p1;
  x2 + y2 + 9 = z2;
  // End of homotopy iteration loop

  e = sin(d*x2);

  x3 + 5 = z3 + e;
  -y3 + 10 = z3*y3 + p1;
  x3 - y3 + 9 = z3;

  f = x + y + z + a + b + c + d + e + x1 + y1 + z1 + x2 + y2 + z2
         + x3 + y3 + z3;
end homotopy4;
```

Homotopy Approaches in OM

Local homotopy

- Each nonlinear system with homotopy operator is solved separately with its own homotopy iteration

- Faster alternative, because homotopy is only performed for components which are directly affected

- Previous homotopy iterations do not affect later systems (problematic if homotopy operator is used on a variable that is an input to the system)

Modelica Model

```
model homotopy4 "testsuite/simulation/modelica/initialization/homotopy4.mos"
  Real x,y,z,x1,y1,z1,x2,y2,z2,x3,y3,z3,a,b,c,d,e,f;
  parameter Real p1 = 10;
equation
  a = sin(p1* time + 3.1415/2);

  x + 5 = z + a;
  -y + 10 = z + p1;
  x - y + 9 = z;

  b = sin(a*x);

  // (Beginning of homotopy iteration loop)
  c = homotopy(b^2, b);
  // (End of homotopy iteration loop)

  x1 + 5 = z1 + c;
  -y1 + 10 = z1 + p1;
  x1 - y1 + 9 = z1;

  d = x1 + y1 + z1;

  // Beginning of homotopy iteration loop
  homotopy(x2^2, x2) + 5 = z2 + 5;
  homotopy(y2^2, -d) + 10 = z2 + p1;
  x2 + y2 + 9 = z2;
  // End of homotopy iteration loop

  e = sin(d*x2);

  x3 + 5 = z3 + e;
  -y3 + 10 = z3*y3 + p1;
  x3 - y3 + 9 = z3;

  f = x + y + z + a + b + c + d + e + x1 + y1 + z1 + x2 + y2 + z2
      + x3 + y3 + z3;
end homotopy4;
```

Modelica Spec. 3.4, p. 26

It is recommended to perform (conceptually) one homotopy iteration over the whole model, and not several homotopy iterations over the respective non-linear algebraic equation systems. The reason is that the following structure can be present:

$$\underline{w} = \underline{f_1}(\underline{x}) \quad \text{// has homotopy operator}$$

$$\underline{0} = \underline{f_2}(\underline{\dot{x}}, \underline{x}, \underline{z}, \underline{w})$$

Modelica Model

```
model homotopy4 "
  testsuite/simulation/modelica/initialization/homotopy4.mos"
  Real x,y,z,x1,y1,z1,x2,y2,z2,x3,y3,z3,a,b,c,d,e,f;
  parameter Real p1 = 10;
equation
  a = sin(p1* time + 3.1415/2);

  x + 5 = z + a;
  -y + 10 = z + p1;
  x - y + 9 = z;

  b = sin(a*x);

  // Beginning of homotopy iteration loop
  c = homotopy(b^2, b);

  x1 + 5 = z1 + c;
  -y1 + 10 = z1 + p1;
  x1 - y1 + 9 = z1;

  d = x1 + y1 + z1;

  homotopy(x2^2, x2) + 5 = z2 + 5;
  homotopy(y2^2, -d) + 10 = z2 + p1;
  x2 + y2 + 9 = z2;

  e = sin(d*x2);

  x3 + 5 = z3 + e;
  -y3 + 10 = z3*y3 + p1;
  x3 - y3 + 9 = z3;
  // End of homotopy iteration loop

  f = x + y + z + a + b + c + d + e + x1 + y1 + z1 + x2 + y2 + z2
          + x3 + y3 + z3;
end homotopy4;
```

Modelica Spec. 3.4, p. 26

It is recommended to perform (conceptually) one homotopy iteration over the whole model, and not several homotopy iterations over the respective non-linear algebraic equation systems. The reason is that the following structure can be present:

$$\underline{w} = \underline{f_1}(\underline{x}) \quad \text{// has homotopy operator}$$
$$\underline{0} = \underline{f_2}(\underline{\dot{x}}, \underline{x}, \underline{z}, \underline{w})$$

- The above structure can be handled with homotopy

- The homotopy iteration loop can become quite large

Modelica Model

```
model homotopy4 "
  testsuite/simulation/modelica/initialization/homotopy4.mos"
  Real x,y,z,x1,y1,z1,x2,y2,z2,x3,y3,z3,a,b,c,d,e,f;
  parameter Real p1 = 10;
equation
  a = sin(p1* time + 3.1415/2);

  x + 5 = z + a;
  -y + 10 = z + p1;
  x - y + 9 = z;

  b = sin(a*x);

  // Beginning of homotopy iteration loop
  c = homotopy(b^2, b);

  x1 + 5 = z1 + c;
  -y1 + 10 = z1 + p1;
  x1 - y1 + 9 = z1;

  d = x1 + y1 + z1;

  homotopy(x2^2, x2) + 5 = z2 + 5;
  homotopy(y2^2, -d) + 10 = z2 + p1;
  x2 + y2 + 9 = z2;

  e = sin(d*x2);

  x3 + 5 = z3 + e;
  -y3 + 10 = z3*y3 + p1;
  x3 - y3 + 9 = z3;
  // End of homotopy iteration loop

  f = x + y + z + a + b + c + d + e + x1 + y1 + z1 + x2 + y2 + z2
        + x3 + y3 + z3;
end homotopy4;
```

Homotopy Approaches in OM

Homotopy with equidistant step size

- Homotopy with equidistant step size is available for both the local and global version

OM Compiler Flags

```
--homotopyApproach=equidistantLocal
--homotopyApproach=equidistantGlobal
```

- Homotopy with equidistant step size is available for both the local and global version
- The user defines a number of homotopy steps to be performed
  → λ is increased equidistantly
  (e.g. 3 steps ⇒ $\lambda = 0, 0.5, 1$)

OM Compiler Flags

```
--homotopyApproach=equidistantLocal
--homotopyApproach=equidistantGlobal
```

OM Runtime Flags

```
-ils=<value>  (default: 4)
```

Homotopy Approaches in OM

Homotopy with equidistant step size

- Homotopy with equidistant step size is available for both the local and global version
- The user defines a number of homotopy steps to be performed
  $\rightarrow$ λ is increased equidistantly
  (e.g. 3 steps $\Rightarrow \lambda = 0, 0.5, 1$)
- No sensitive homotopy parameters that influence the convergence
- λ can only be increased and does not react to changes in the properties of the homotopy path
  $\rightarrow$ Homotopy path is not followed correctly and can be lost easily

OM Compiler Flags

```
--homotopyApproach=equidistantLocal
--homotopyApproach=equidistantGlobal
```

OM Runtime Flags

```
-ils=<value>  (default: 4)
```

Backend preparation for the equidistant global approach:

- Creation and optimization of a separate λ_0-DAE
  - Duplicate DAE and internally set `--initOptModules+=replaceHomotopyWithSimplified`

Backend preparation for the equidistant global approach:

- Creation and optimization of a separate λ_0-DAE
  - Duplicate DAE and internally set `--initOptModules+=replaceHomotopyWithSimplified`

Default

The **equidistant global** approach is currently the default homotopy method for initialization. It is activated as **fallback method** if the normal initialization fails.

OM Runtime Flags

`-homotopyOnFirstTry`

to directly use the homotopy method to solve the initialization problem.

Homotopy Approaches in OM

Homotopy with adaptive step size

- Homotopy with adaptive step size is available for both the local and global version

OM Compiler Flags

```
--homotopyApproach=adaptiveLocal
--homotopyApproach=adaptiveGlobal
```

Homotopy Approaches in OM

Homotopy with adaptive step size

- Homotopy with adaptive step size is available for both the local and global version

- Use of homotopy solver with predictor-corrector steps
  - → λ is increased/decreased according to the properties of the path

OM Compiler Flags

```
--homotopyApproach=adaptiveLocal
--homotopyApproach=adaptiveGlobal
```

OM Runtime Flags

... *will be introduced on the following slides* ...

- Homotopy with adaptive step size is available for both the local and global version

- Use of homotopy solver with predictor-corrector steps
  - → λ is increased/decreased according to the properties of the path

- Homotopy parameter λ is calculated by the homotopy algorithm to follow the homotopy path

- Many parameters influence the behaviour of the algorithm (e.g. initial step size, tolerances, ...)

OM Compiler Flags

```
--homotopyApproach=adaptiveLocal
--homotopyApproach=adaptiveGlobal
```

OM Runtime Flags

... *will be introduced on the following slides* ...

Backend preparation for the adaptive approaches:

- Inlining of the homotopy operator
  - Internally set `--initOptModules+=inlineHomotopy`
- λ is introduced as a variable in each component with homotopy operator
  - Internally set `--initOptModules+=generateHomotopyComponents`

  $\rightarrow$ system size: $n \times (n+1)$
- Symbolic calculation of the Jacobian of the $n \times (n+1)$ system

Backend preparation for the adaptive approaches:

- Inlining of the homotopy operator
  - Internally set `--initOptModules+=inlineHomotopy`
- λ is introduced as a variable in each component with homotopy operator
  - Internally set `--initOptModules+=generateHomotopyComponents`

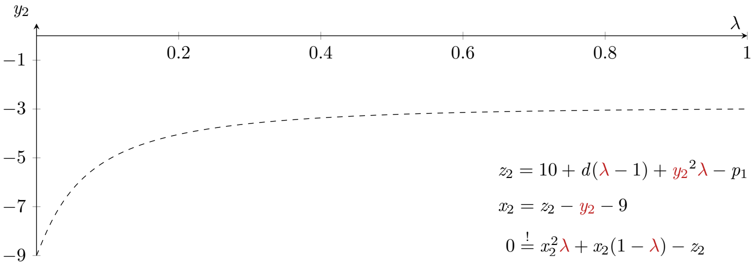
  $\rightarrow$ system size: $n \times (n+1)$
- Symbolic calculation of the Jacobian of the $n \times (n+1)$ system

Additionally for the **global** approach:

- Creation and optimization of a separate λ_0-DAE
  - Duplicate DAE and internally set `--initOptModules+=replaceHomotopyWithSimplified`
- Determination of the smallest homotopy iteration loop in the original initialization system
  - Internally set `--initOptModules+=generateHomotopyComponents`
  - Beginning: First BLT block in which a homotopy operator is present
  - End: Last BLT block that describes a non-linear algebraic equation system
  - Creation of one nonlinear system

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

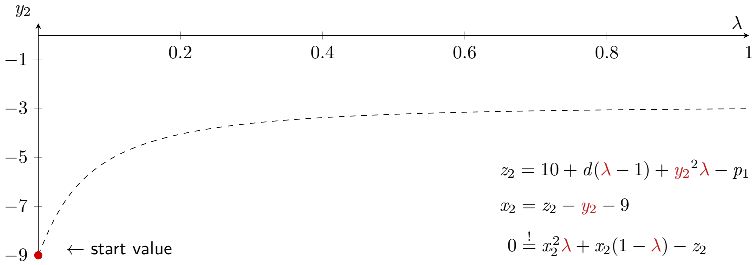
$$x_2 = z_2 - y_2 - 9$$

$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

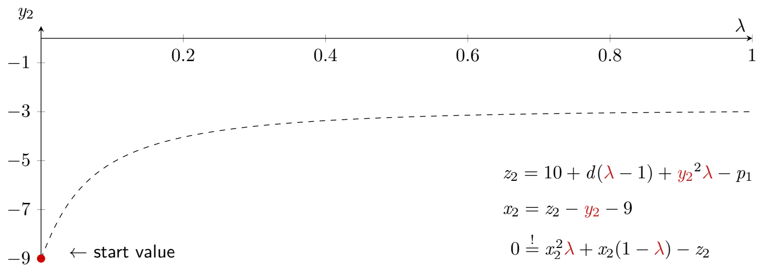
$$x_2 = z_2 - y_2 - 9$$

$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Predictor Step

Derive:

$$\underline{H}(\underline{x}(s), \lambda(s)) = \underline{0}$$

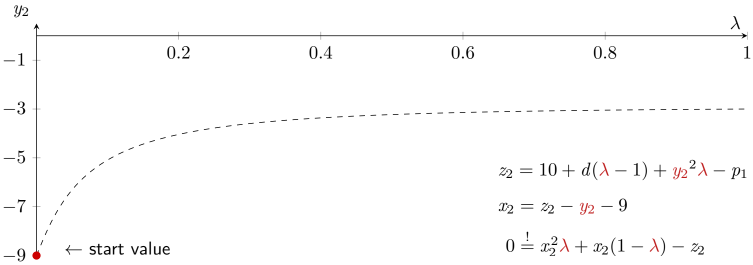
Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

FH Bielefeld
University of
Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



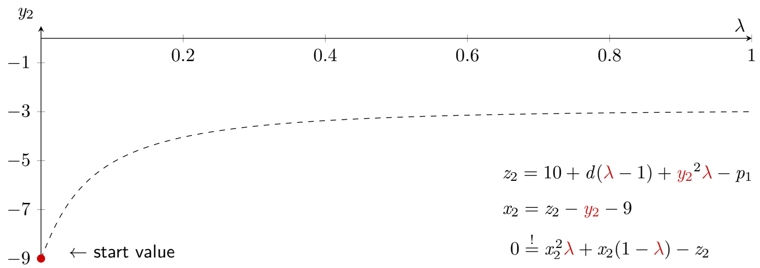
Procedure: Predictor Step

Derive:

$$\underline{H}(\underline{x}(s), \lambda(s)) = \underline{0}$$
$$\Rightarrow \frac{\partial \underline{H}}{\partial \underline{x}} \cdot \underline{x}'(s) + \frac{\partial \underline{H}}{\partial \lambda} \cdot \lambda'(s) = \underline{0}$$

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Predictor Step

Derive:

$$\underline{H}(\underline{x}(s), \lambda(s)) = \underline{0}$$

$$\Rightarrow \frac{\partial \underline{H}}{\partial \underline{x}} \cdot \underline{x}'(s) + \frac{\partial \underline{H}}{\partial \lambda} \cdot \lambda'(s) = \underline{0}$$

Solve linear system:

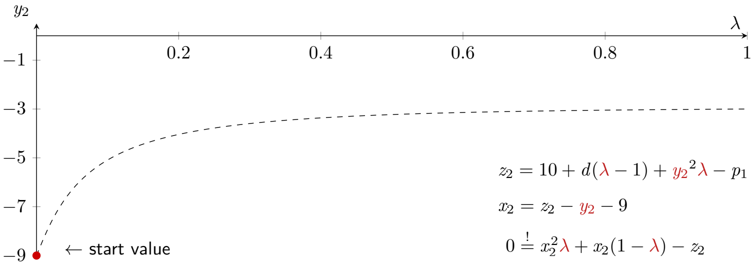
$$J_{\underline{H}}(\underline{x}_i, \lambda_i) \cdot \underline{v}_i = \underline{0}$$

$J_{\underline{H}} \in \mathbb{R}^{(n,n+1)}$, Jacobian

$$0 \overset{!}{\leq} (\underline{v}^i)^T \cdot \underline{v}^{i-1}$$

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure: Predictor Step

Derive:

$$\underline{H}(\underline{x}(s), \lambda(s)) = \underline{0}$$

$$\Rightarrow \frac{\partial \underline{H}}{\partial \underline{x}} \cdot \underline{x}'(s) + \frac{\partial \underline{H}}{\partial \lambda} \cdot \lambda'(s) = \underline{0}$$

Solve linear system:

$$J_{\underline{H}}(\underline{x}_i, \lambda_i) \cdot \underline{v}_i = \underline{0}$$

$J_{\underline{H}} \in \mathbb{R}^{(n,n+1)}$, Jacobian

$$0 \stackrel{!}{\leq} (\underline{v}^i)^T \cdot \underline{v}^{i-1}$$

Perform predictor step:

$$\begin{pmatrix} \underline{x}^\#_{i+1} \\ \lambda^\#_{i+1} \end{pmatrix} = \begin{pmatrix} \underline{x}_i \\ \lambda_i \end{pmatrix} + \tau_i \cdot \underline{v}_i,$$

τ_i step size, $\underline{v}_i$ normalized direction.

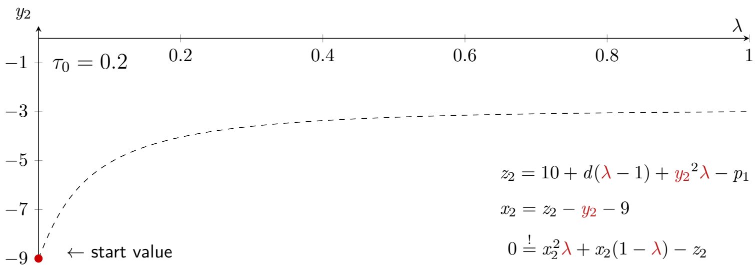
OM Runtime Flags

`-homTauStart=<value>`

to define the initial step size τ_0.

Default: 0.2

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Predictor Step

Derive:

$$\underline{H}(\underline{x}(s), \lambda(s)) = \underline{0}$$

$$\Rightarrow \frac{\partial \underline{H}}{\partial \underline{x}} \cdot \underline{x}'(s) + \frac{\partial \underline{H}}{\partial \lambda} \cdot \lambda'(s) = \underline{0}$$

Solve linear system:

$$J_{\underline{H}}(\underline{x}_i, \lambda_i) \cdot \underline{v}_i = \underline{0}$$

$J_{\underline{H}} \in \mathbb{R}^{(n,n+1)}$, Jacobian

$$0 \overset{!}{\leq} (\underline{v}^i)^T \cdot \underline{v}^{i-1}$$

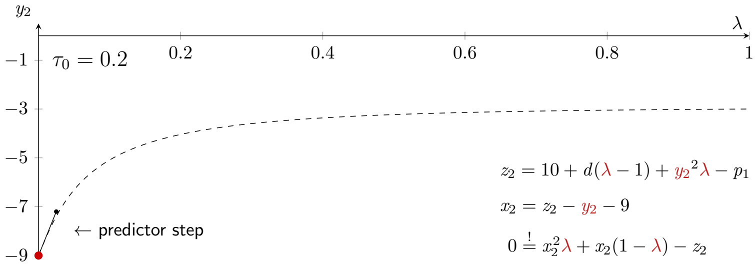
Perform predictor step:

$$\begin{pmatrix} \underline{x}_{i+1}^{\#} \\ \lambda_{i+1}^{\#} \end{pmatrix} = \begin{pmatrix} \underline{x}_i \\ \lambda_i \end{pmatrix} + \tau_i \cdot \underline{v}_i,$$

τ_i step size, $\underline{v}_i$ normalized direction.

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Predictor Step

Derive:

$$\underline{H}(\underline{x}(s), \lambda(s)) = \underline{0}$$
$$\Rightarrow \frac{\partial \underline{H}}{\partial \underline{x}} \cdot \underline{x}'(s) + \frac{\partial \underline{H}}{\partial \lambda} \cdot \lambda'(s) = \underline{0}$$

Solve linear system:

$$J_{\underline{H}}(\underline{x}_i, \lambda_i) \cdot \underline{v}_i = \underline{0}$$

$J_{\underline{H}} \in \mathbb{R}^{(n,n+1)}$, Jacobian

$$0 \overset{!}{\leq} (\underline{v}^i)^T \cdot \underline{v}^{i-1}$$

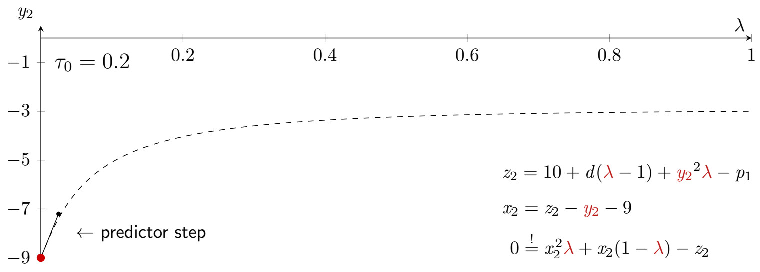
Perform predictor step:

$$\begin{pmatrix} \underline{x}_{i+1}^{\#} \\ \lambda_{i+1}^{\#} \end{pmatrix} = \begin{pmatrix} \underline{x}_i \\ \lambda_i \end{pmatrix} + \tau_i \cdot \underline{v}_i,$$

τ_i step size, $\underline{v}_i$ normalized direction.

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Predictor Step

- Assert?
  - If $\underline{H}(\underline{x}^{\#}_{i+1}, \lambda^{\#}_{i+1})$ cannot be calculated, τ is decreased by a factor: $\tau := \frac{\tau}{homTauDecFacPredictor}$

→ No assert in the example

Perform predictor step:

$$\begin{pmatrix} \underline{x}^{\#}_{i+1} \\ \lambda^{\#}_{i+1} \end{pmatrix} = \begin{pmatrix} \underline{x}_i \\ \lambda_i \end{pmatrix} + \tau_i \cdot \underline{v}_i,$$

τ_i step size, $\underline{v}_i$ normalized direction.

Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

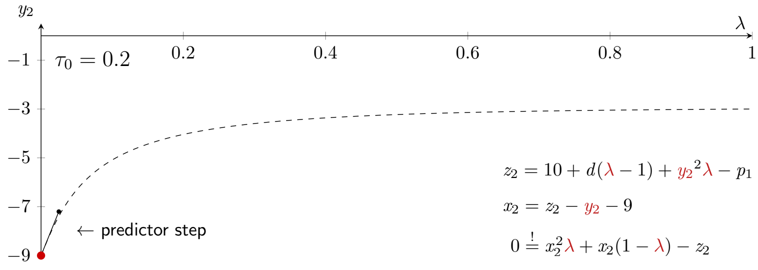
OM Runtime Flags

-homTauDecFacPredictor
=<value>

to define the factor by which τ is decreased if the predictor step fails.

Default: 2.0

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Predictor Step

- Assert?
  - If $\underline{H}(\underline{x}^{\#}_{i+1}, \lambda^{\#}_{i+1})$ cannot be calculated, τ is decreased by a factor:
    $\tau := \frac{\tau}{homTauDecFacPredictor}$

→ No assert in the example

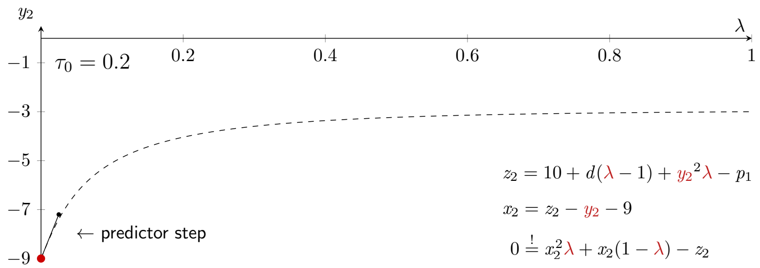
Perform predictor step:

$$\begin{pmatrix} \underline{x}^{\#}_{i+1} \\ \lambda^{\#}_{i+1} \end{pmatrix} = \begin{pmatrix} \underline{x}_i \\ \lambda_i \end{pmatrix} + \tau_i \cdot \underline{v}_i,$$

τ_i step size, $\underline{v}_i$ normalized direction.

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Corrector Step

<u>Newton</u>:

Get back to the homotopy path by Newton iteration steps with start values $(\underline{x}^{\#}_{i+1}, \lambda^{\#}_{i+1})$ until

$$\underline{H}(\underline{x}_{i+1}, \lambda_{i+1}) \approx \underline{0}.$$

Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

OM Runtime Flags

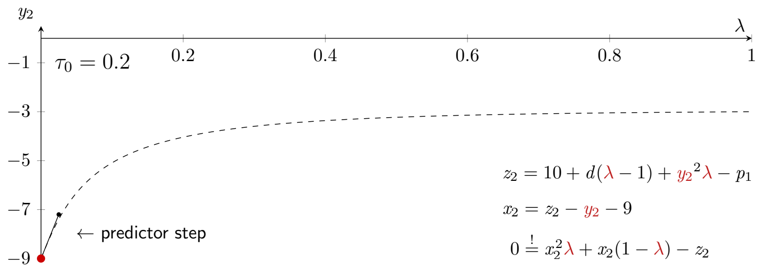
`-homHEps=<value>`

to define the tolerance for the residuals in the Newton corrector step.

Default: 1e-5

(In the last step ($\lambda = 1$) the global accuracy is used.)

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Corrector Step

Newton:
Get back to the homotopy path by Newton iteration steps with start values $(\underline{x}^{\#}_{i+1}, \lambda^{\#}_{i+1})$ until

$$\underline{H}(\underline{x}_{i+1}, \lambda_{i+1}) \approx \underline{0}.$$

Backtracing Strategies:

- Fix one coordinate:

$$x^{j}_{i+1} - x^{\#,j}_{i+1} = 0$$

- Orthogonal to tangent vector:

$$(\underline{x}_{i+1} - \underline{x}^{\#}_{i+1})^T \cdot \underline{v}_i = 0$$

Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

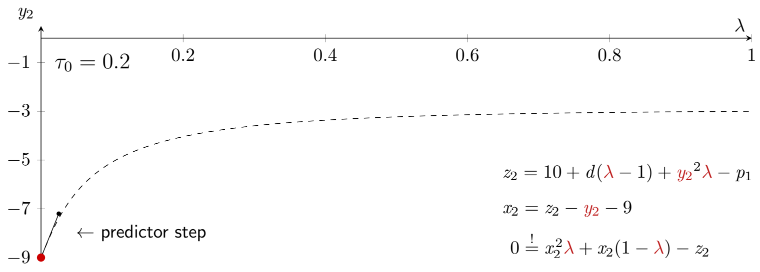
OM Runtime Flags

```
-homMaxNewtonSteps
=<value>
```

to define the maximum number of iterations in the Newton corrector step.

Default: 20

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



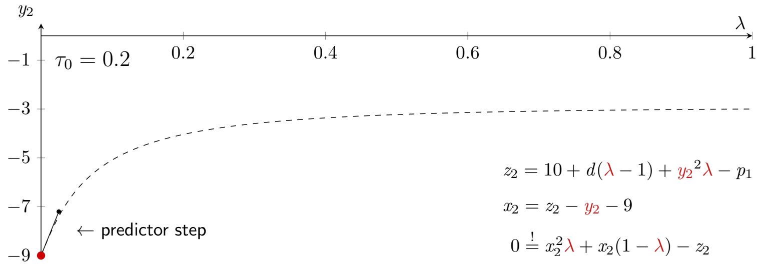
Procedure: Corrector Step

<u>Newton</u>:
Get back to the homotopy path by Newton iteration steps with start values $(\underline{x}^{\#}_{i+1}, \lambda^{\#}_{i+1})$ until

$$\underline{H}(\underline{x}_{i+1}, \lambda_{i+1}) \approx \underline{0}.$$

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Corrector Step

Newton:

Get back to the homotopy path by Newton iteration steps with start values $(\underline{x}^{\#}_{i+1}, \lambda^{\#}_{i+1})$ until

$$\underline{H}(\underline{x}_{i+1}, \lambda_{i+1}) \approx \underline{0}.$$

Backtracing Strategies:

1. Fix one coordinate:

$$x^{j_i}_{i+1} - x^{\#j_i}_{i+1} = 0$$

2. Orthogonal to tangent vector:

$$\left(\underline{x}_{i+1} - \underline{x}^{\#}_{i+1}\right)^T \cdot \underline{v}_i = 0$$

Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

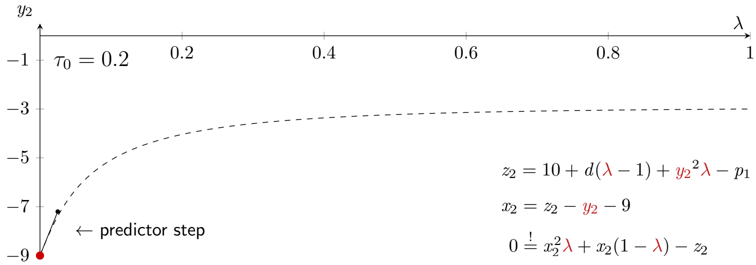
OM Runtime Flags

`-homBacktraceStrategy =<fix/orthogonal>`

to define the backtracing strategy in the Newton corrector step.

Default: fix

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure: Corrector Step

Newton:
Get back to the homotopy path by Newton iteration steps with start values $(\underline{x}^{\#}_{i+1}, \lambda^{\#}_{i+1})$ until

$$\underline{H}(\underline{x}_{i+1}, \lambda_{i+1}) \approx \underline{0}.$$

Backtracing Strategies:

1. Fix one coordinate:

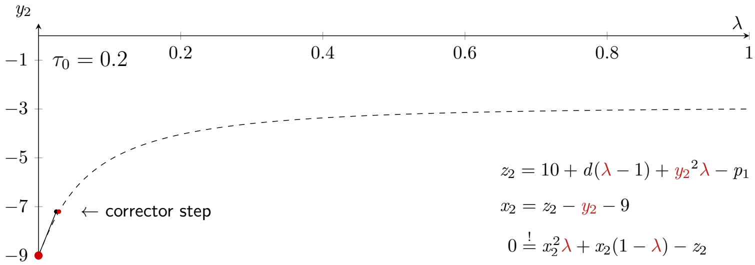
$$x^{j_i}_{i+1} - x^{\# j_i}_{i+1} = 0$$

2. Orthogonal to tangent vector:

$$(\underline{x}_{i+1} - \underline{x}^{\#}_{i+1})^T \cdot \underline{v}_i = 0$$

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Corrector Step

Newton:

Get back to the homotopy path by Newton iteration steps with start values $(\underline{x}^{\#}_{i+1}, \lambda^{\#}_{i+1})$ until

$$\underline{H}(\underline{x}_{i+1}, \lambda_{i+1}) \approx \underline{0}.$$

Backtracing Strategies:

1. Fix one coordinate:

$$x^{j_i}_{i+1} - x^{\# j_i}_{i+1} = 0$$

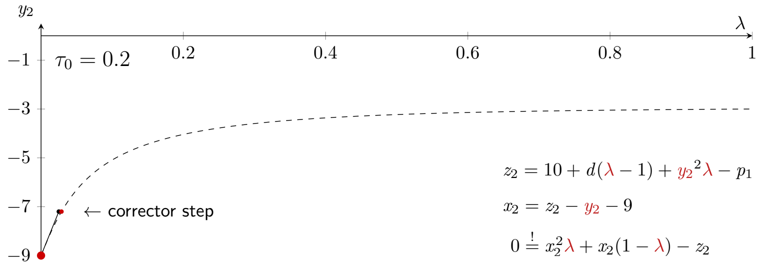
2. Orthogonal to tangent vector:

$$\left(\underline{x}_{i+1} - \underline{x}^{\#}_{i+1}\right)^T \cdot \underline{v}_i = 0$$

FH Bielefeld University of Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

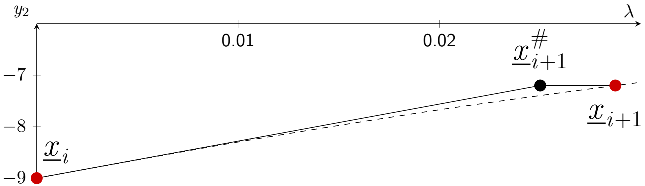
$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure: Check Step Acceptance and Adapt Step Size

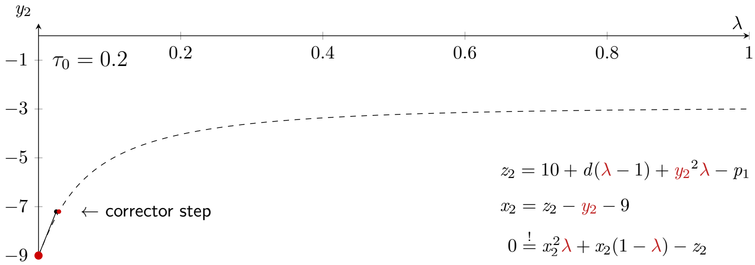
Calculate the bending:

$$b = \frac{\|\underline{x}_{i+1} - \underline{x}^{\#}_{i+1}\|}{\|\underline{x}^{\#}_{i+1} - \underline{x}_i\|}$$



OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure: Check Step Acceptance and Adapt Step Size

Calculate the bending:

$$b = \frac{\|x_{i+1} - x_{i+1}^{\#}\|}{\|x_{i+1}^{\#} - x_i\|}$$

$b > homAdaptBend$

→ Reject step!

→ $\tau := \frac{\tau}{homTauDecFac}$

→ Perform predictor and corrector step again with new τ

$b < homAdaptBend$

→ Accept step!

→ $b < \frac{homAdaptBend}{homTauIncThreshold}$

→ $\tau := \tau \cdot homTauIncFac$

Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

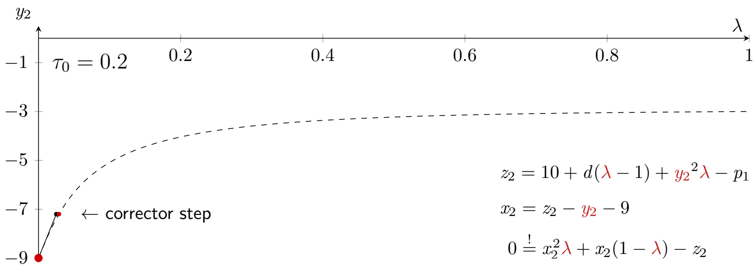
OM Runtime Flags

`-homAdaptBend=<value>`

to define the maximum trajectory bending to accept the homotopy step.

Default: 0.5

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Check Step Acceptance and Adapt Step Size

Calculate the bending:

$$b = \frac{\|x_{i+1} - x^{\#}_{i+1}\|}{\|x^{\#}_{i+1} - x_i\|}$$

$b > homAdaptBend$

→ Reject step!

→ $\tau := \frac{\tau}{homTauDecFac}$

→ Perform predictor and corrector step again with new τ

$b < homAdaptBend$

→ Accept step!

→ $b < \frac{homAdaptBend}{homTauIncThreshold}$

→ $\tau := \tau \cdot homTauIncFac$

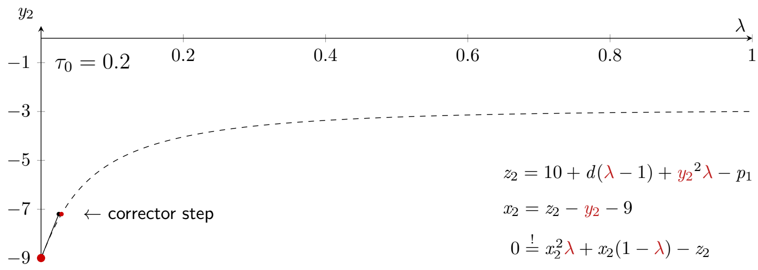
OM Runtime Flags

`-homTauDecFac=<value>`

to define the factor by which τ is decreased if the bending is too high.

Default: 10.0

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Check Step Acceptance and Adapt Step Size

Calculate the bending:

$$b = \frac{\|x_{i+1} - x^{\#}_{i+1}\|}{\|x^{\#}_{i+1} - x_i\|}$$

$b > homAdaptBend$

- → Reject step!
- → $\tau := \frac{\tau}{homTauDecFac}$
- → Perform predictor and corrector step again with new τ

$b < homAdaptBend$

- → Accept step!
- → $b < \frac{homAdaptBend}{homTauIncThreshold}$
  - → $\tau := \tau \cdot homTauIncFac$

Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

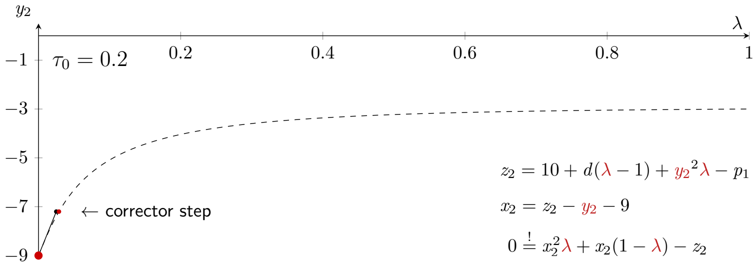
OM Runtime Flags

`-homTauMin=<value>`

to define the minimum step size τ.

Default: 1e-4

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Check Step Acceptance and Adapt Step Size

Calculate the bending:

$$b = \frac{\|x_{i+1} - x^{\#}_{i+1}\|}{\|x^{\#}_{i+1} - x_i\|}$$

$b > homAdaptBend$

- → Reject step!
- → $\tau := \frac{\tau}{homTauDecFac}$
- → Perform predictor and corrector step again with new τ

$b < homAdaptBend$

- → Accept step!
- → $b < \frac{homAdaptBend}{homTauIncThreshold}$
  - → $\tau := \tau \cdot homTauIncFac$

FH Bielefeld University of Applied Sciences

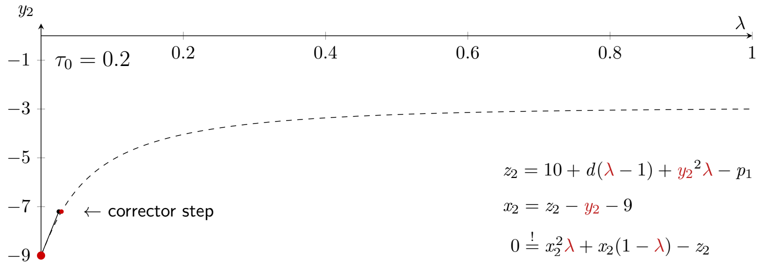
OM Runtime Flags

`-homTauIncThreshold =<value>`

to define a threshold for increasing τ.

Default: 10.0

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Check Step Acceptance and Adapt Step Size

Calculate the bending:

$$b = \frac{\|x_{i+1} - x^{\#}_{i+1}\|}{\|x^{\#}_{i+1} - x_i\|}$$

$b > homAdaptBend$

→ Reject step!

→ $\tau := \frac{\tau}{homTauDecFac}$

→ Perform predictor and corrector step again with new τ

$b < homAdaptBend$

→ Accept step!

→ $b < \frac{homAdaptBend}{homTauIncThreshold}$

→ $\tau := \tau \cdot homTauIncFac$

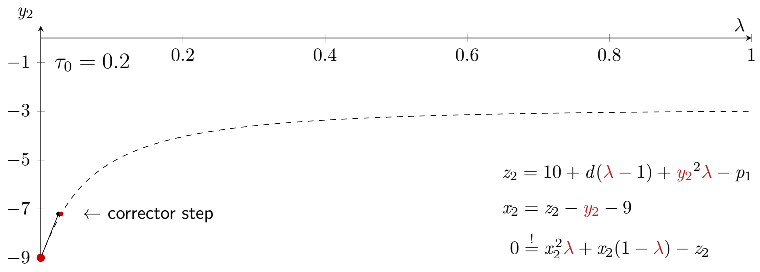
OM Runtime Flags

`-homTauIncFac=<value>`

to define the factor by which τ is increased if the bending is lower than the threshold.

Default: 2.0

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Check Step Acceptance and Adapt Step Size

Calculate the bending:

$$b = \frac{\|x_{i+1} - x^{\#}_{i+1}\|}{\|x^{\#}_{i+1} - x_i\|}$$

$b > homAdaptBend$

→ Reject step!

→ $\tau := \dfrac{\tau}{homTauDecFac}$

→ Perform predictor and corrector step again with new τ

$b < homAdaptBend$

→ Accept step!

→ $b < \dfrac{homAdaptBend}{homTauIncThreshold}$

→ $\tau := \tau \cdot homTauIncFac$

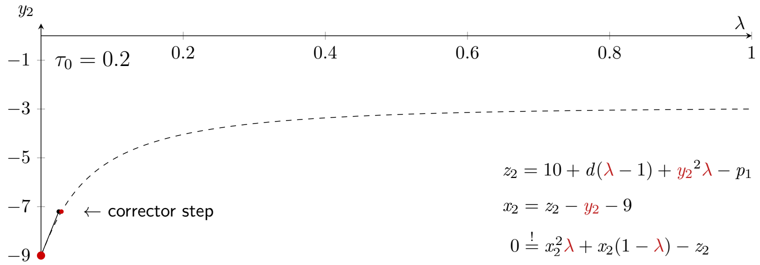
OM Runtime Flags

`-homTauMax=<value>`

to define the maximum step size τ.

Default: 10.0

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Check Step Acceptance and Adapt Step Size

Calculate the bending:

$$b = \frac{\|x_{i+1} - x^{\#}_{i+1}\|}{\|x^{\#}_{i+1} - x_i\|}$$

$b > homAdaptBend$

$\rightarrow$ Reject step!

$\rightarrow \tau := \frac{\tau}{homTauDecFac}$

$\rightarrow$ Perform predictor and corrector step again with new τ

$b < homAdaptBend$

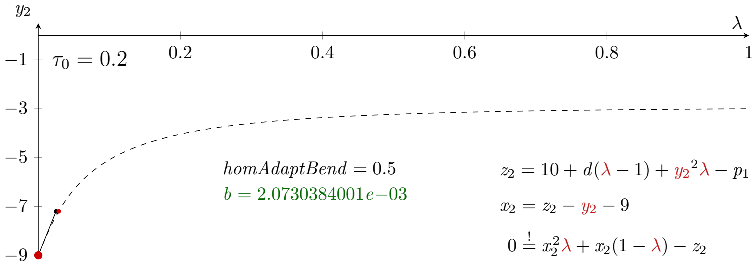
$\rightarrow$ Accept step!

$\rightarrow b < \frac{homAdaptBend}{homTauIncThreshold}$

$\rightarrow \tau := \tau \cdot homTauIncFac$

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure: Check Step Acceptance and Adapt Step Size

Calculate the bending:

$$b = \frac{\|x_{i+1} - x^{\#}_{i+1}\|}{\|x^{\#}_{i+1} - x_i\|}$$

$b > homAdaptBend$

$\rightarrow$ Reject step!

$\rightarrow \tau := \frac{\tau}{homTauDecFac}$

$\rightarrow$ Perform predictor and corrector step again with new τ

$b < homAdaptBend$

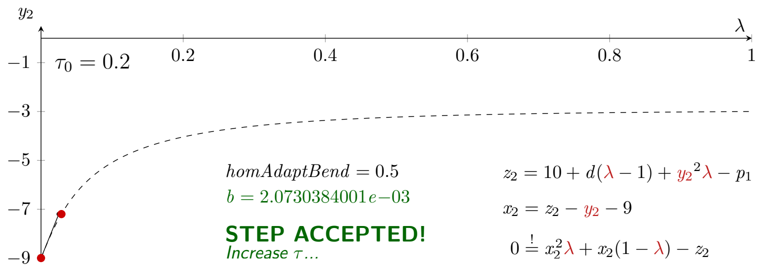
$\rightarrow$ Accept step!

$\rightarrow b < \frac{homAdaptBend}{homTauIncThreshold}$

$\rightarrow \tau := \tau \cdot homTauIncFac$

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Check Step Acceptance and Adapt Step Size

Calculate the bending:

$$b = \frac{\|\underline{x}_{i+1} - \underline{x}^{\#}_{i+1}\|}{\|\underline{x}^{\#}_{i+1} - \underline{x}_i\|}$$

$b > homAdaptBend$

- $\rightarrow$ Reject step!
- $\rightarrow \tau := \dfrac{\tau}{homTauDecFac}$
- $\rightarrow$ Perform predictor and corrector step again with new τ

$b < homAdaptBend$

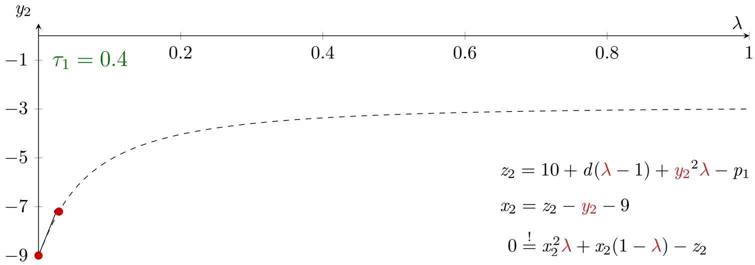
- $\rightarrow$ Accept step!
- $\rightarrow b < \dfrac{homAdaptBend}{homTauIncThreshold}$
  - $\rightarrow \tau := \tau \cdot homTauIncFac$

Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure

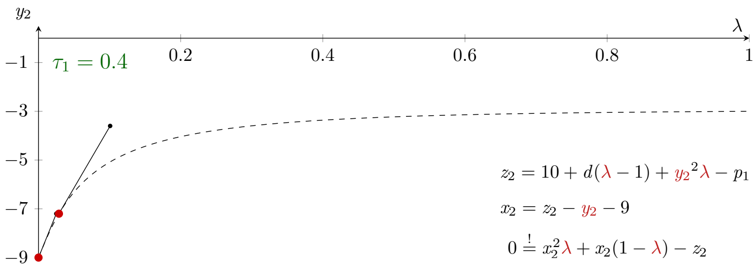
Perform further predictor and corrector steps...

Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)

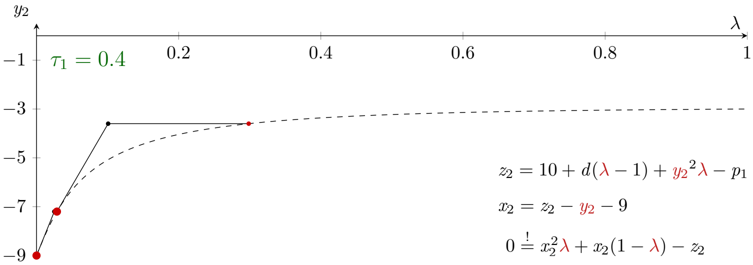


Procedure

Perform further predictor and corrector steps...

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

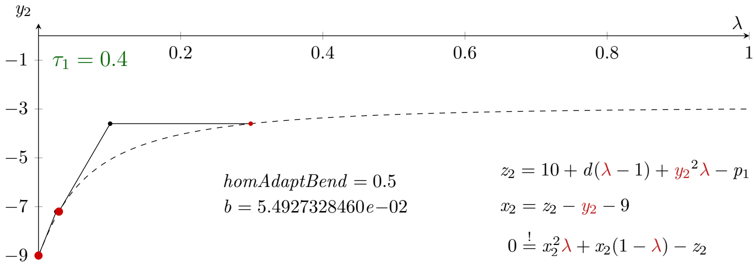
Procedure

Perform further predictor and corrector steps...

FH Bielefeld University of Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)

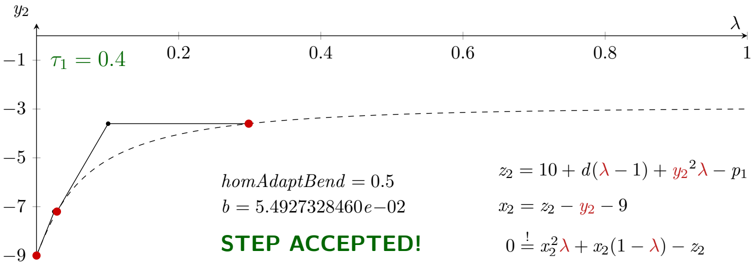


Procedure

Perform further predictor and corrector steps...

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$homAdaptBend = 0.5$

$b = 5.4927328460e{-}02$

STEP ACCEPTED!

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1 - \lambda) - z_2$$

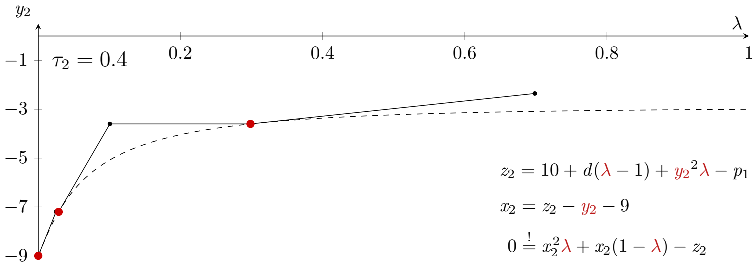
Procedure

Perform further predictor and corrector steps...

FH Bielefeld University of Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



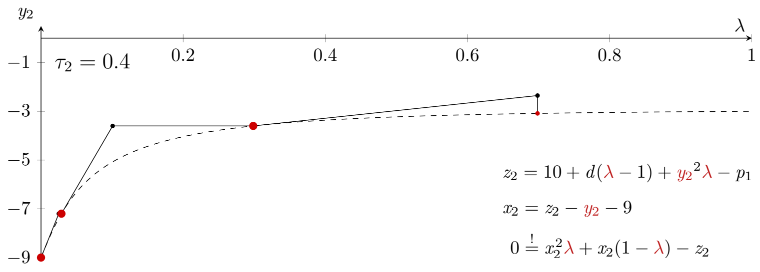
Procedure

Perform further predictor and corrector steps...

FH Bielefeld University of Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



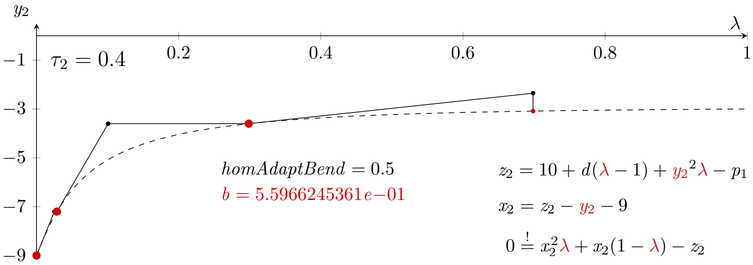
Procedure

Perform further predictor and corrector steps...

FH Bielefeld University of Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

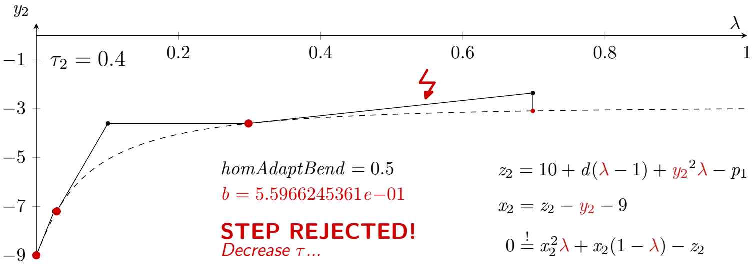
Procedure

Perform further predictor and corrector steps...

FH Bielefeld
University of
Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$homAdaptBend = 0.5$

$b = 5.5966245361e{-}01$

STEP REJECTED!

Decrease τ...

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

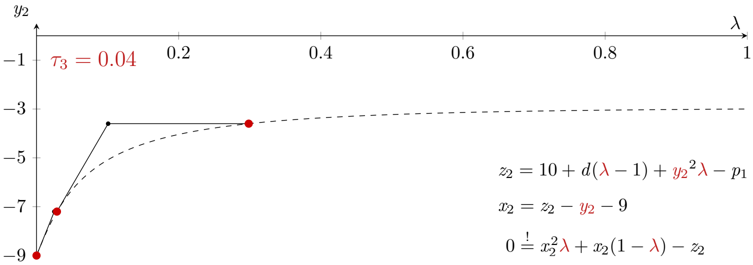
$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure

Perform further predictor and corrector steps...

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

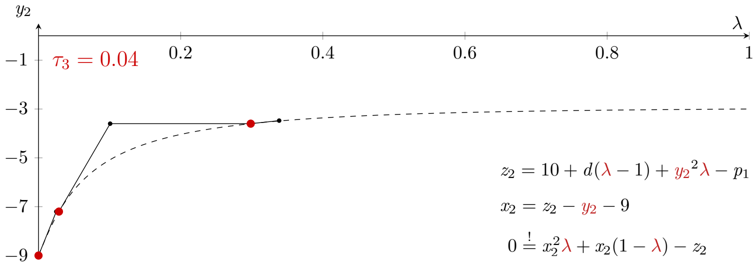
$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure

Perform further predictor and corrector steps...

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$

$x_2 = z_2 - y_2 - 9$

$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$

Procedure

Perform further predictor and corrector steps...

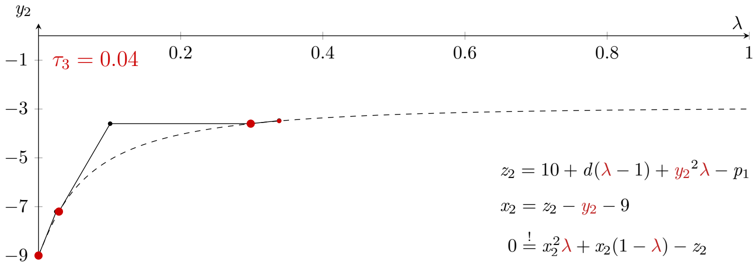
Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

FH Bielefeld
University of
Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

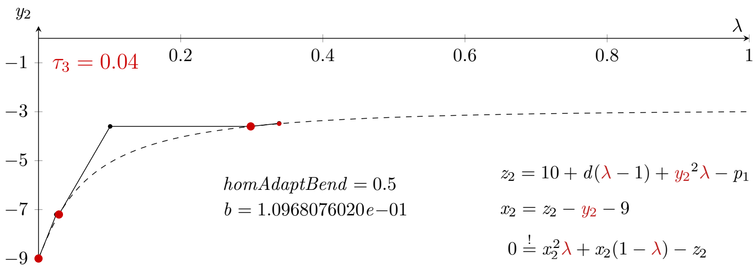
Procedure

Perform further predictor and corrector steps...

FH Bielefeld
University of
Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure

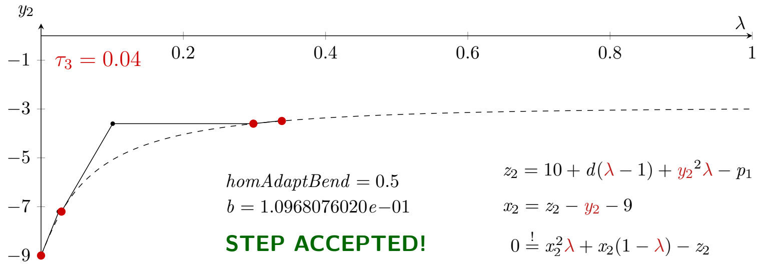
Perform further predictor and corrector steps...

Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$homAdaptBend = 0.5$

$b = 1.0968076020e{-}01$

STEP ACCEPTED!

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

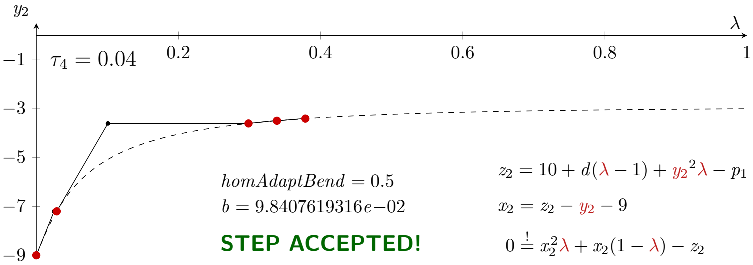
Procedure

Perform further predictor and corrector steps...

FH Bielefeld
University of Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$homAdaptBend = 0.5$

$b = 9.8407619316e{-}02$

STEP ACCEPTED!

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure

Perform further predictor and corrector steps...

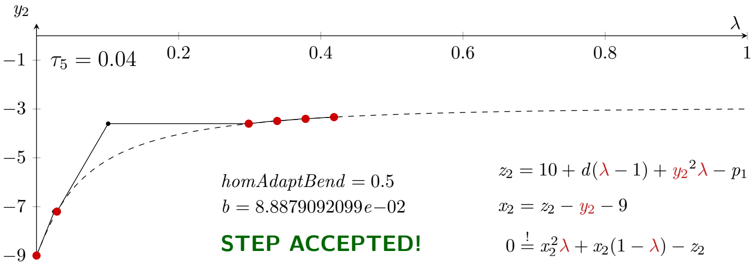
Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

FH Bielefeld
University of
Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$homAdaptBend = 0.5$

$b = 8.8879092099e{-}02$

STEP ACCEPTED!

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

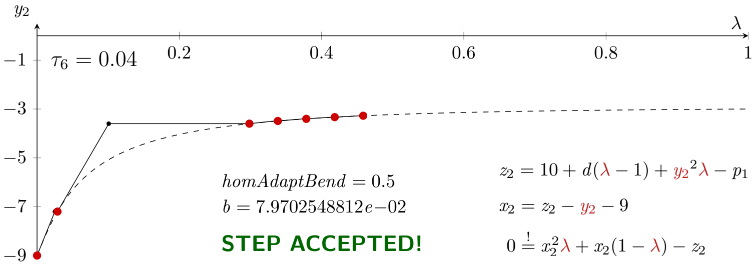
Procedure

Perform further predictor and corrector steps...

FH Bielefeld University of Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$homAdaptBend = 0.5$

$b = 7.9702548812e{-}02$

STEP ACCEPTED!

$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$

$x_2 = z_2 - y_2 - 9$

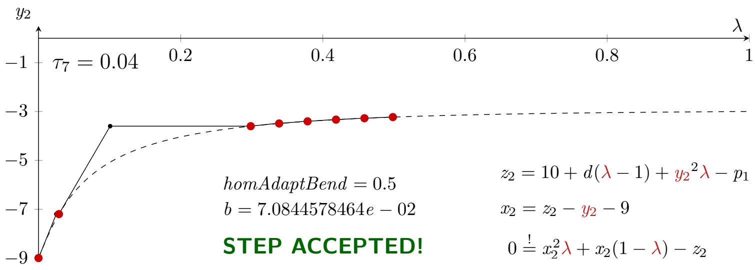
$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$

Procedure

Perform further predictor and corrector steps...

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



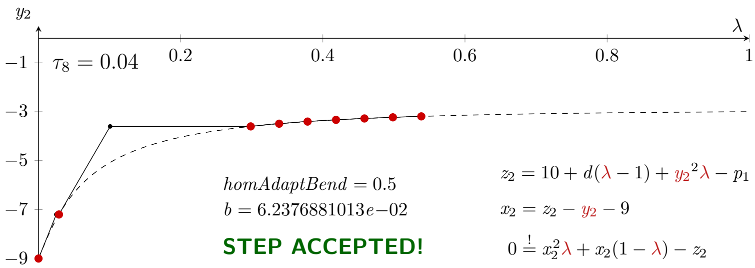
Procedure

Perform further predictor and corrector steps...

FH Bielefeld
University of
Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)

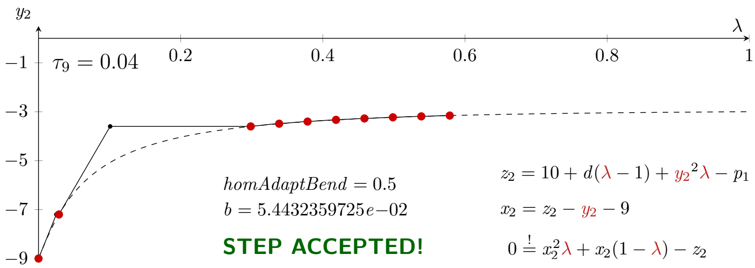


Procedure

Perform further predictor and corrector steps...

OM Runtime Flags

Homotopy Path Calculation `(omc homotopy4.mos --homotopyApproach=adaptiveLocal)`



$homAdaptBend = 0.5$

$b = 5.4432359725e{-}02$

STEP ACCEPTED!

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure

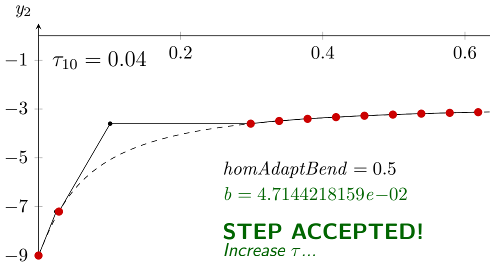
Perform further predictor and corrector steps...

Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

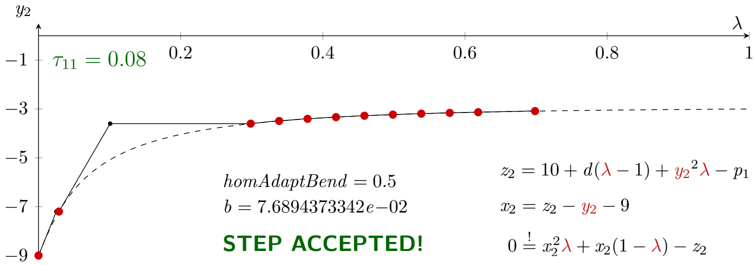
Procedure

Perform further predictor and corrector steps...

FH Bielefeld
University of Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$homAdaptBend = 0.5$

$b = 7.6894373342e{-}02$

STEP ACCEPTED!

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

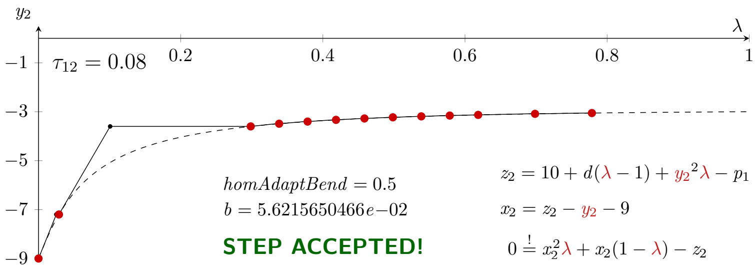
$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

Procedure

Perform further predictor and corrector steps...

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$homAdaptBend = 0.5$

$b = 5.6215650466e{-02}$

STEP ACCEPTED!

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

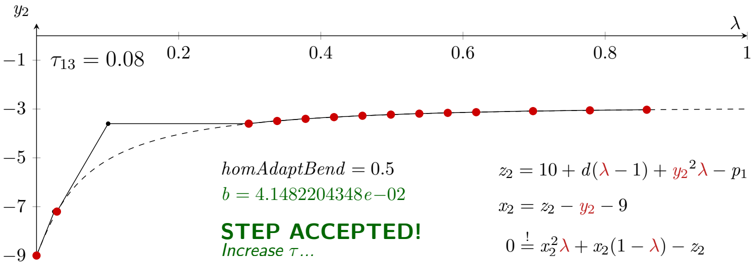
Procedure

Perform further predictor and corrector steps...

FH Bielefeld University of Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



$homAdaptBend = 0.5$

$b = 4.1482204348e{-}02$

STEP ACCEPTED!

Increase τ...

$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$

$x_2 = z_2 - y_2 - 9$

$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$

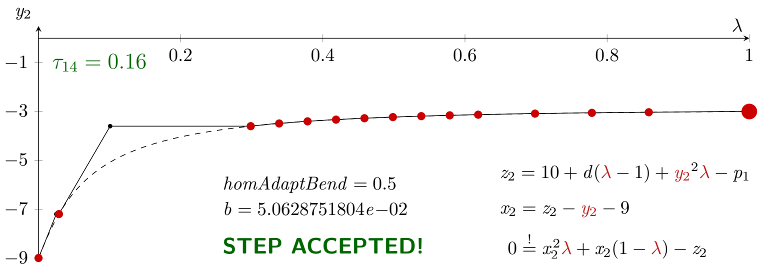
Procedure

Perform further predictor and corrector steps...

FH Bielefeld
University of
Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



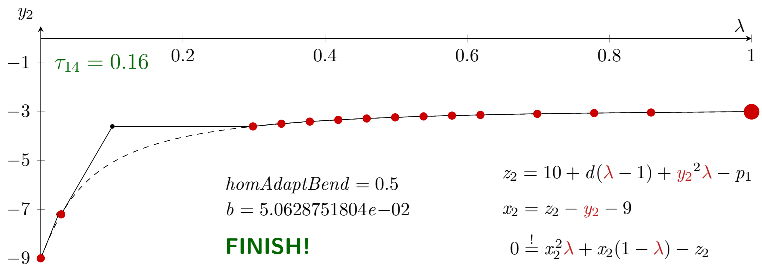
Procedure

Perform further predictor and corrector steps...

FH Bielefeld
University of
Applied Sciences

OM Runtime Flags

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Finish or Break

- Finish:
  - The system is solved successfully for $\lambda = 1$.
- Break:
  - $homMaxLambdaSteps$ is reached
  - $homMaxTries$ for one λ is reached
  - $\tau = homTauMin$ and current step fails
  - $\lambda < -1$

Homotopy Approaches in OM

Homotopy with adaptive step size: The Homotopy Algorithm

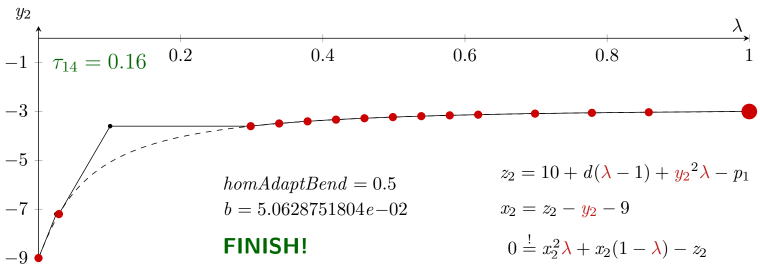
OM Runtime Flags

`-homMaxLambdaSteps =<value>`

to define the maximum number of homotopy λ-steps as a termination criterion.

Default: 100·sizeOfSystem

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Finish or Break

- Finish:
  - The system is solved successfully for $\lambda = 1$.
- Break:
  - *homMaxLambdaSteps* is reached
  - *homMaxTries* for one λ is reached
  - $\tau = homTauMin$ and current step fails
  - $\lambda < -1$

FH Bielefeld
University of Applied Sciences

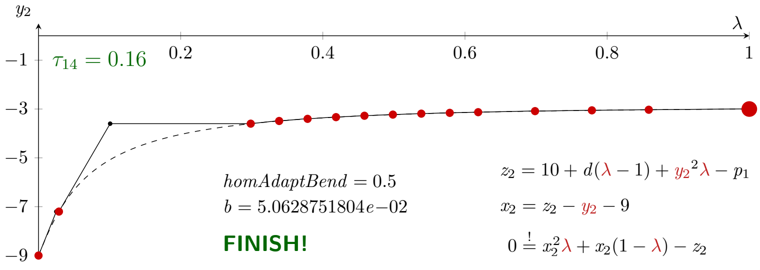
OM Runtime Flags

`-homMaxTries=<value>`

to define the maximum number of predictor/corrector step tries for one λ-step as a termination criterion.

Default: 10

Homotopy Path Calculation (omc homotopy4.mos --homotopyApproach=adaptiveLocal)



Procedure: Finish or Break

- Finish:
  - The system is solved successfully for $\lambda = 1$.
- Break:
  - $homMaxLambdaSteps$ is reached
  - $homMaxTries$ for one λ is reached
  - $\tau = homTauMin$ and current step fails
  - $\lambda < -1$

Characteristic Features of the Homotopy Algorithm

Start Direction

- λ does not have to be monotonically nondecreasing
- The continuation direction is handled by the algorithm

$\rightarrow$ Problem: For $\lambda = 0$ the correct start direction is unknown

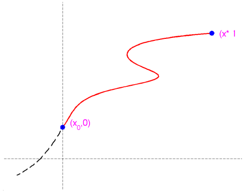
Characteristic Features of the Homotopy Algorithm

Start Direction

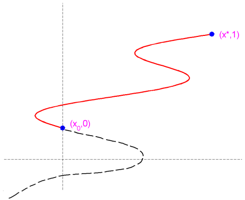
- λ does not have to be monotonically nondecreasing
- The continuation direction is handled by the algorithm

$\rightarrow$ Problem: For $\lambda = 0$ the correct start direction is unknown

Use positive start direction:



Use negative start direction:



Characteristic Features of the Homotopy Algorithm

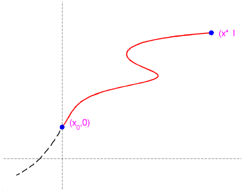
Start Direction

- λ does not have to be monotonically nondecreasing
- The continuation direction is handled by the algorithm

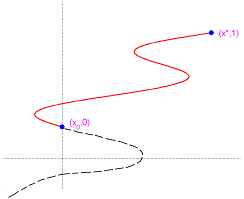
$\rightarrow$ Problem: For $\lambda = 0$ the correct start direction is unknown

- Homotopy algorithm first tries the positive direction
  - If no convergence, try negative direction

Use positive start direction:



Use negative start direction:



Characteristic Features of the Homotopy Algorithm

Start Direction

- λ does not have to be monotonically nondecreasing
- The continuation direction is handled by the algorithm

→ Problem: For $\lambda = 0$ the correct start direction is unknown

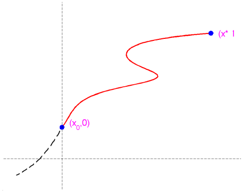
- Homotopy algorithm first tries the positive direction
  - If no convergence, try negative direction

OM Runtime Flags

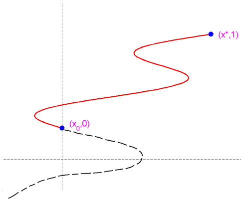
`-homNegStartDir`

to start the continuation method in the negative direction first.
If no convergence, the positive direction is tried.

Use positive start direction:

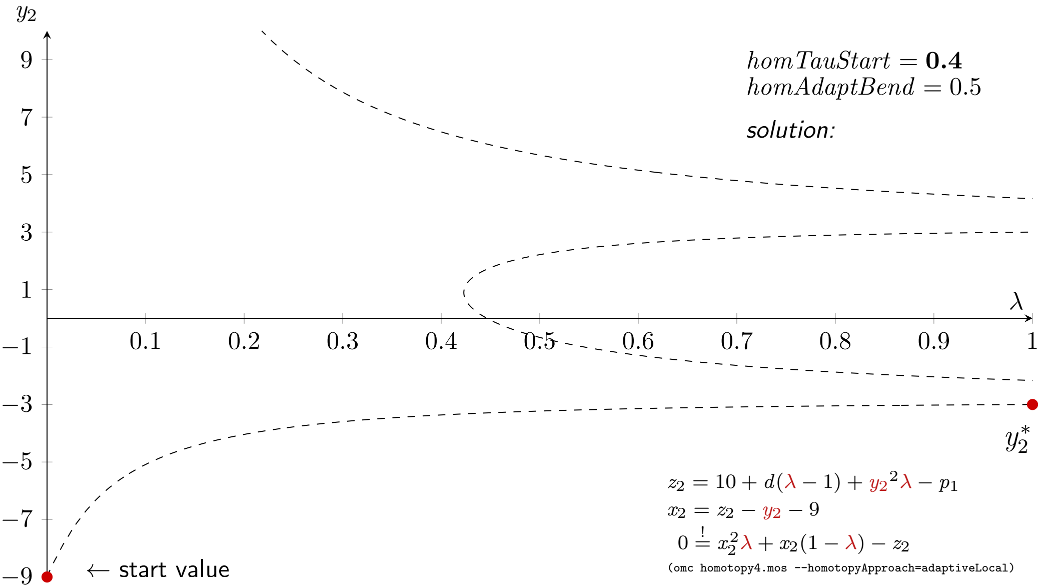


Use negative start direction:



Small changes to the homotopy parameters can lead to leaving the correct path:

Small changes to the homotopy parameters can lead to leaving the correct path:



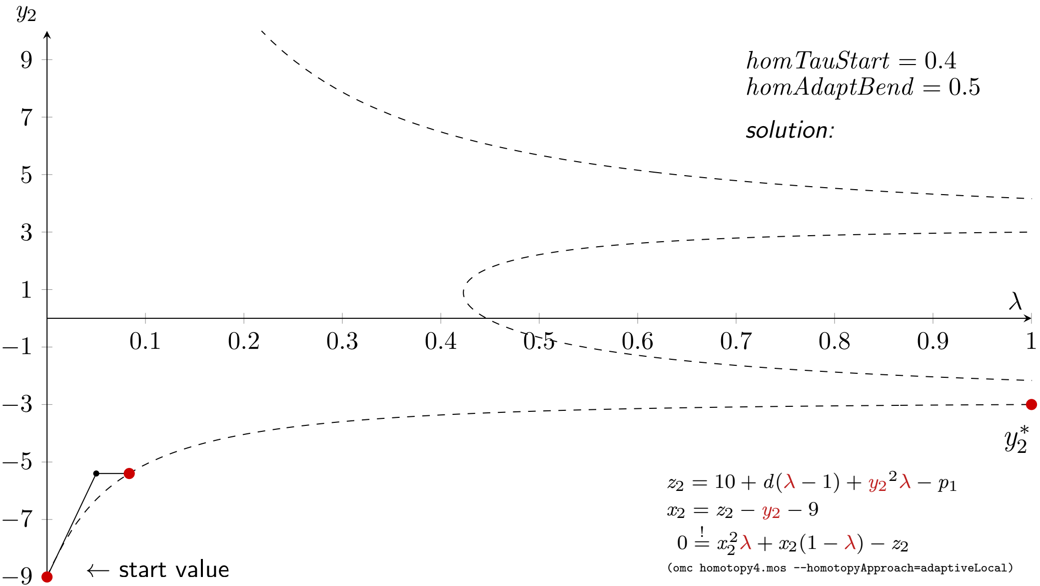
$homTauStart = \mathbf{0.4}$
$homAdaptBend = 0.5$

solution:

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

(omc homotopy4.mos --homotopyApproach=adaptiveLocal)

Small changes to the homotopy parameters can lead to leaving the correct path:



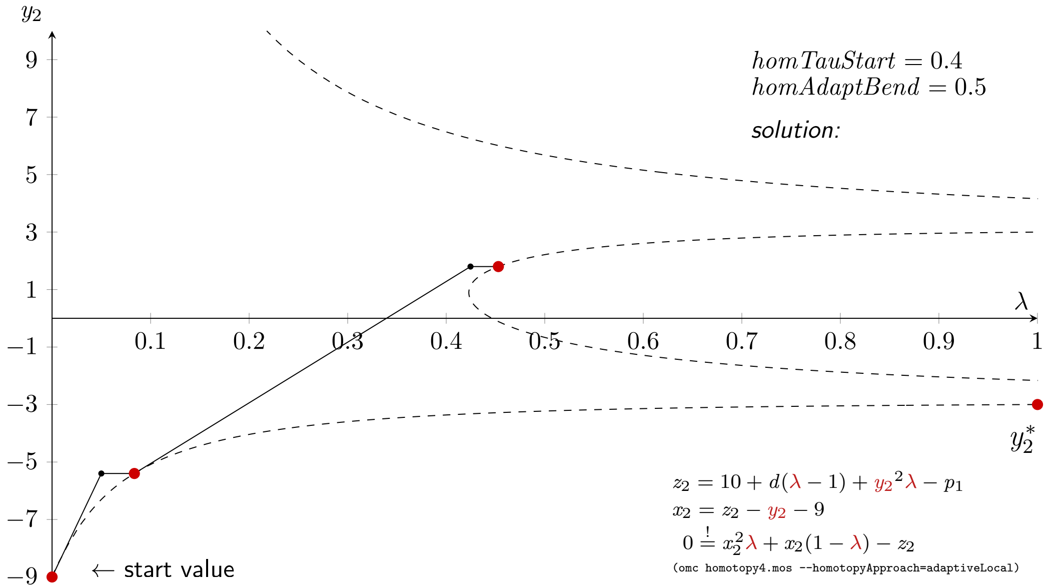
$homTauStart = 0.4$
$homAdaptBend = 0.5$

solution:

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

`(omc homotopy4.mos --homotopyApproach=adaptiveLocal)`

Small changes to the homotopy parameters can lead to leaving the correct path:



$homTauStart = 0.4$
$homAdaptBend = 0.5$

solution:

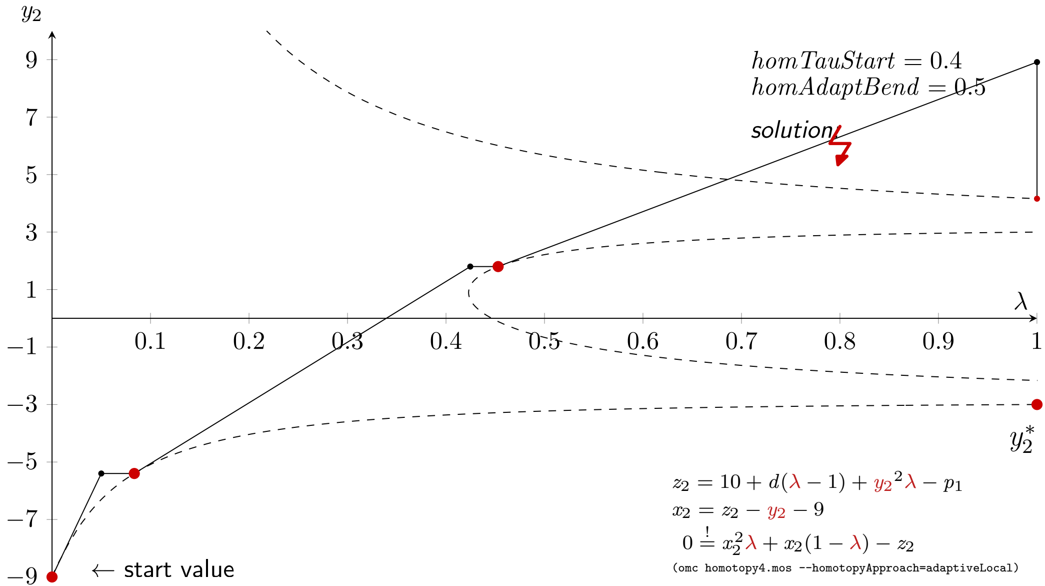
$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

(omc homotopy4.mos --homotopyApproach=adaptiveLocal)

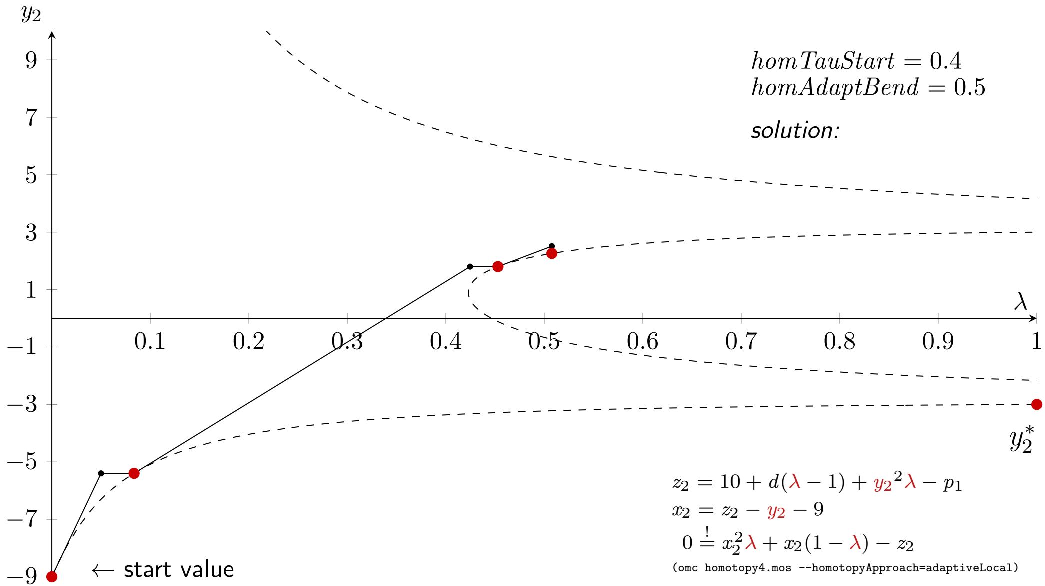
Characteristic Features of the Homotopy Algorithm

Sensitivity to the Homotopy Parameters

Small changes to the homotopy parameters can lead to leaving the correct path:



Small changes to the homotopy parameters can lead to leaving the correct path:



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

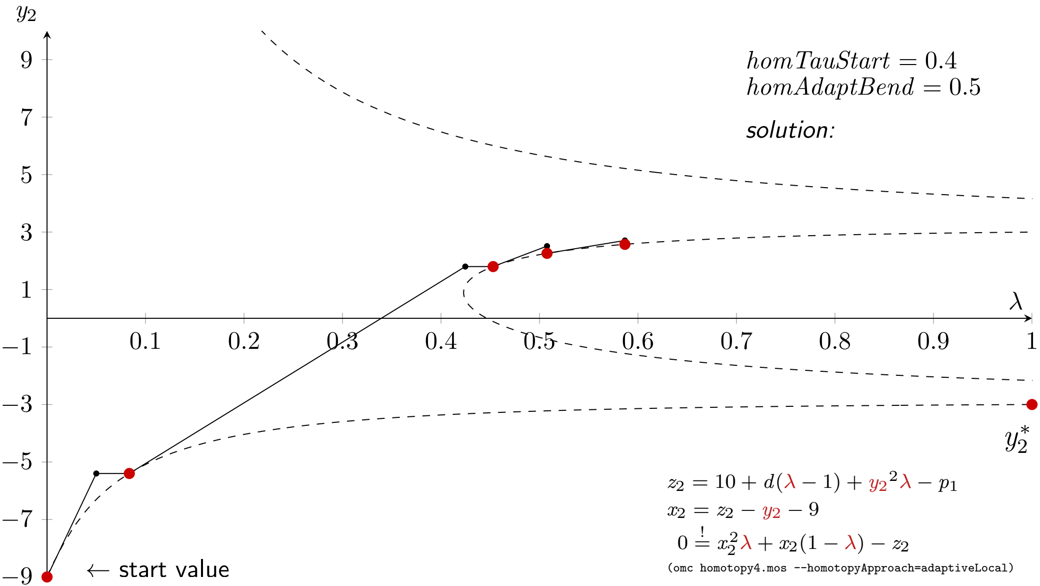
$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1 - \lambda) - z_2$$

(omc homotopy4.mos --homotopyApproach=adaptiveLocal)

Characteristic Features of the Homotopy Algorithm

Sensitivity to the Homotopy Parameters

Small changes to the homotopy parameters can lead to leaving the correct path:



$homTauStart = 0.4$
$homAdaptBend = 0.5$

solution:

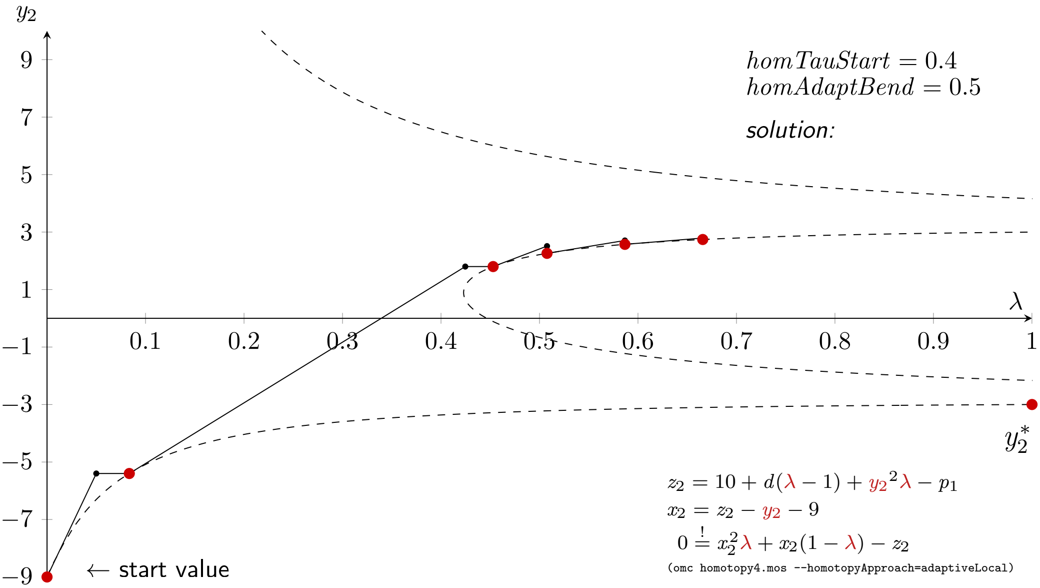
$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

`(omc homotopy4.mos --homotopyApproach=adaptiveLocal)`

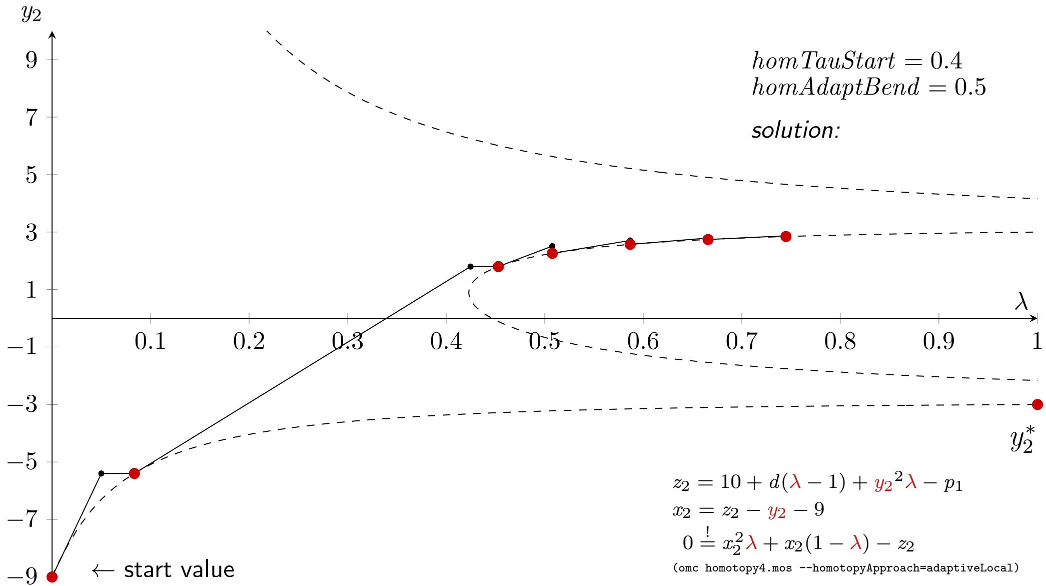
Characteristic Features of the Homotopy Algorithm

Sensitivity to the Homotopy Parameters

Small changes to the homotopy parameters can lead to leaving the correct path:



Small changes to the homotopy parameters can lead to leaving the correct path:



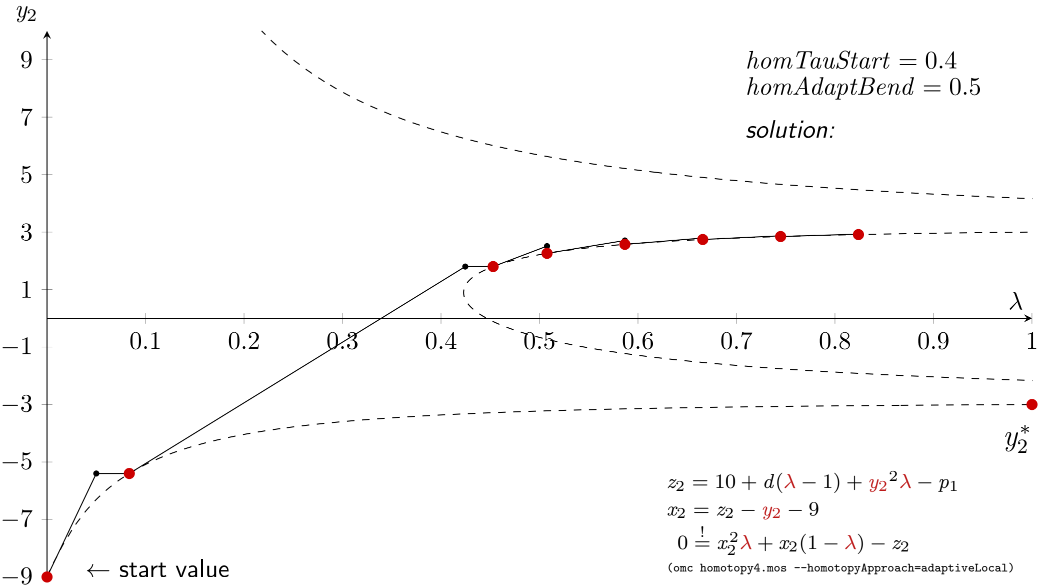
$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

(omc homotopy4.mos --homotopyApproach=adaptiveLocal)

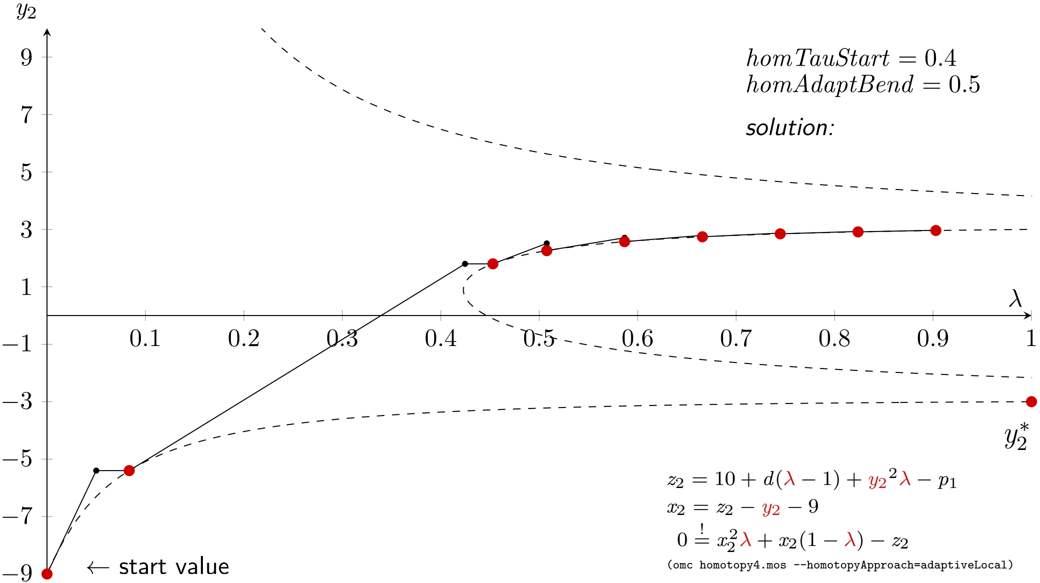
Characteristic Features of the Homotopy Algorithm

Sensitivity to the Homotopy Parameters

Small changes to the homotopy parameters can lead to leaving the correct path:



Small changes to the homotopy parameters can lead to leaving the correct path:



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

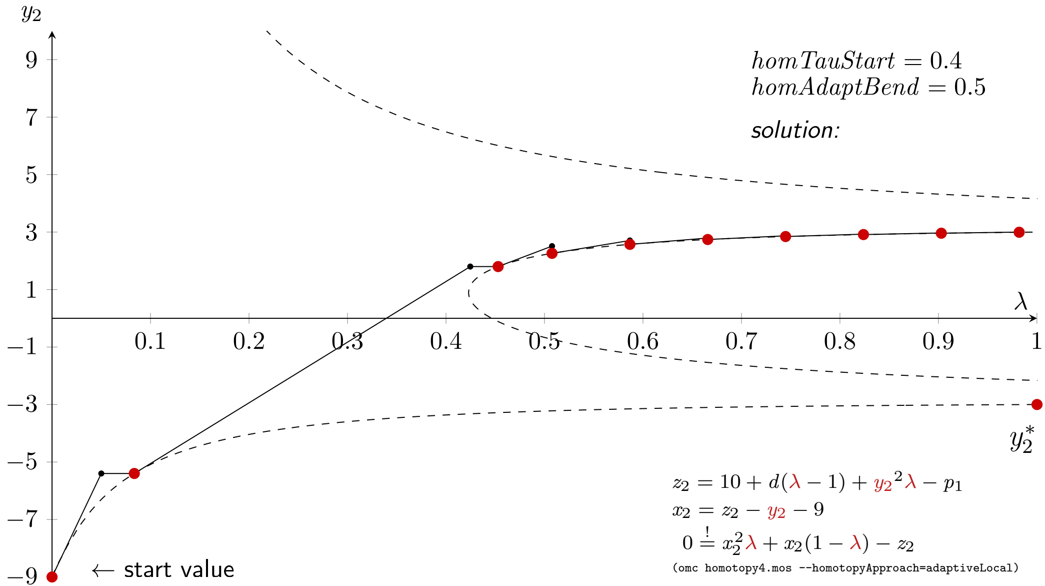
$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

`(omc homotopy4.mos --homotopyApproach=adaptiveLocal)`

Characteristic Features of the Homotopy Algorithm

Sensitivity to the Homotopy Parameters

Small changes to the homotopy parameters can lead to leaving the correct path:



$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

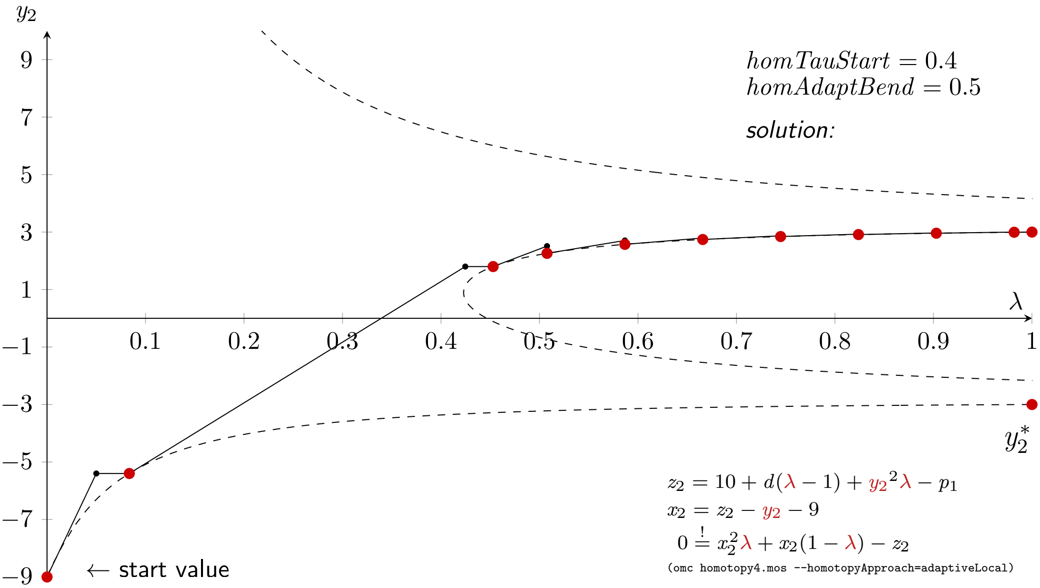
$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

```
(omc homotopy4.mos --homotopyApproach=adaptiveLocal)
```

Characteristic Features of the Homotopy Algorithm

Sensitivity to the Homotopy Parameters

Small changes to the homotopy parameters can lead to leaving the correct path:



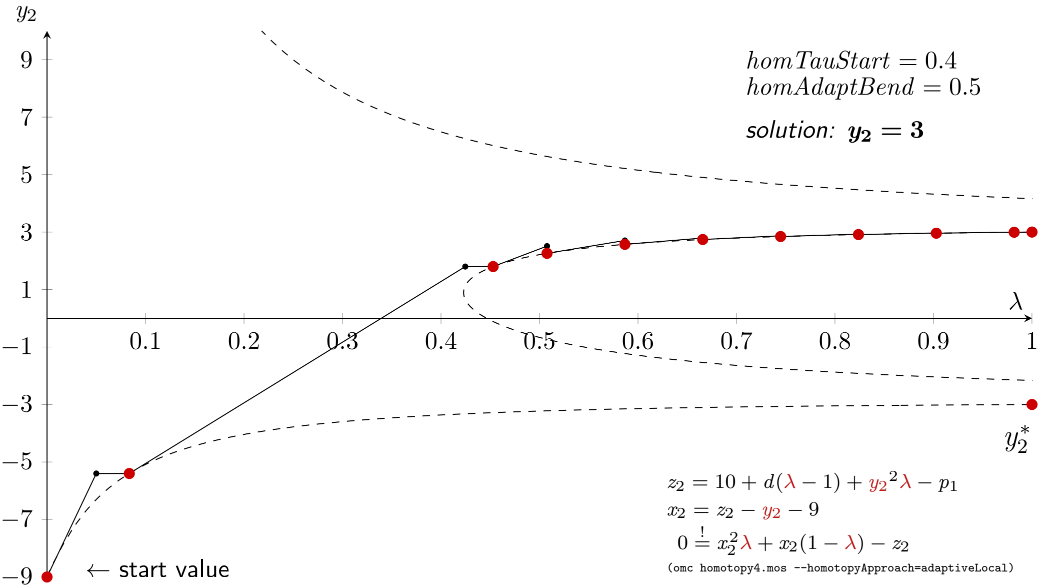
$homTauStart = 0.4$
$homAdaptBend = 0.5$

solution:

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

`(omc homotopy4.mos --homotopyApproach=adaptiveLocal)`

Small changes to the homotopy parameters can lead to leaving the correct path:



$homTauStart = 0.4$
$homAdaptBend = 0.5$

solution: $\boldsymbol{y_2 = 3}$

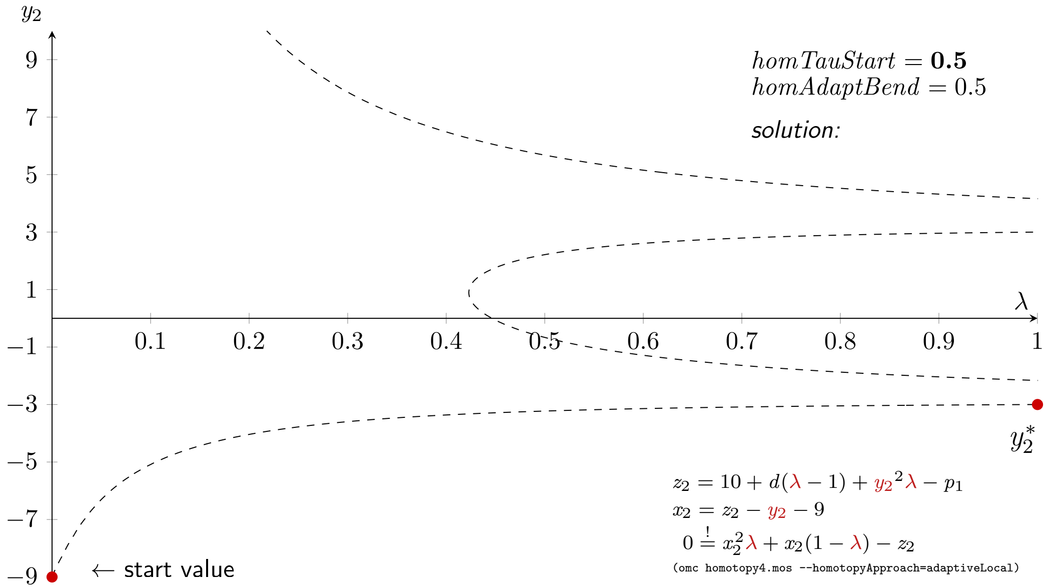
$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

(omc homotopy4.mos --homotopyApproach=adaptiveLocal)

Small changes to the homotopy parameters can lead to leaving the correct path:



$homTauStart = \mathbf{0.5}$
$homAdaptBend = 0.5$

solution:

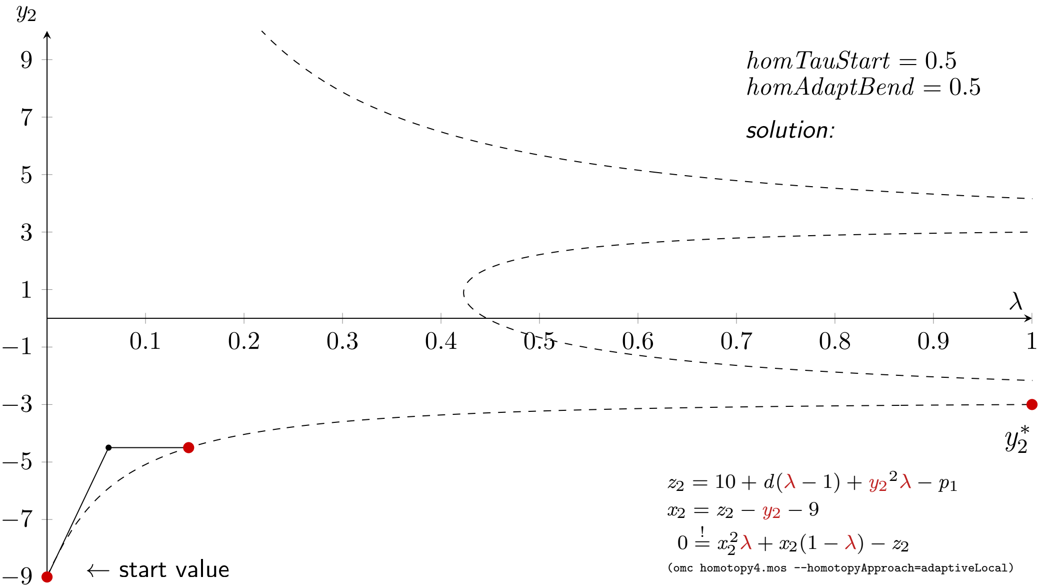
$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

(omc homotopy4.mos --homotopyApproach=adaptiveLocal)

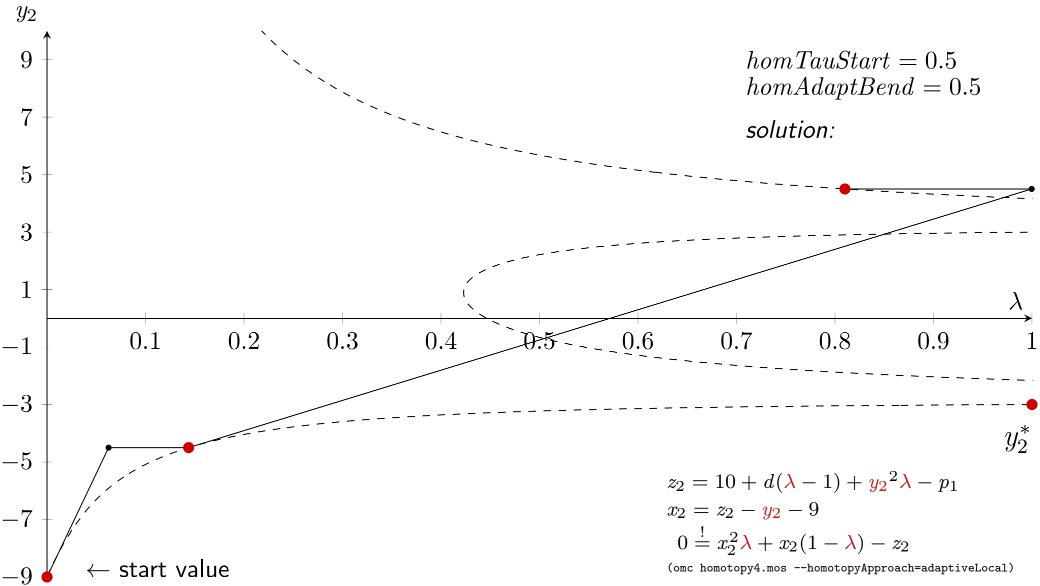
Characteristic Features of the Homotopy Algorithm

Sensitivity to the Homotopy Parameters

Small changes to the homotopy parameters can lead to leaving the correct path:



Small changes to the homotopy parameters can lead to leaving the correct path:



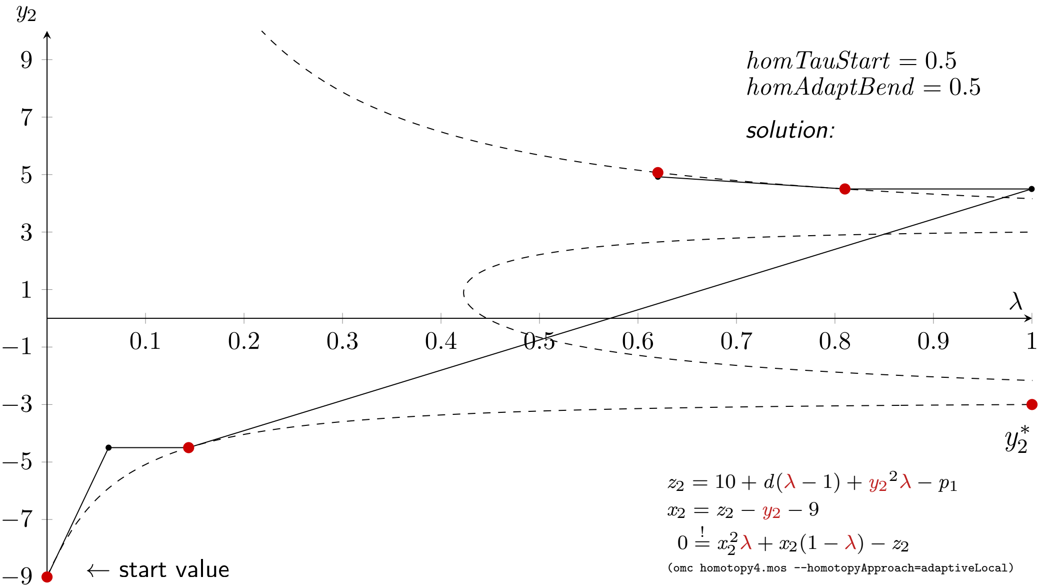
$homTauStart = 0.5$
$homAdaptBend = 0.5$

solution:

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \overset{!}{=} x_2^2\lambda + x_2(1 - \lambda) - z_2$$

`(omc homotopy4.mos --homotopyApproach=adaptiveLocal)`

Small changes to the homotopy parameters can lead to leaving the correct path:



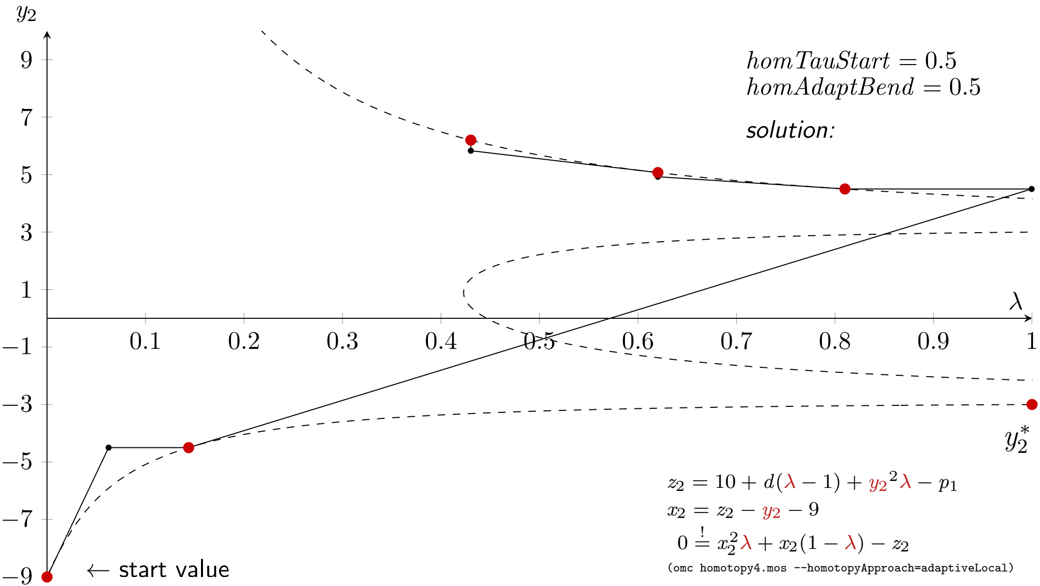
$homTauStart = 0.5$
$homAdaptBend = 0.5$

solution:

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1 - \lambda) - z_2$$

(omc homotopy4.mos --homotopyApproach=adaptiveLocal)

Small changes to the homotopy parameters can lead to leaving the correct path:



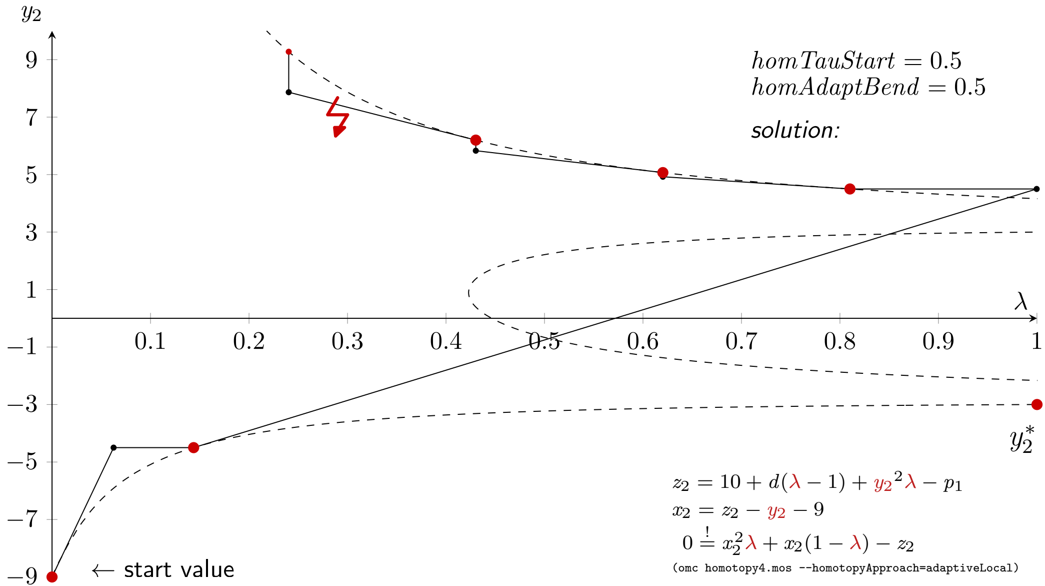
$homTauStart = 0.5$
$homAdaptBend = 0.5$

solution:

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

(omc homotopy4.mos --homotopyApproach=adaptiveLocal)

Small changes to the homotopy parameters can lead to leaving the correct path:



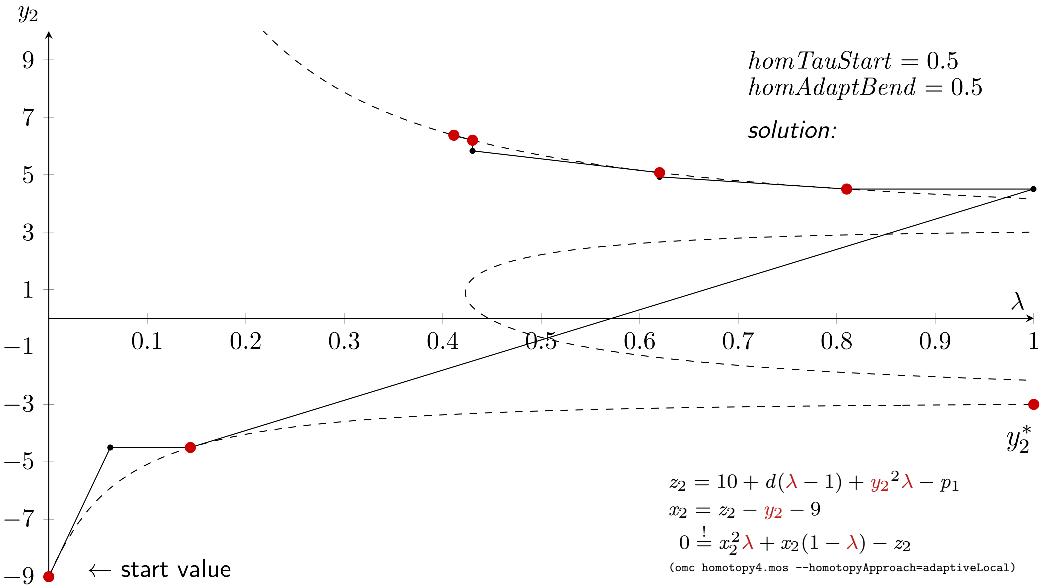
$homTauStart = 0.5$
$homAdaptBend = 0.5$

solution:

$$z_2 = 10 + d(\lambda - 1) + y_2{}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \overset{!}{=} x_2^2\lambda + x_2(1 - \lambda) - z_2$$

`(omc homotopy4.mos --homotopyApproach=adaptiveLocal)`

Small changes to the homotopy parameters can lead to leaving the correct path:



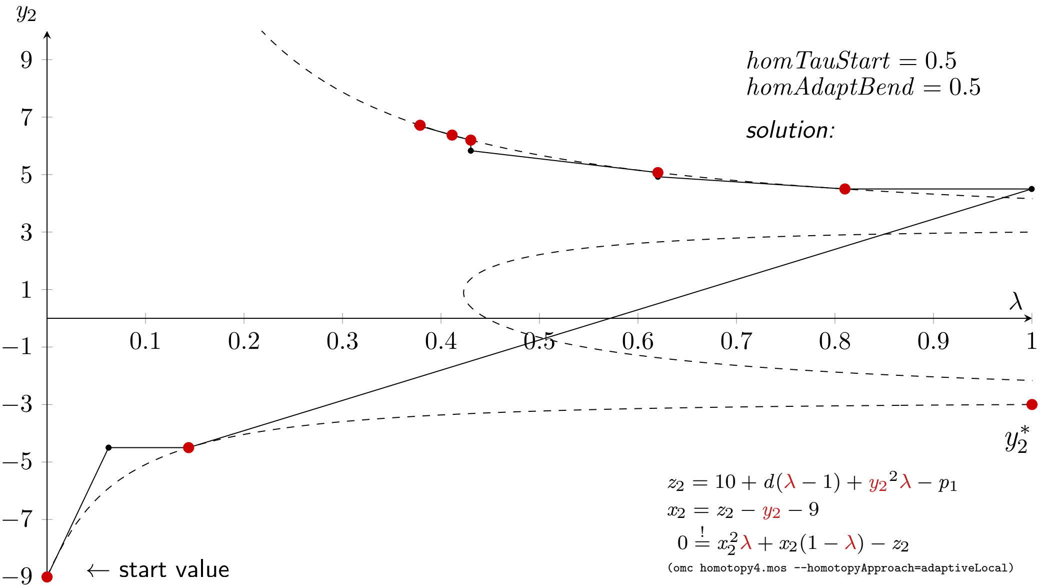
$homTauStart = 0.5$
$homAdaptBend = 0.5$

solution:

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

`(omc homotopy4.mos --homotopyApproach=adaptiveLocal)`

Small changes to the homotopy parameters can lead to leaving the correct path:



$homTauStart = 0.5$
$homAdaptBend = 0.5$

solution:

$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$

$x_2 = z_2 - y_2 - 9$

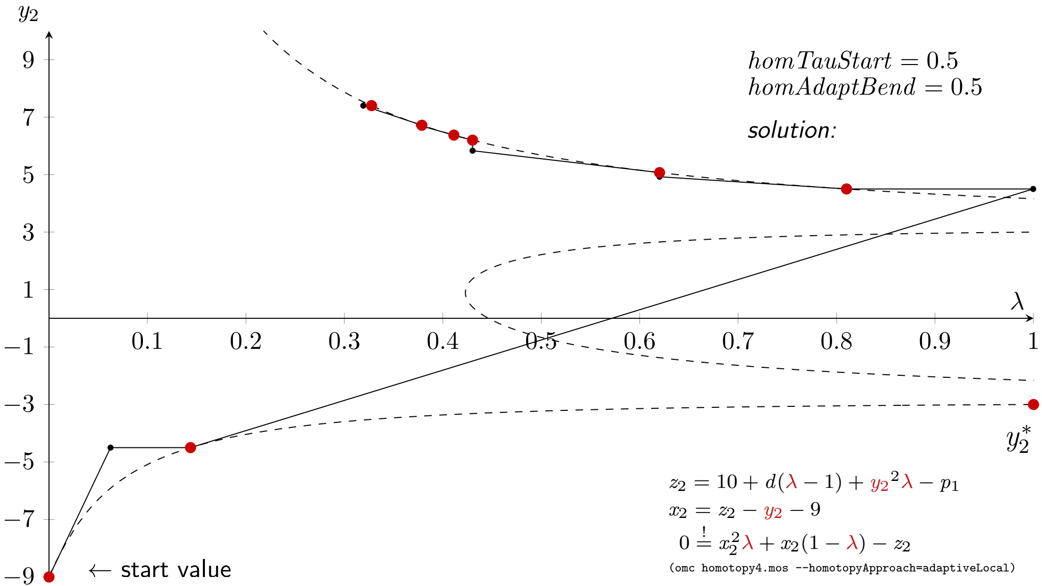
$0 \stackrel{!}{=} x_2^2\lambda + x_2(1 - \lambda) - z_2$

`(omc homotopy4.mos --homotopyApproach=adaptiveLocal)`

Characteristic Features of the Homotopy Algorithm

Sensitivity to the Homotopy Parameters

Small changes to the homotopy parameters can lead to leaving the correct path:



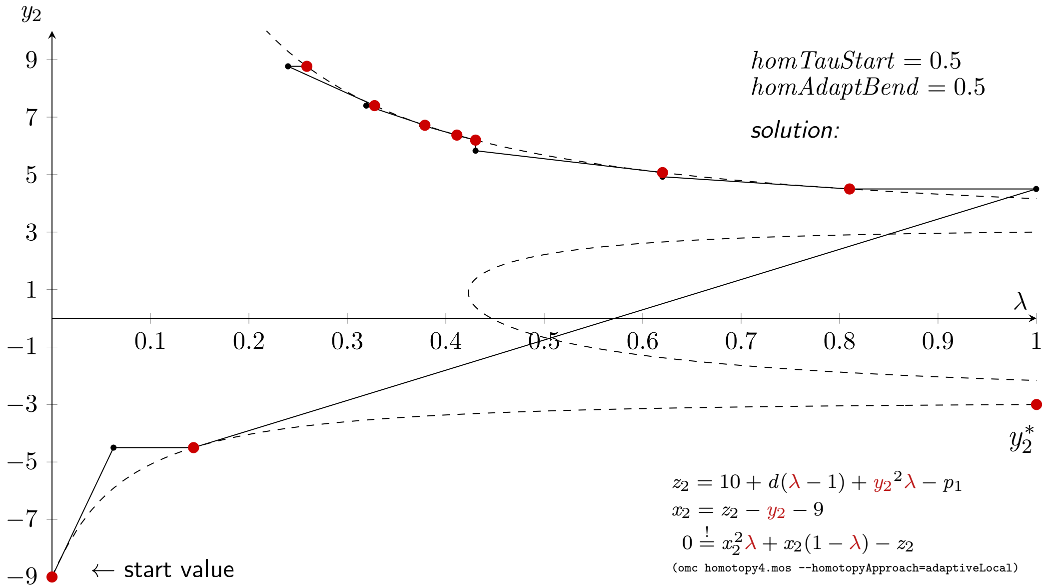
$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1 - \lambda) - z_2$$

`(omc homotopy4.mos --homotopyApproach=adaptiveLocal)`

Small changes to the homotopy parameters can lead to leaving the correct path:



$homTauStart = 0.5$
$homAdaptBend = 0.5$

solution:

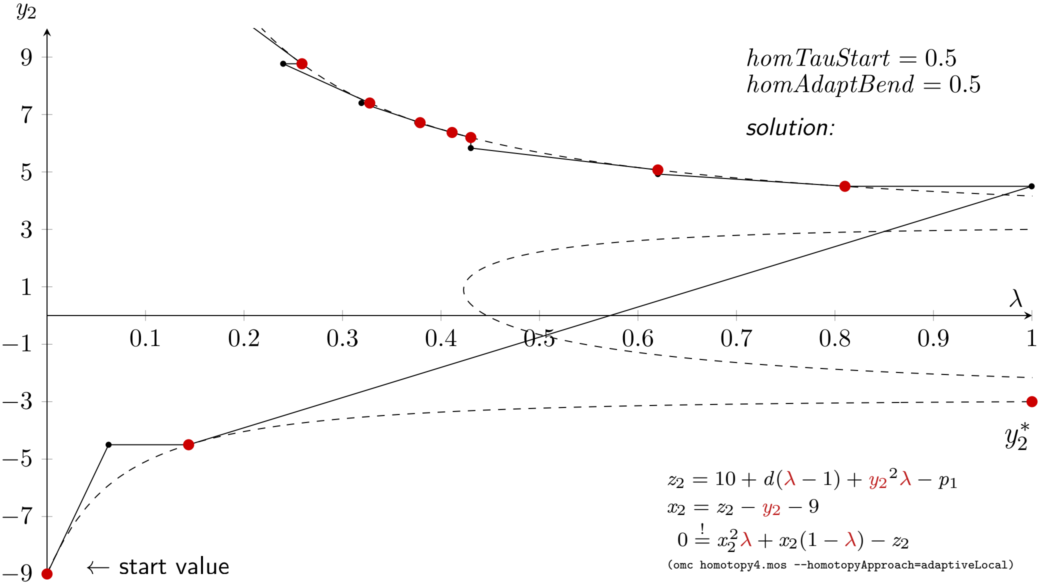
$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$

$x_2 = z_2 - y_2 - 9$

$0 \stackrel{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$

(`omc homotopy4.mos --homotopyApproach=adaptiveLocal`)

Small changes to the homotopy parameters can lead to leaving the correct path:



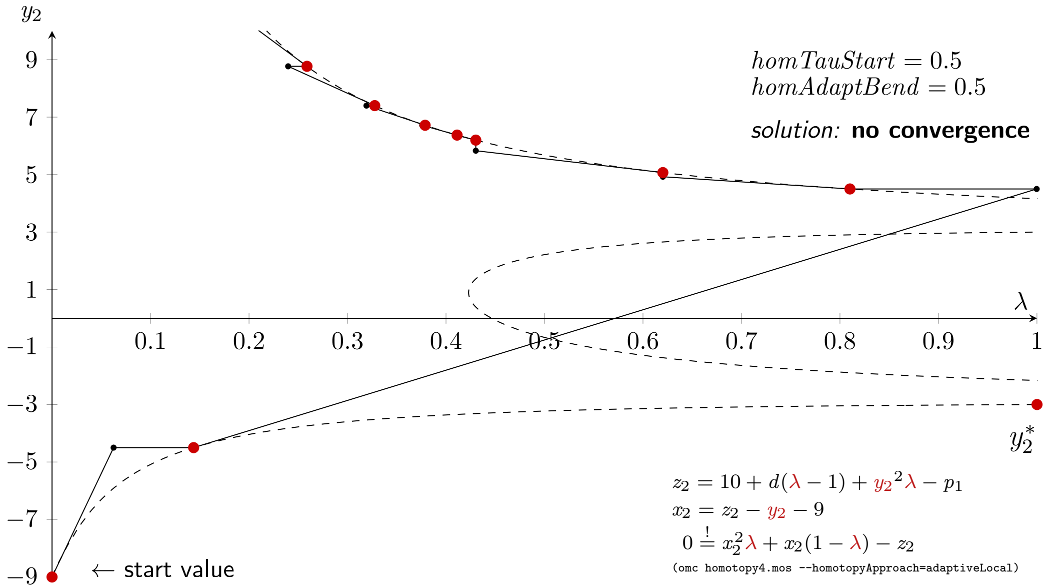
$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$

$$x_2 = z_2 - y_2 - 9$$

$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

(omc homotopy4.mos --homotopyApproach=adaptiveLocal)

Small changes to the homotopy parameters can lead to leaving the correct path:



$homTauStart = 0.5$
$homAdaptBend = 0.5$

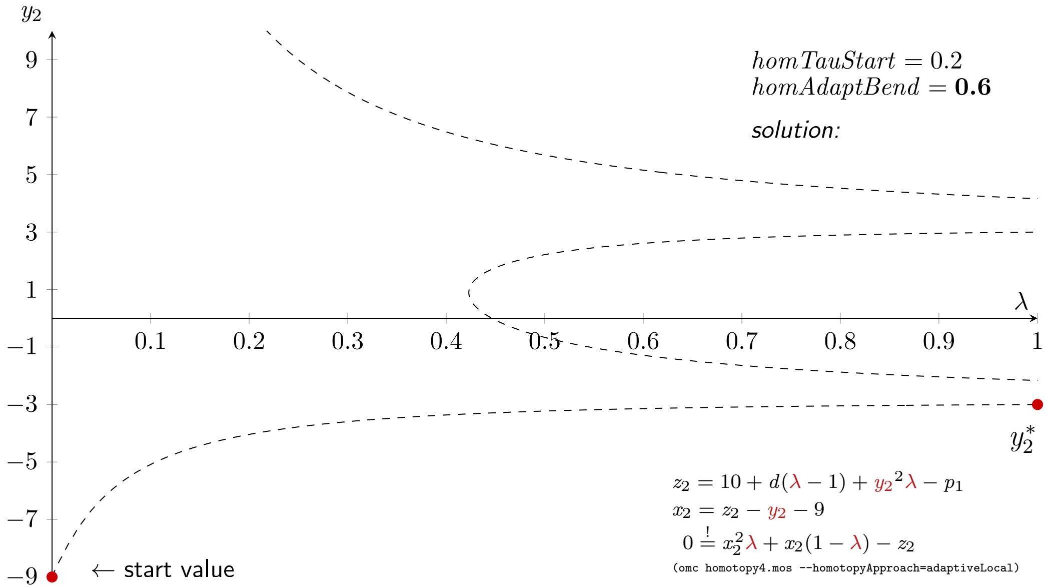
solution: **no convergence**

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \overset{!}{=} x_2^2\lambda + x_2(1 - \lambda) - z_2$$

(omc homotopy4.mos --homotopyApproach=adaptiveLocal)

FH Bielefeld
University of
Applied Sciences

Small changes to the homotopy parameters can lead to leaving the correct path:



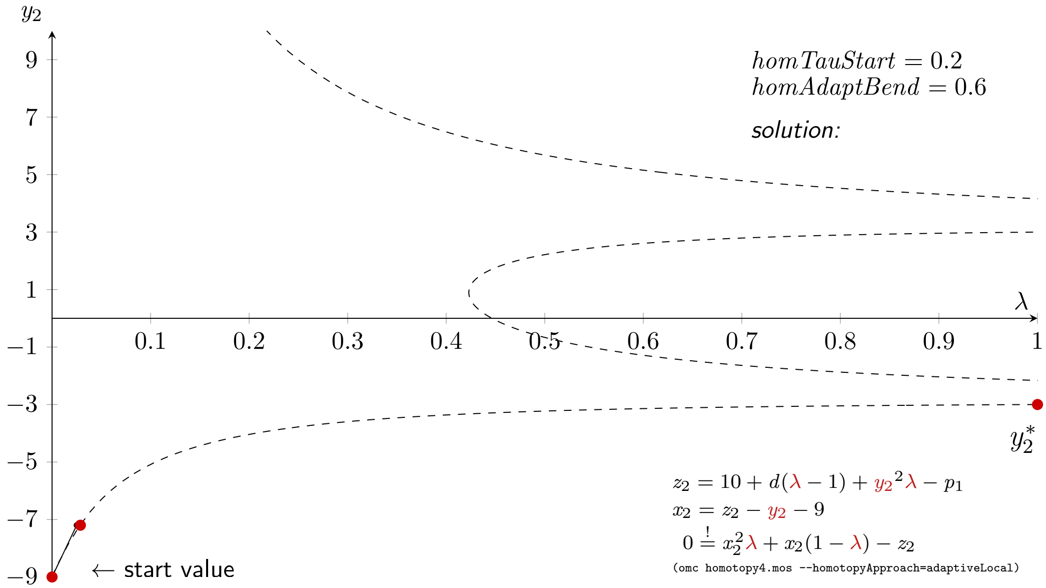
$$homTauStart = 0.2$$
$$homAdaptBend = \mathbf{0.6}$$

solution:

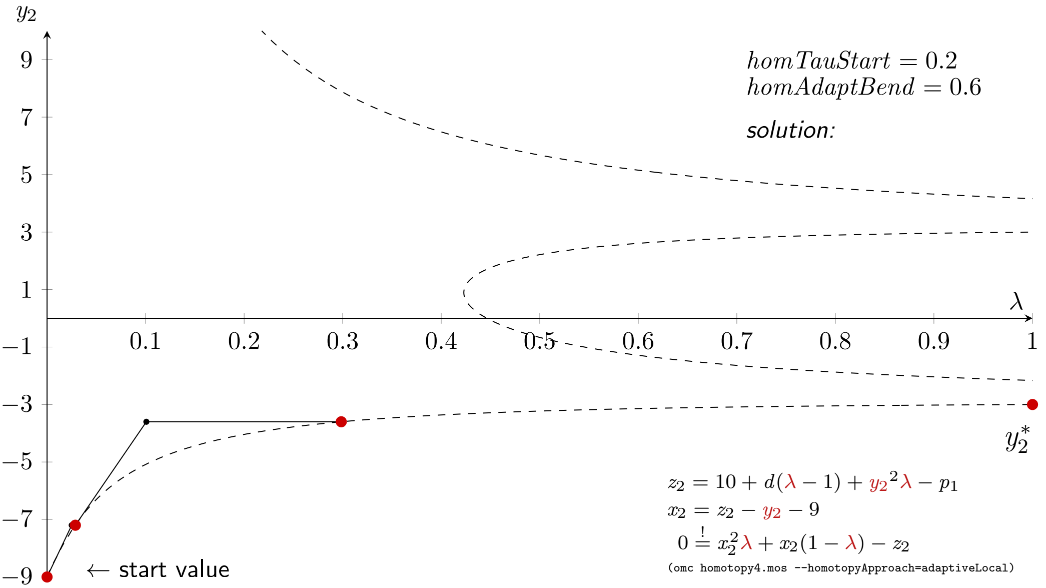
$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1 - \lambda) - z_2$$

`(omc homotopy4.mos --homotopyApproach=adaptiveLocal)`

Small changes to the homotopy parameters can lead to leaving the correct path:



Small changes to the homotopy parameters can lead to leaving the correct path:



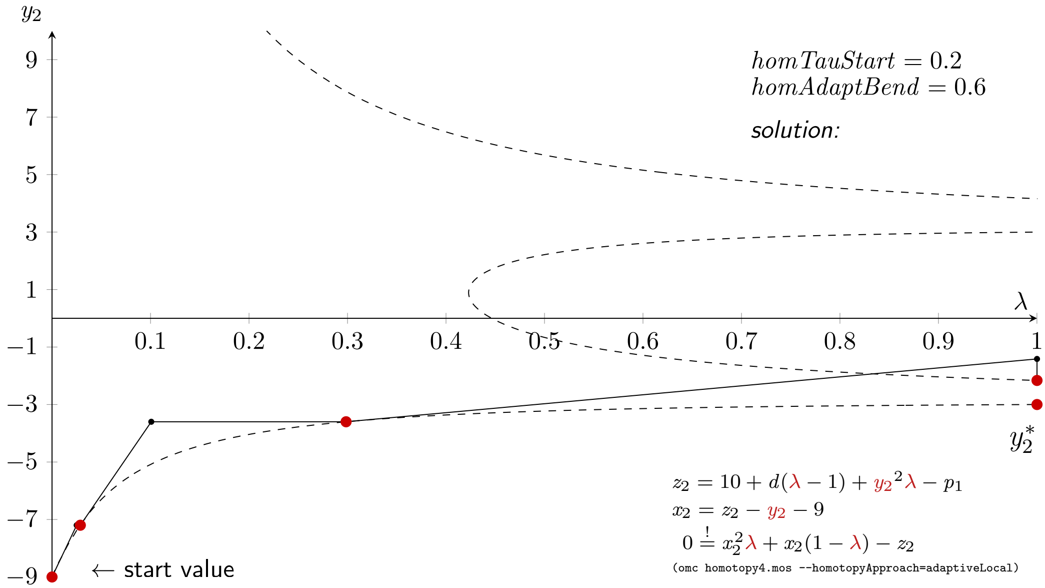
$homTauStart = 0.2$
$homAdaptBend = 0.6$

solution:

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \stackrel{!}{=} x_2^2\lambda + x_2(1 - \lambda) - z_2$$

(`omc homotopy4.mos --homotopyApproach=adaptiveLocal`)

Small changes to the homotopy parameters can lead to leaving the correct path:



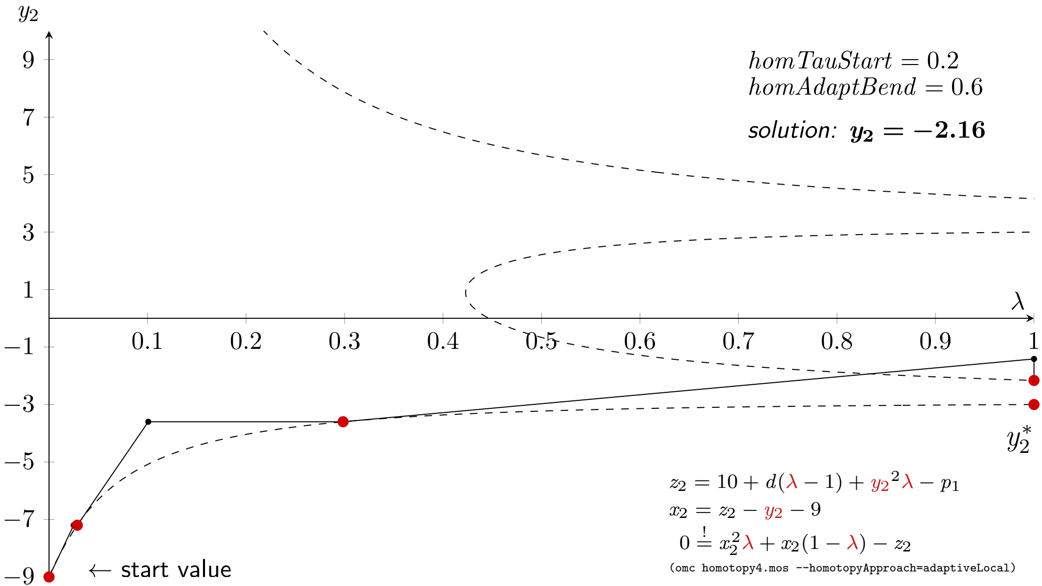
$homTauStart = 0.2$
$homAdaptBend = 0.6$

solution:

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \overset{!}{=} x_2^2\lambda + x_2(1 - \lambda) - z_2$$

(omc homotopy4.mos --homotopyApproach=adaptiveLocal)

Small changes to the homotopy parameters can lead to leaving the correct path:



$homTauStart = 0.2$
$homAdaptBend = 0.6$

solution: $\mathbf{y_2 = -2.16}$

$$z_2 = 10 + d(\lambda - 1) + {y_2}^2\lambda - p_1$$
$$x_2 = z_2 - y_2 - 9$$
$$0 \overset{!}{=} x_2^2\lambda + x_2(1-\lambda) - z_2$$

(`omc homotopy4.mos --homotopyApproach=adaptiveLocal`)

Characteristic Features of the Homotopy Algorithm

Impact of Tearing

Tearing can influence the homotopy path calculation:

Tearing Set 1 (omc homotopy4.mos --homotopyApproach=adaptiveGlobal)

Iteration variables: z_1, y_2, z_3

$$
\begin{aligned}
c &= b \cdot (1-\lambda) + b^2 \cdot \lambda \\
x_1 &= z_1 + c - 5 \\
y_1 &= 9 + x_1 - z_1 \\
d &= x_1 + y_1 + z_1 \\
z_2 &= 10 + d \cdot (\lambda - 1) + y_2^2 \cdot \lambda - p_1 \\
x_2 &= z_2 - (9 + y_2) \\
e &= \sin(d \cdot x_2) \\
x_3 &= z_3 + e - 5 \\
y_3 &= 9 + x_3 - z_3 \\
0 &\overset{!}{=} -10 + p_1 + z_1 + y_1 \\
0 &\overset{!}{=} x_2^2 \cdot \lambda + x_2 \cdot (1-\lambda) - z_2 \\
0 &\overset{!}{=} -10 + p_1 + y_3 \cdot (1 + z_3)
\end{aligned}
$$

Tearing Set 2 (omc homotopy4.mos --homotopyApproach=adaptiveGlobal)

Iteration variables: z_1, y_2, x_3

$$
\begin{aligned}
c &= b \cdot (1-\lambda) + b^2 \cdot \lambda \\
x_1 &= z_1 + c - 5 \\
y_1 &= 9 + x_1 - z_1 \\
d &= x_1 + y_1 + z_1 \\
z_2 &= 10 + d \cdot (\lambda - 1) + y_2^2 \cdot \lambda - p_1 \\
x_2 &= z_2 - (9 + y_2) \\
e &= sin(d \cdot x_2) \\
z_3 &= 5 + x_3 - e \\
y_3 &= 9 + x_3 - z_3 \\
0 &\overset{!}{=} -10 + p_1 + z_1 + y_1 \\
0 &\overset{!}{=} x_2^2 \cdot \lambda + x_2 \cdot (1-\lambda) - z_2 \\
0 &\overset{!}{=} -10 + p_1 + y_3 \cdot (1 + z_3)
\end{aligned}
$$

Characteristic Features of the Homotopy Algorithm

Impact of Tearing

FH Bielefeld
University of
Applied Sciences

Tearing can influence the homotopy path calculation:

Tearing Set 1 (omc homotopy4.mos --homotopyApproach=adaptiveGlobal)

Iteration variables: z_1, y_2, z_3

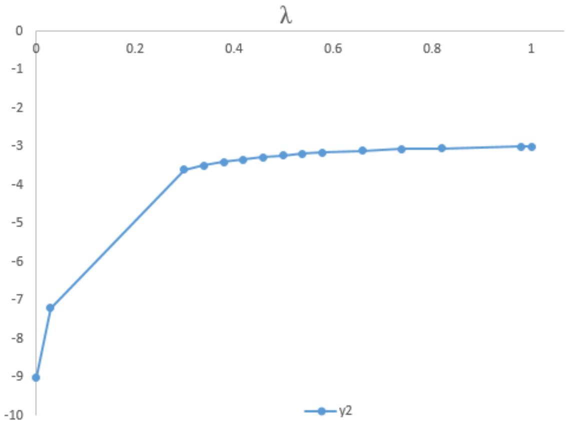
$$
\begin{aligned}
c &= b \cdot (1-\lambda) + b^2 \cdot \lambda \\
x_1 &= z_1 + c - 5 \\
y_1 &= 9 + x_1 - z_1 \\
d &= x_1 + y_1 + z_1 \\
z_2 &= 10 + d \cdot (\lambda - 1) + y_2^2 \cdot \lambda - p_1 \\
x_2 &= z_2 - (9 + y_2) \\
e &= \sin(d \cdot x_2) \\
x_3 &= z_3 + e - 5 \\
y_3 &= 9 + x_3 - z_3 \\
0 &\stackrel{!}{=} -10 + p_1 + z_1 + y_1 \\
0 &\stackrel{!}{=} x_2^2 \cdot \lambda + x_2 \cdot (1-\lambda) - z_2 \\
0 &\stackrel{!}{=} -10 + p_1 + y_3 \cdot (1 + z_3)
\end{aligned}
$$

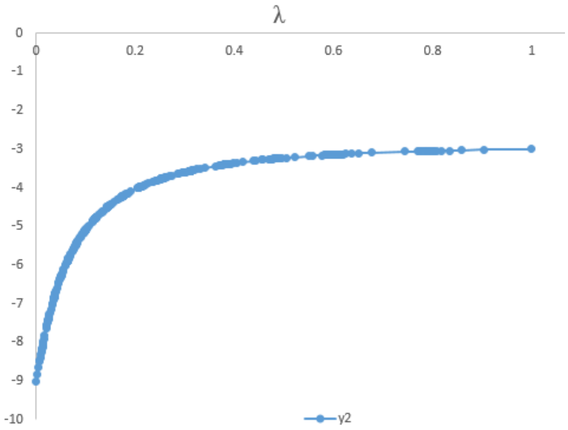
Tearing Set 2 (omc homotopy4.mos --homotopyApproach=adaptiveGlobal)

Iteration variables: z_1, y_2, x_3

$$
\begin{aligned}
c &= b \cdot (1-\lambda) + b^2 \cdot \lambda \\
x_1 &= z_1 + c - 5 \\
y_1 &= 9 + x_1 - z_1 \\
d &= x_1 + y_1 + z_1 \\
z_2 &= 10 + d \cdot (\lambda - 1) + y_2^2 \cdot \lambda - p_1 \\
x_2 &= z_2 - (9 + y_2) \\
e &= sin(d \cdot x_2) \\
z_3 &= 5 + x_3 - e \\
y_3 &= 9 + x_3 - z_3 \\
0 &\stackrel{!}{=} -10 + p_1 + z_1 + y_1 \\
0 &\stackrel{!}{=} x_2^2 \cdot \lambda + x_2 \cdot (1-\lambda) - z_2 \\
0 &\stackrel{!}{=} -10 + p_1 + y_3 \cdot (1 + z_3)
\end{aligned}
$$

Tearing can influence the homotopy path calculation:

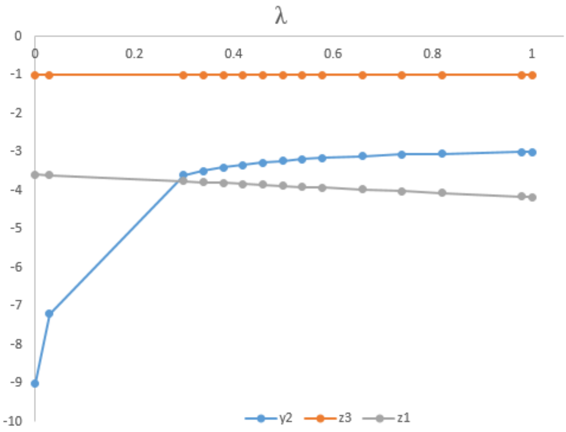




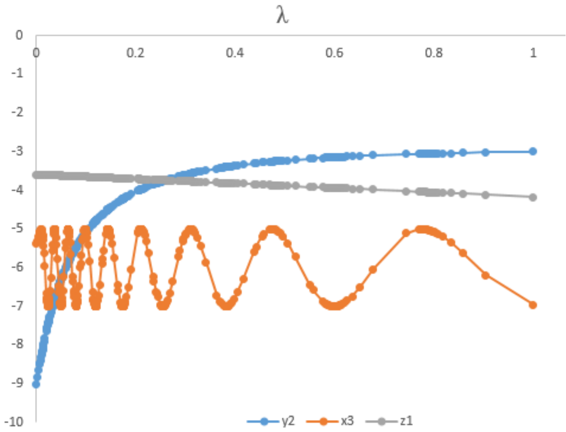
15 homotopy steps

426 homotopy steps

Tearing can influence the homotopy path calculation:



15 homotopy steps



426 homotopy steps

Debugging

Shell

- `-lv=LOG_INIT`
  - General information about initialization and the homotopy process
- `-lv=LOG_NLS_V`
  - General information about the solving process of nonlinear systems
- `-lv=LOG_NLS_JAC`
  - To print the Jacobians
- `-lv=LOG_NLS_HOMOTOPY`
  - Detailed information about the solving process of the homotopy solver

Homotopy Path

- `-lv=LOG_INIT` creates a csv-file with the homotopy path,
  e.g. *initializationTests.homotopy4_nonlinsys0_adaptive_global_homotopy_pos.csv*

→ Visualize with OMPlot, Excel, ...

Debugging

Shell

- -lv=LOG_INIT
  - General information about initialization and the homotopy process
- -lv=LOG_NLS_V
  - General information about the solving process of nonlinear systems
- -lv=LOG_NLS_JAC
  - To print the Jacobians
- -lv=LOG_NLS_HOMOTOPY
  - Detailed information about the solving process of the homotopy solver

Homotopy Path

- -lv=LOG_INIT creates a csv-file with the homotopy path, e.g. *initializationTests.homotopy4_nonlinsys0_adaptive_global_homotopy_pos.csv*

→ Visualize with OMPlot, Excel, ...

Conclusion

Summary

- OpenModelica provides several options for the initialization with homotopy
  - local/global, equidistant/adaptive
- Some expertise in homotopy-based initialization is needed to ...
  - guarantee a sensible usage of the homotopy operator.
  - create a unique homotopy path wherever possible.
  - choose the appropriate method.
  - set the homotopy parameters properly.
  - promote the convergence of the method (Tearing, ...).

Outlook

- Adding a homotopy library with well-conceived homotopy examples to the OpenModelica coverage test
  - → Find good default values for the homotopy parameters
- Activate one of the adaptive methods as the default fallback method for initialization

Conclusion

Summary

- OpenModelica provides several options for the initialization with homotopy
  - local/global, equidistant/adaptive
- Some expertise in homotopy-based initialization is needed to ...
  - guarantee a sensible usage of the homotopy operator.
  - create a unique homotopy path wherever possible.
  - choose the appropriate method.
  - set the homotopy parameters properly.
  - promote the convergence of the method (Tearing, ...).

Outlook

- Adding a homotopy library with well-conceived homotopy examples to the OpenModelica coverage test

  $\rightarrow$ Find good default values for the homotopy parameters
- Activate one of the adaptive methods as the default fallback method for initialization