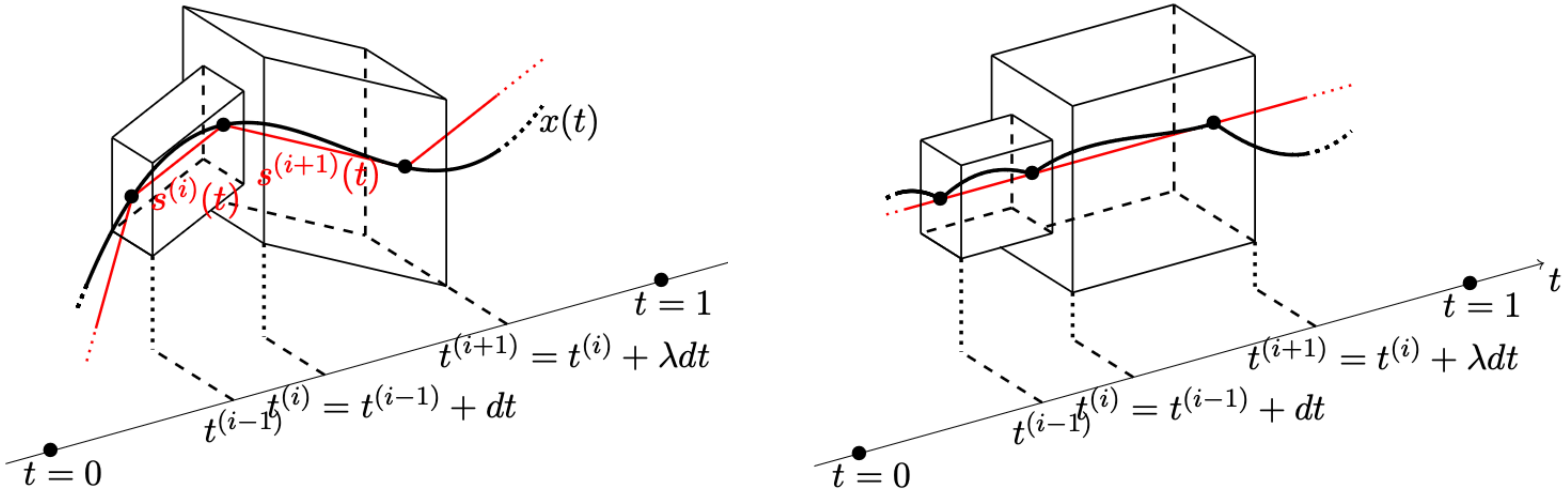
Certified homotopy tracking using the Krawczyk method

Joint work with Tim Duff



Kisun Lee (Clemson University) - kisunl@clemson.edu

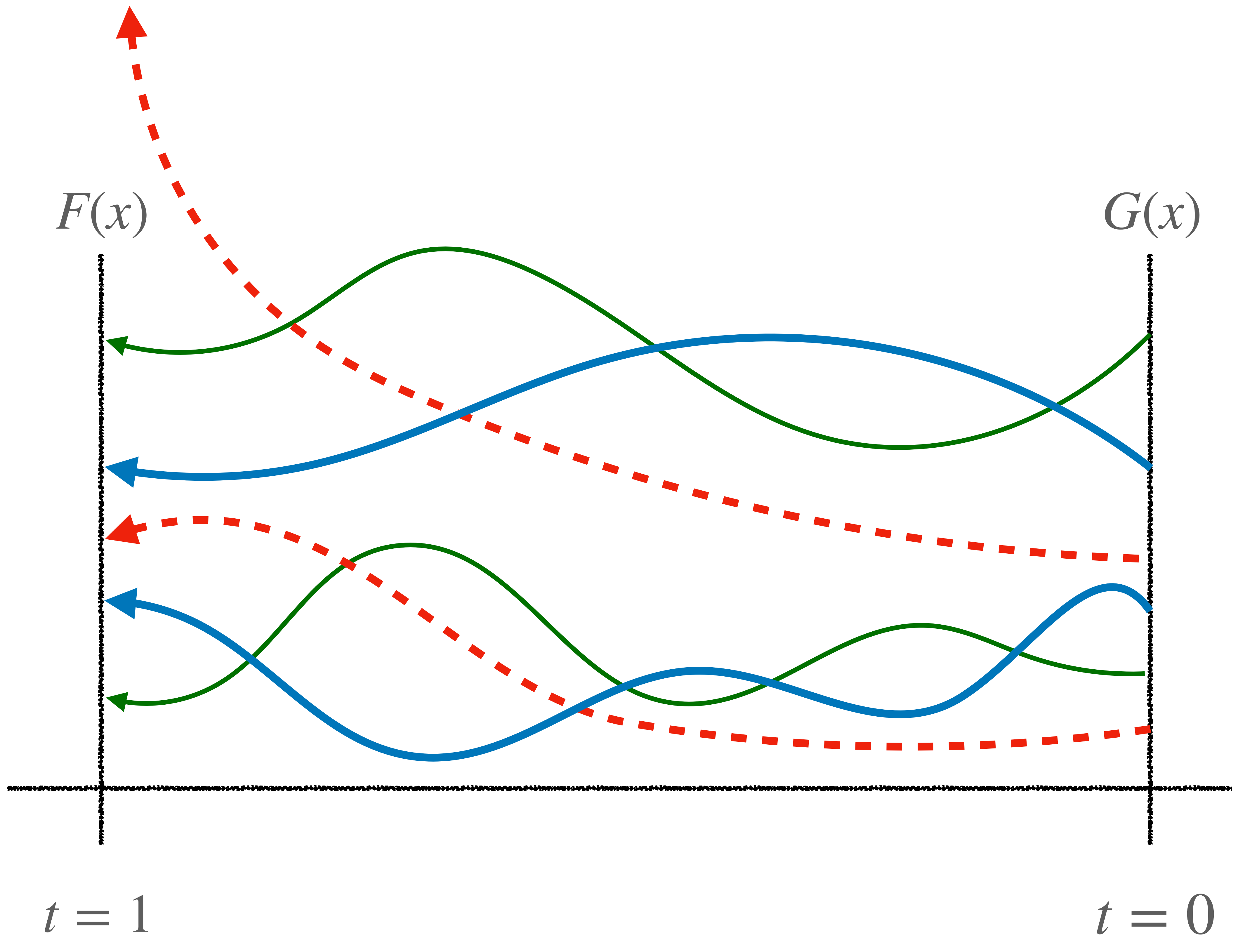
The International Symposium on Symbolic and Algebraic Computation (ISSAC) 2024

Homotopy continuation

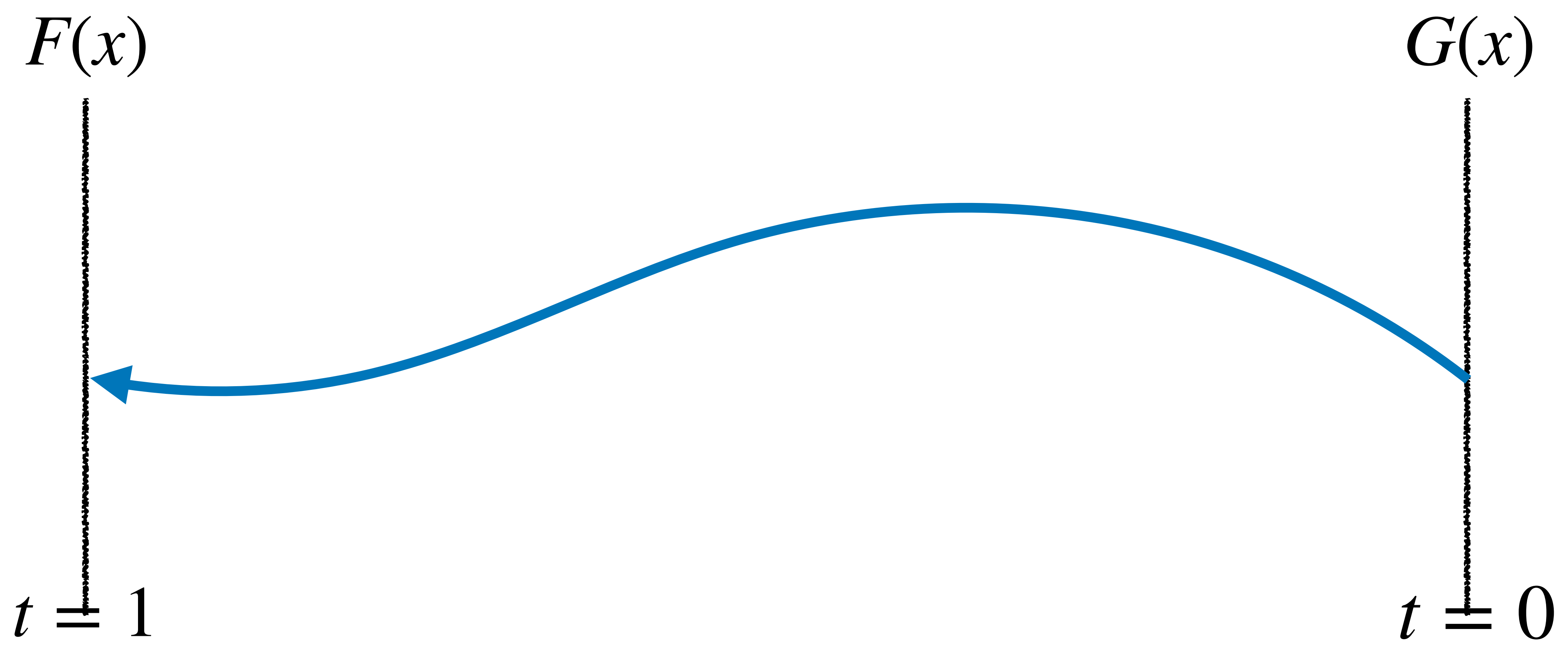
Finding solutions by tracking homotopy

$$H(t, x) = (1 - t)\gamma G(x) + tF(x), \quad t \in [0,1]$$

Solve F (target system) by constructing a homotopy with G (start system) whose solutions are known



Homotopy continuation



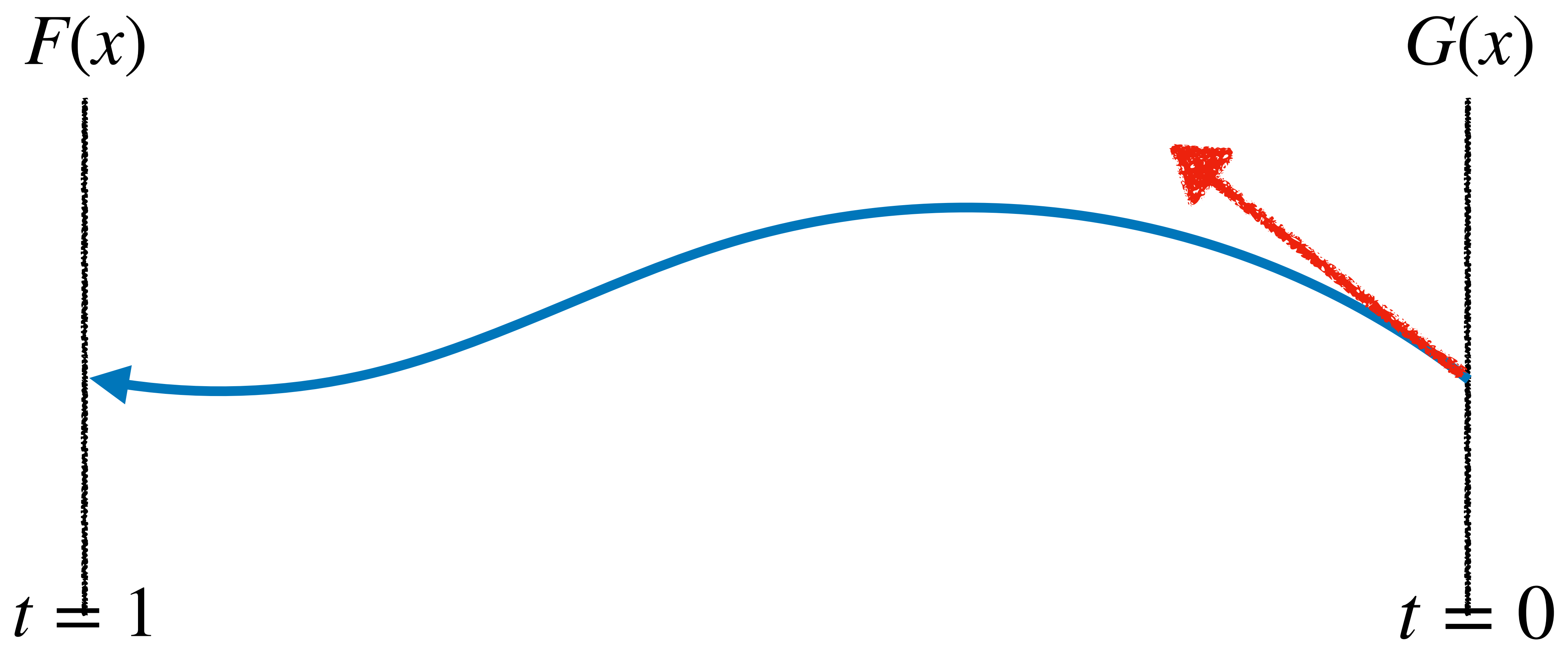
Homotopy continuation

- Predictor step

  Solving ODE

$$x'(t_0) = \left(\partial_x F_t\right)^{-1} \partial_t F_t \Big|_{x=x_0, t=t_0}$$

  to get the prediction.



Homotopy continuation

- Predictor step

  Solving ODE

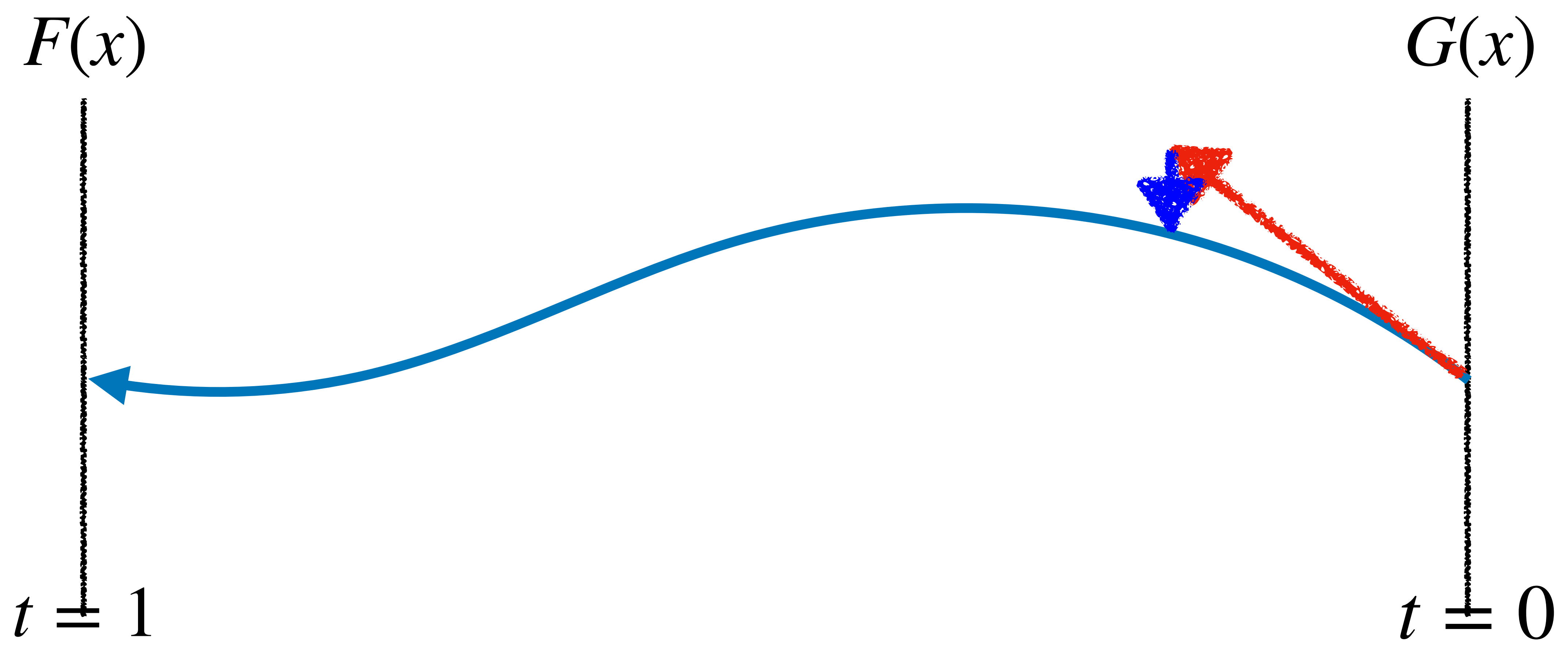
  $$x'(t_0) = \left(\partial_x F_t\right)^{-1} \partial_t F_t \Big|_{x=x_0, t=t_0}$$

  to get the prediction.

- Corrector step

  Refine the approximation using

  Newton's method



Homotopy continuation

- Predictor step

  Solving ODE

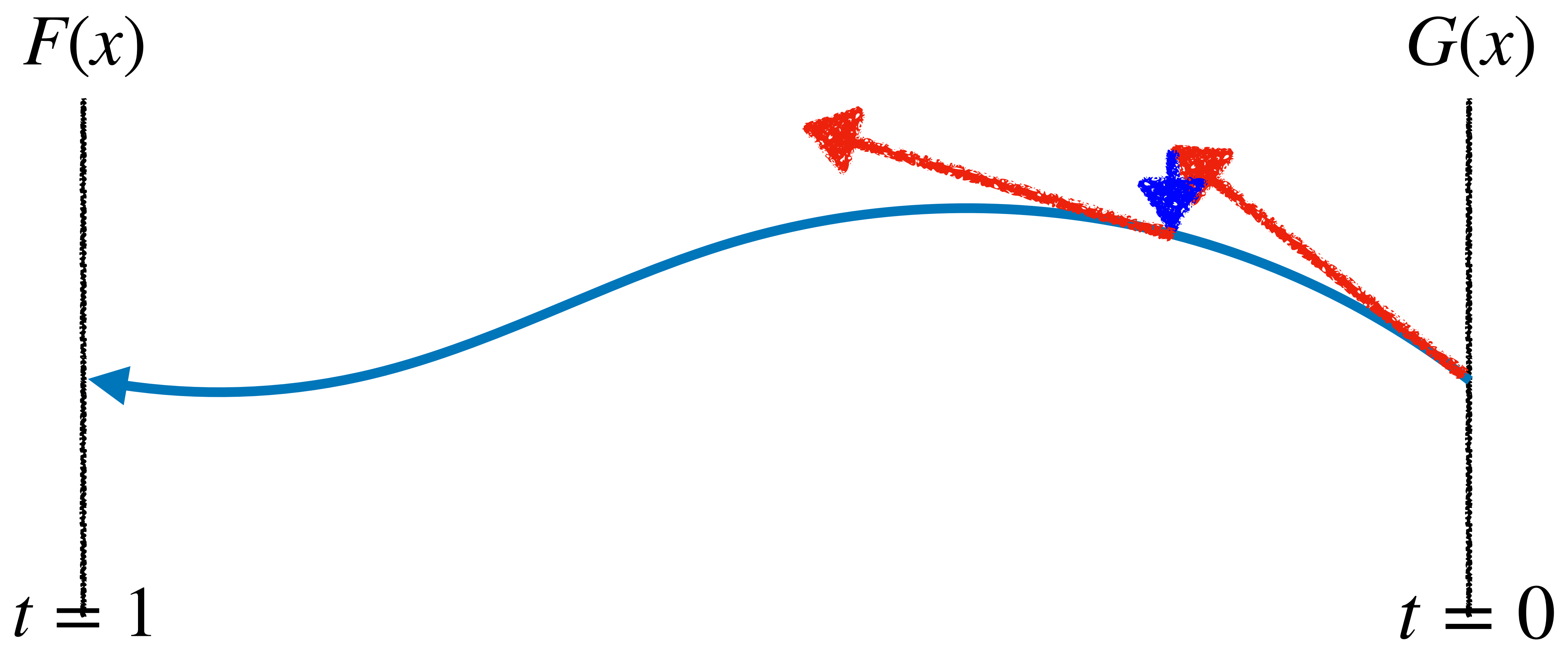
  $$x'(t_0) = \left(\partial_x F_t\right)^{-1} \partial_t F_t \Big|_{x=x_0, t=t_0}$$

  to get the prediction.

- Corrector step

  Refine the approximation using

  Newton's method



Homotopy continuation

- **Predictor step**

  Solving ODE

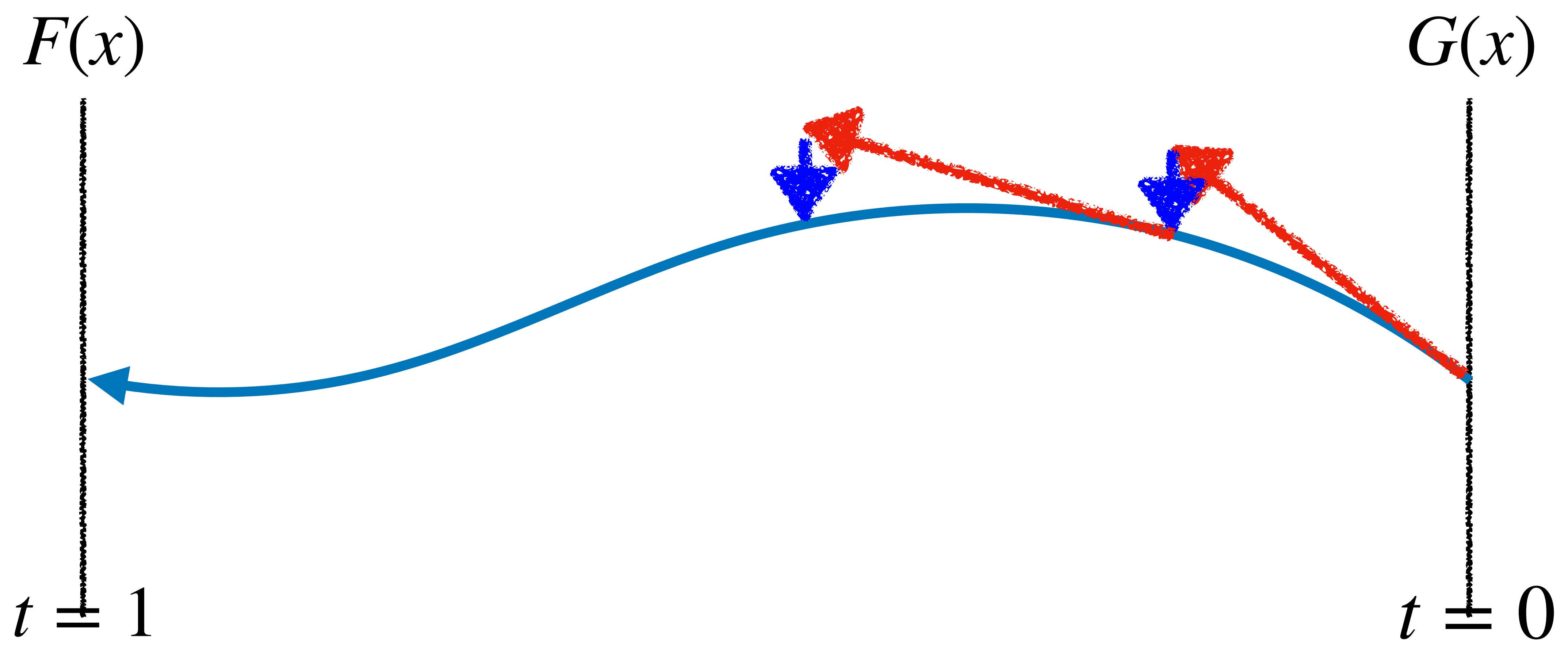
  $$x'(t_0) = \left(\partial_x F_t\right)^{-1} \partial_t F_t \Big|_{x=x_0, t=t_0}$$

  to get the prediction.

- **Corrector step**

  Refine the approximation using

  Newton's method



Homotopy continuation

- Predictor step

  Solving ODE

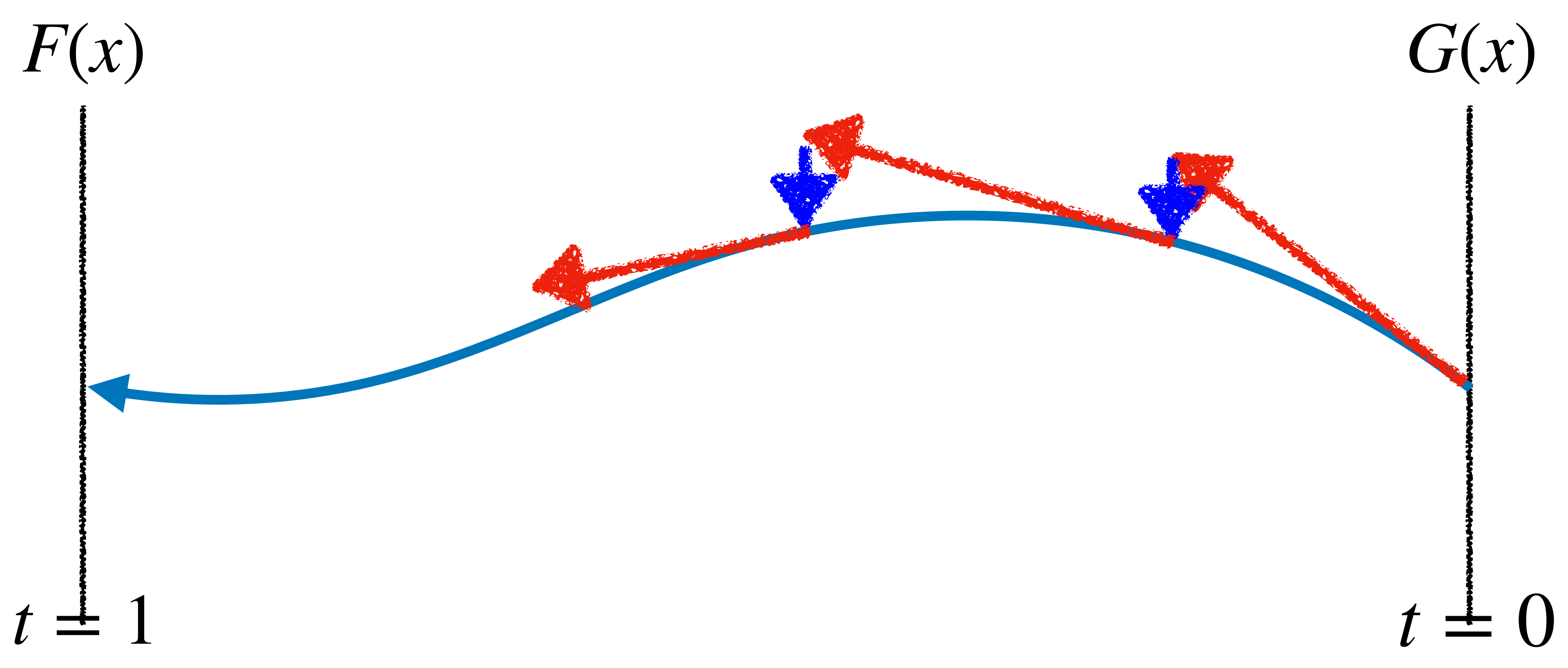
  $$x'(t_0) = \left(\partial_x F_t\right)^{-1} \partial_t F_t \Big|_{x=x_0, t=t_0}$$

  to get the prediction.

- Corrector step

  Refine the approximation using

  Newton's method



Homotopy continuation

- Predictor step

  Solving ODE

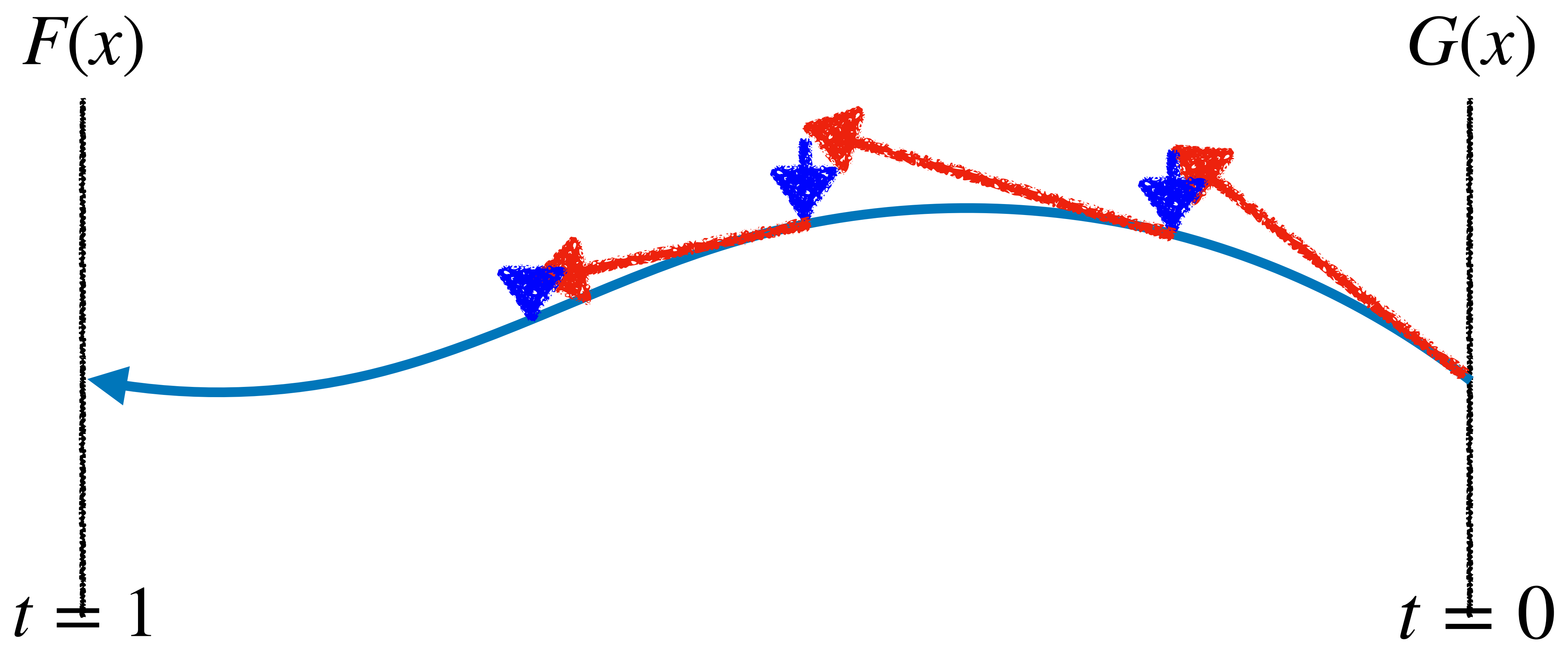
  $$x'(t_0) = \left(\partial_x F_t\right)^{-1} \partial_t F_t \Big|_{x=x_0, t=t_0}$$

  to get the prediction.

- Corrector step

  Refine the approximation using

  Newton's method



Homotopy continuation

- Predictor step

  Solving ODE

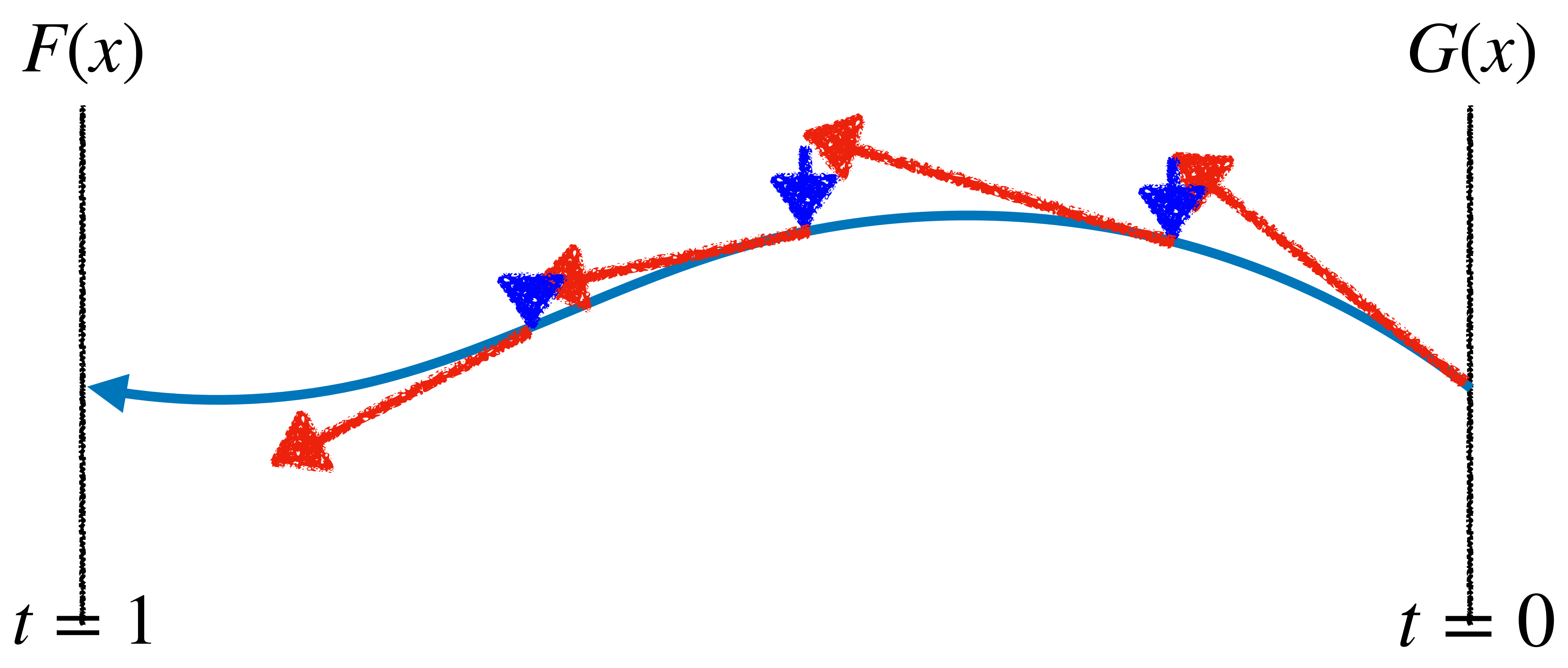
  $$x'(t_0) = \left(\partial_x F_t\right)^{-1} \partial_t F_t \Big|_{x=x_0, t=t_0}$$

  to get the prediction.

- Corrector step

  Refine the approximation using

  Newton's method



Homotopy continuation

- Predictor step

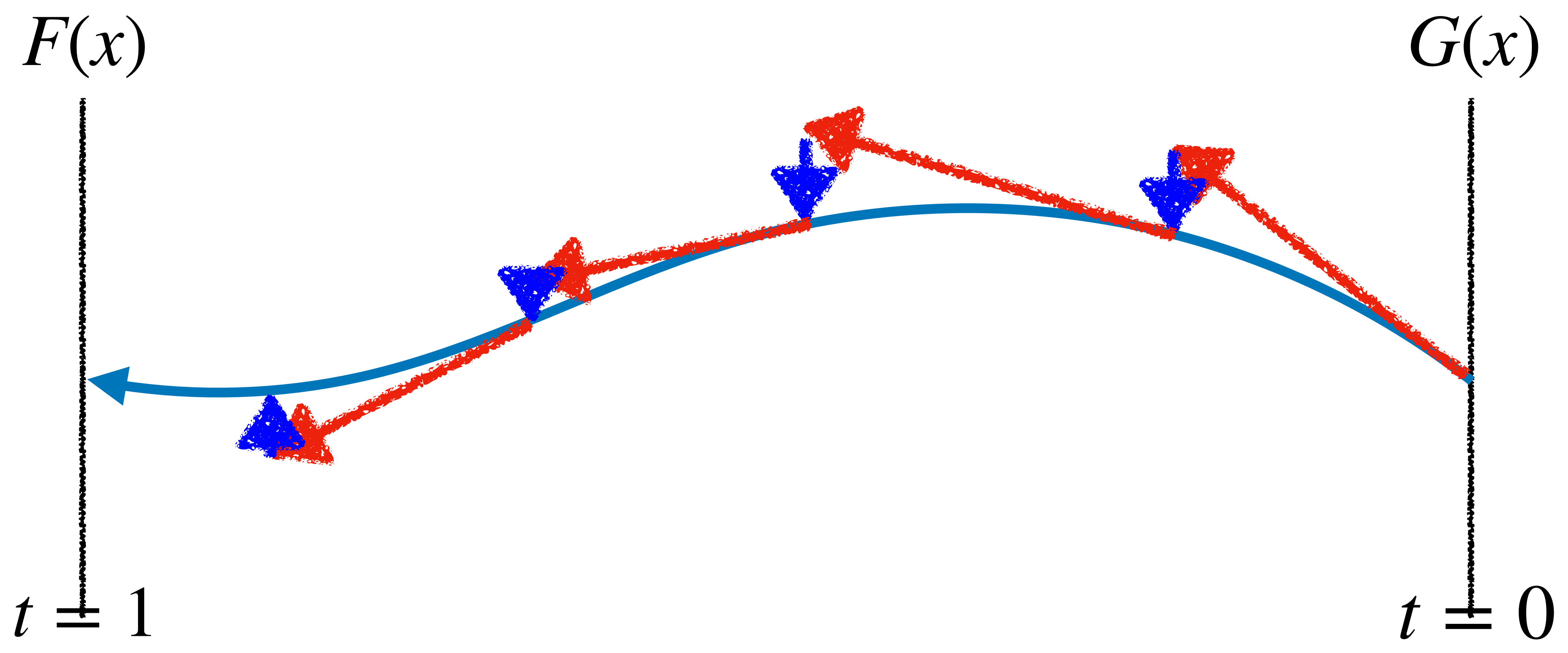
  Solving ODE

  $$x'(t_0) = \left(\partial_x F_t\right)^{-1} \partial_t F_t \Big|_{x=x_0, t=t_0}$$

  to get the prediction.

- Corrector step

  Refine the approximation using Newton's method



Homotopy continuation

- **Predictor step**

  Solving ODE

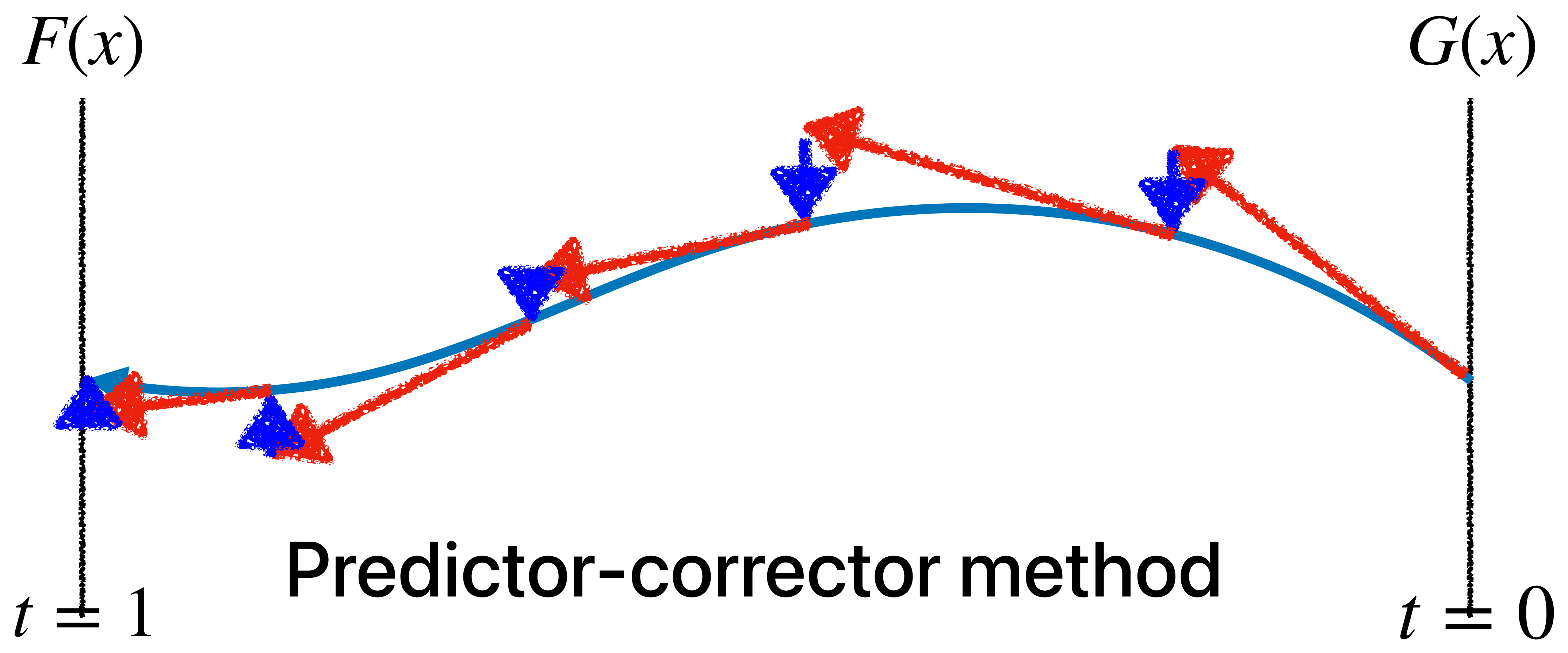
  $$x'(t_0) = \left(\partial_x F_t\right)^{-1} \partial_t F_t \Big|_{x=x_0, t=t_0}$$

  to get the prediction.

- **Corrector step**

  Refine the approximation using

  Newton's method



Predictor-corrector method

Homotopy continuation

- Homotopy continuation produces numerical solutions (approximations) to the system (Certification needed)

Homotopy continuation

- Homotopy continuation produces numerical solutions (approximations) to the system (Certification needed)
- For a system F and a numerical solution $x \in \mathbb{C}^n$ to F, we say that x is certified if x is (possibly) contained in a small region that also contains a unique exact solution $x^\star$ to F

Homotopy continuation

- Homotopy continuation produces numerical solutions (approximations) to the system (Certification needed)

- For a system F and a numerical solution $x \in \mathbb{C}^n$ to F, we say that x is certified if x is (possibly) contained in a small region that also contains a unique exact solution $x^\star$ to F (So that x can be refined to be arbitrarily close to $x^\star$ using a finite procedure)

Homotopy continuation

- Homotopy continuation produces numerical solutions (approximations) to the system (Certification needed)

- For a system F and a numerical solution $x \in \mathbb{C}^n$ to F, we say that x is certified if x is (possibly) contained in a small region that also contains a unique exact solution $x^\star$ to F (So that x can be refined to be arbitrarily close to $x^\star$ using a finite procedure)

(*a posteriori* certification)

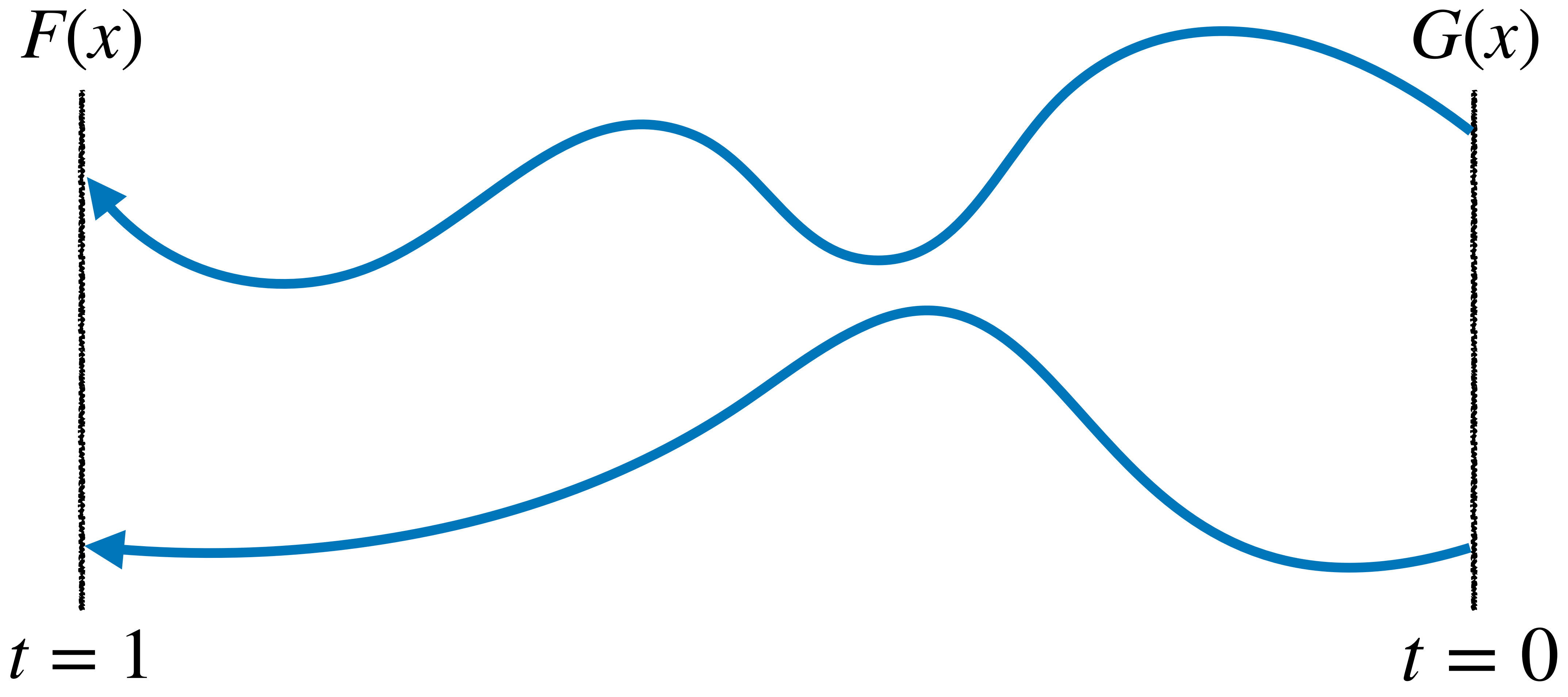
Certified homotopy tracking

- Post-processing certification does not verify the correctness of path tracking

Certified homotopy tracking

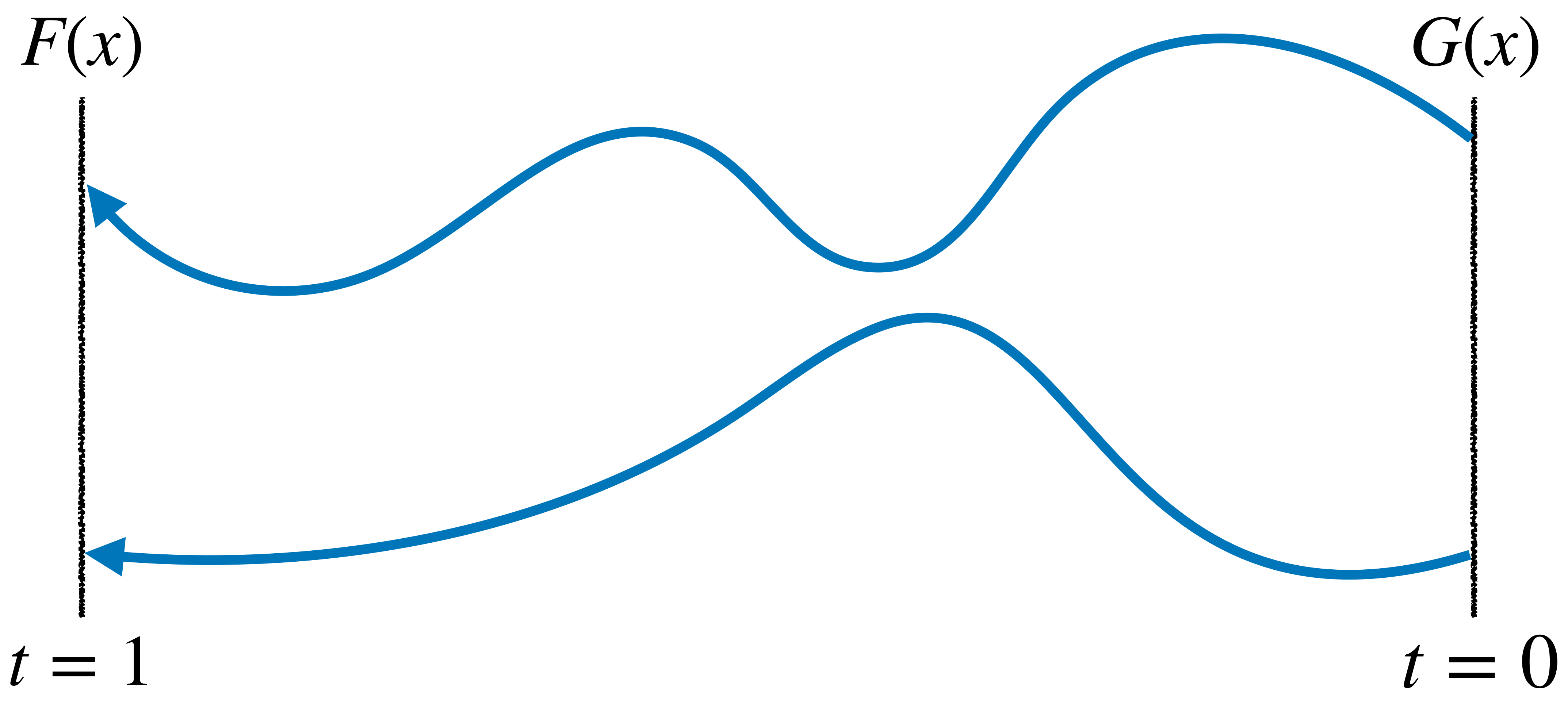
- Post-processing certification does not verify the correctness of path tracking
- In some algorithms or computations (e.g. monodromy or Galois group computation), the order and correspondence of solutions are important

Why certified homotopy tracking?



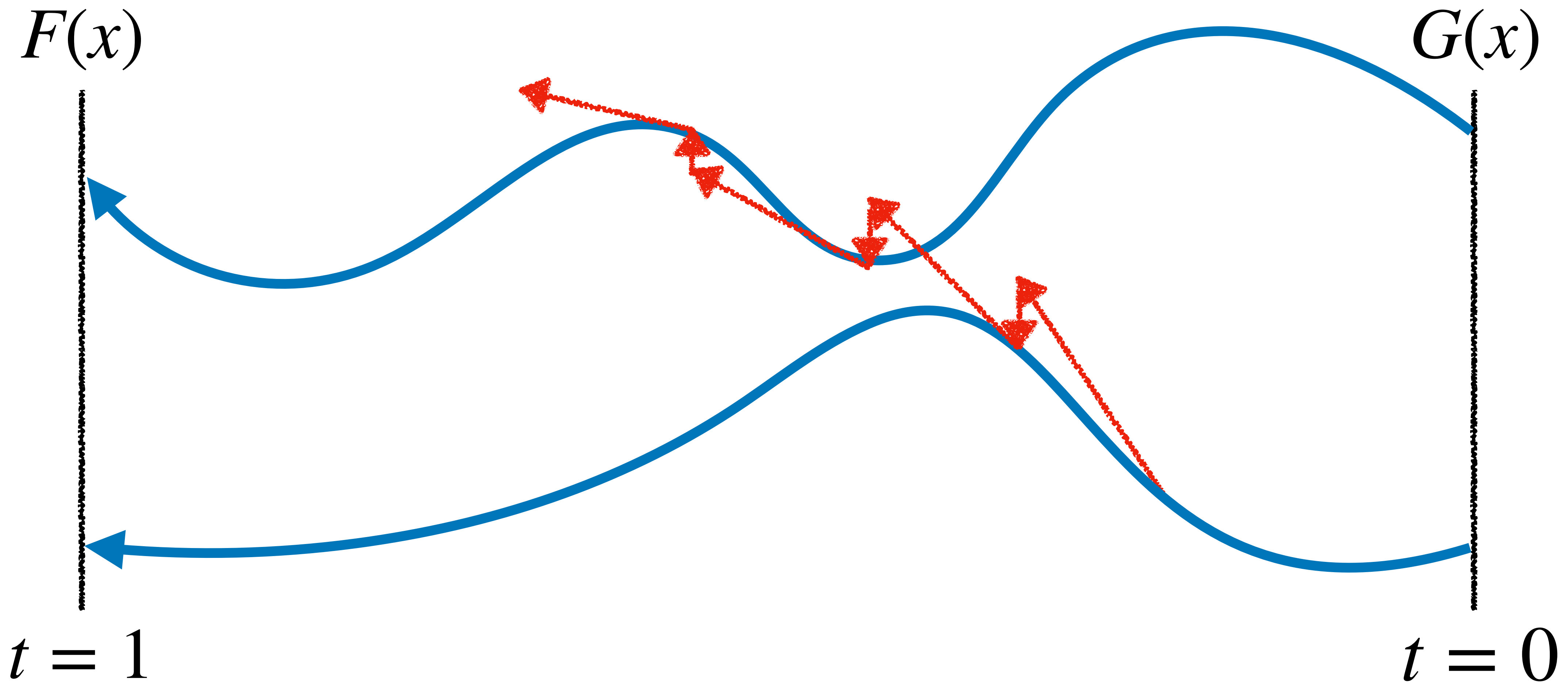
Why certified homotopy tracking?

- Scenario 1) Path jumping



Why certified homotopy tracking?

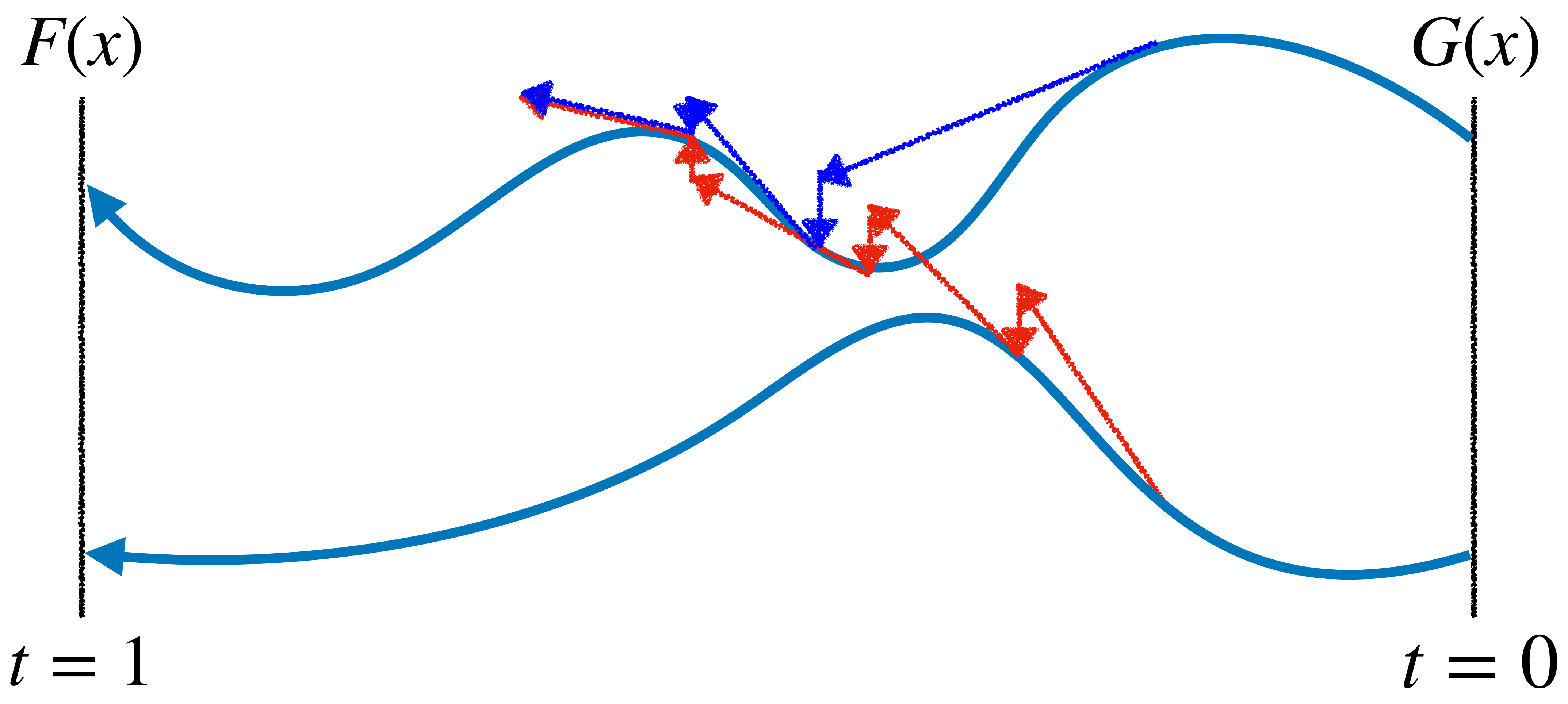
- Scenario 1) Path jumping



Why certified homotopy tracking?

- Scenario 1) Path jumping

Path jumping may result in false multiple solutions and a lost solution.

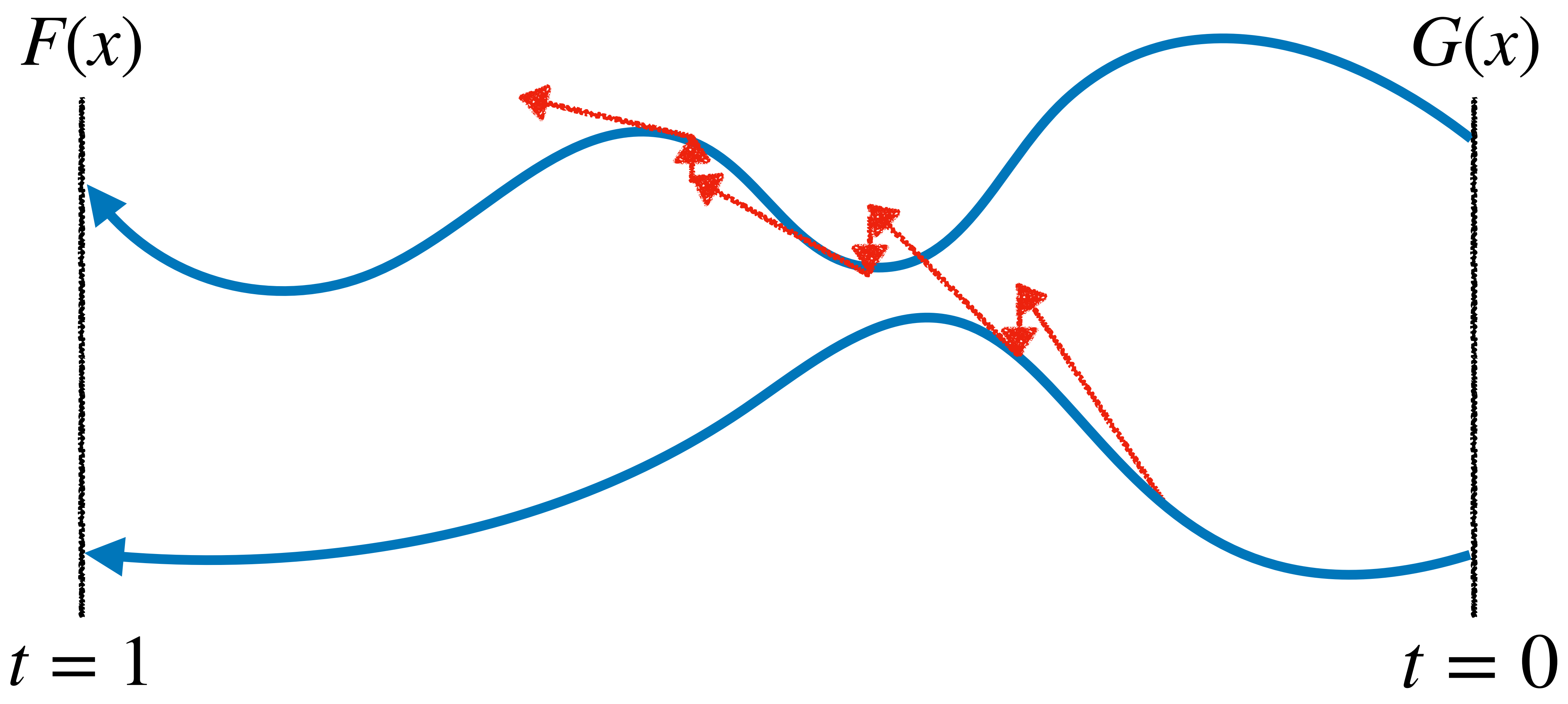


Why certified homotopy tracking?

- Scenario 1) Path jumping

  Path jumping may result in false multiple solutions and a lost solution.

- Scenario 2) Path crossing

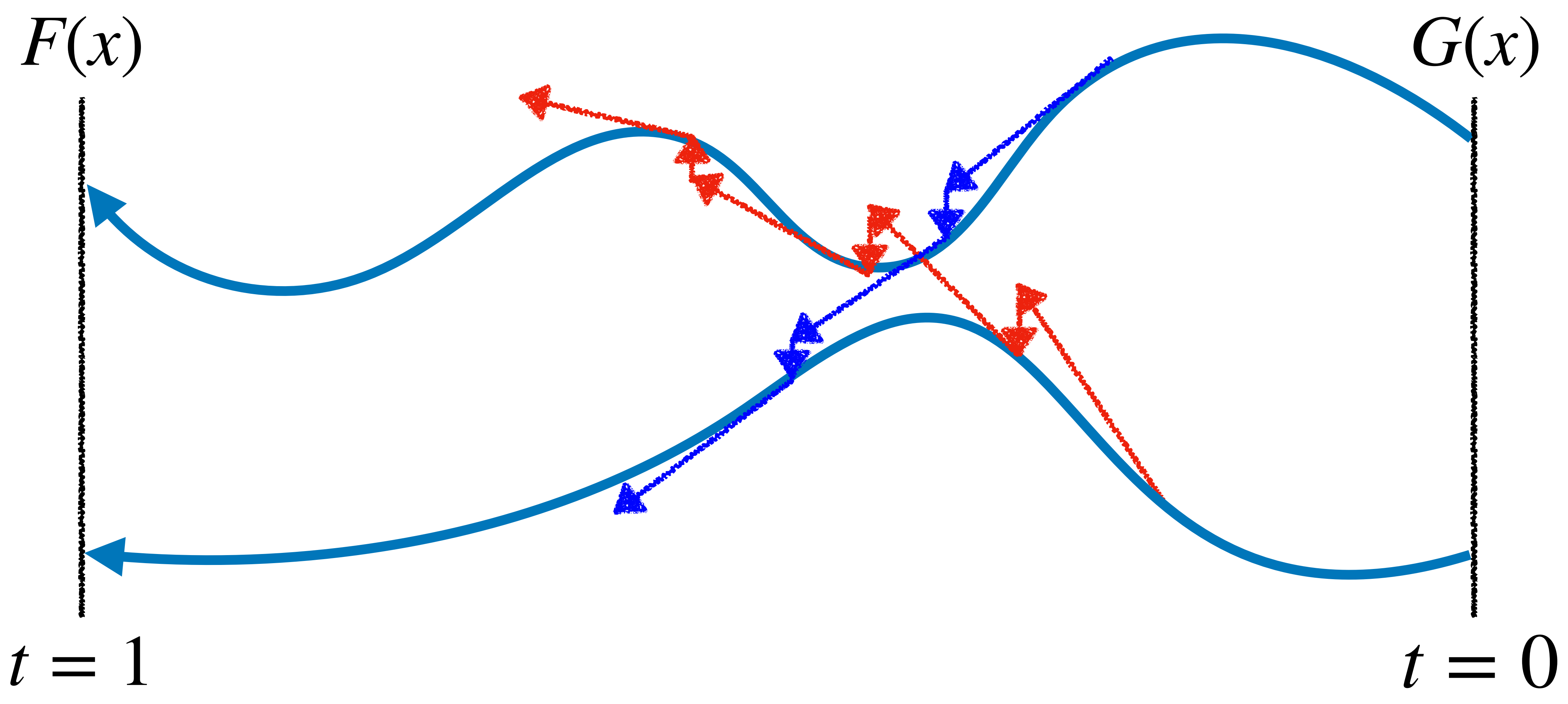


Why certified homotopy tracking?

- Scenario 1) Path jumping

  Path jumping may result in false multiple solutions and a lost solution.

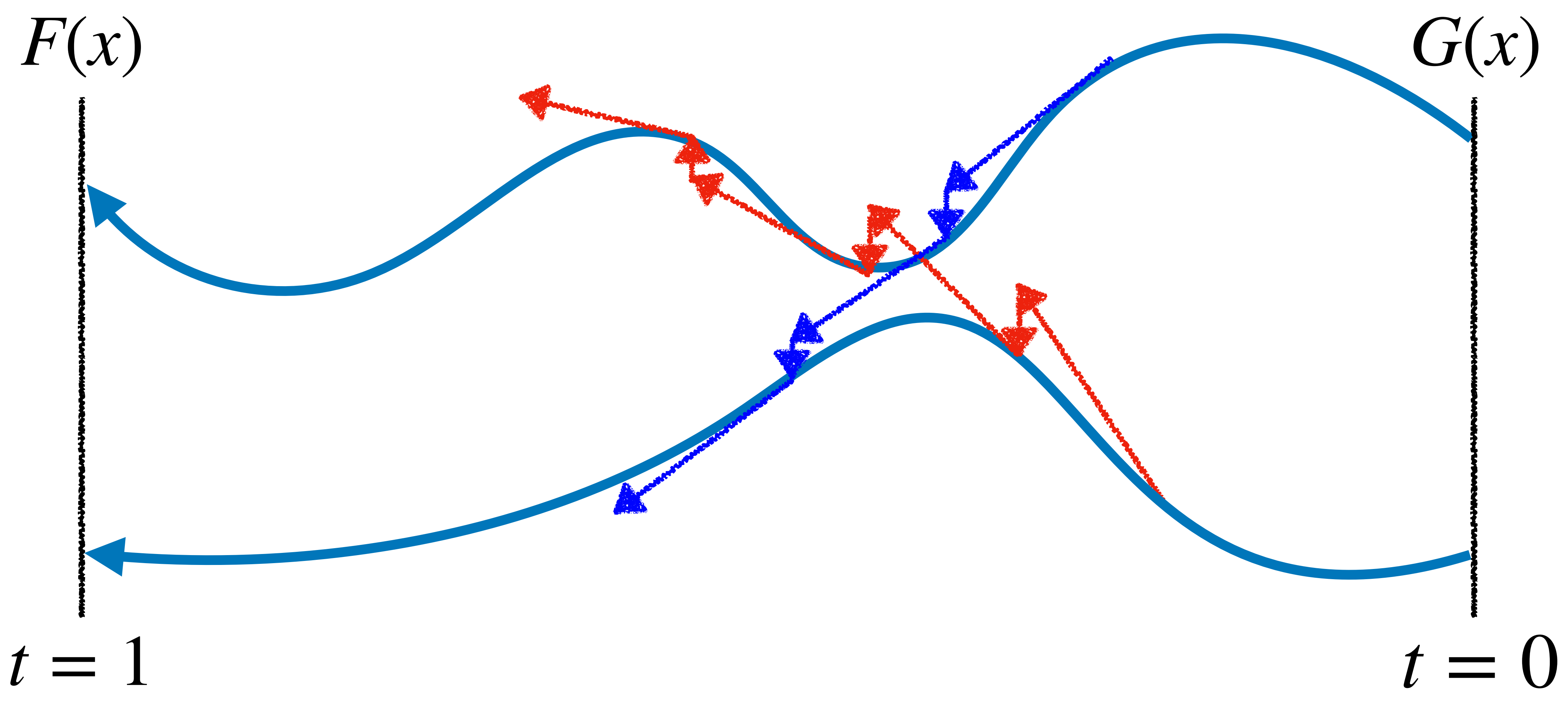
- Scenario 2) Path crossing



Why certified homotopy tracking?

- Scenario 1) Path jumping

  Path jumping may result in false multiple solutions and a lost solution.



- Scenario 2) Path crossing

  Path crossing loses correspondence between the start solution and the target solution.

Why certified homotopy tracking?

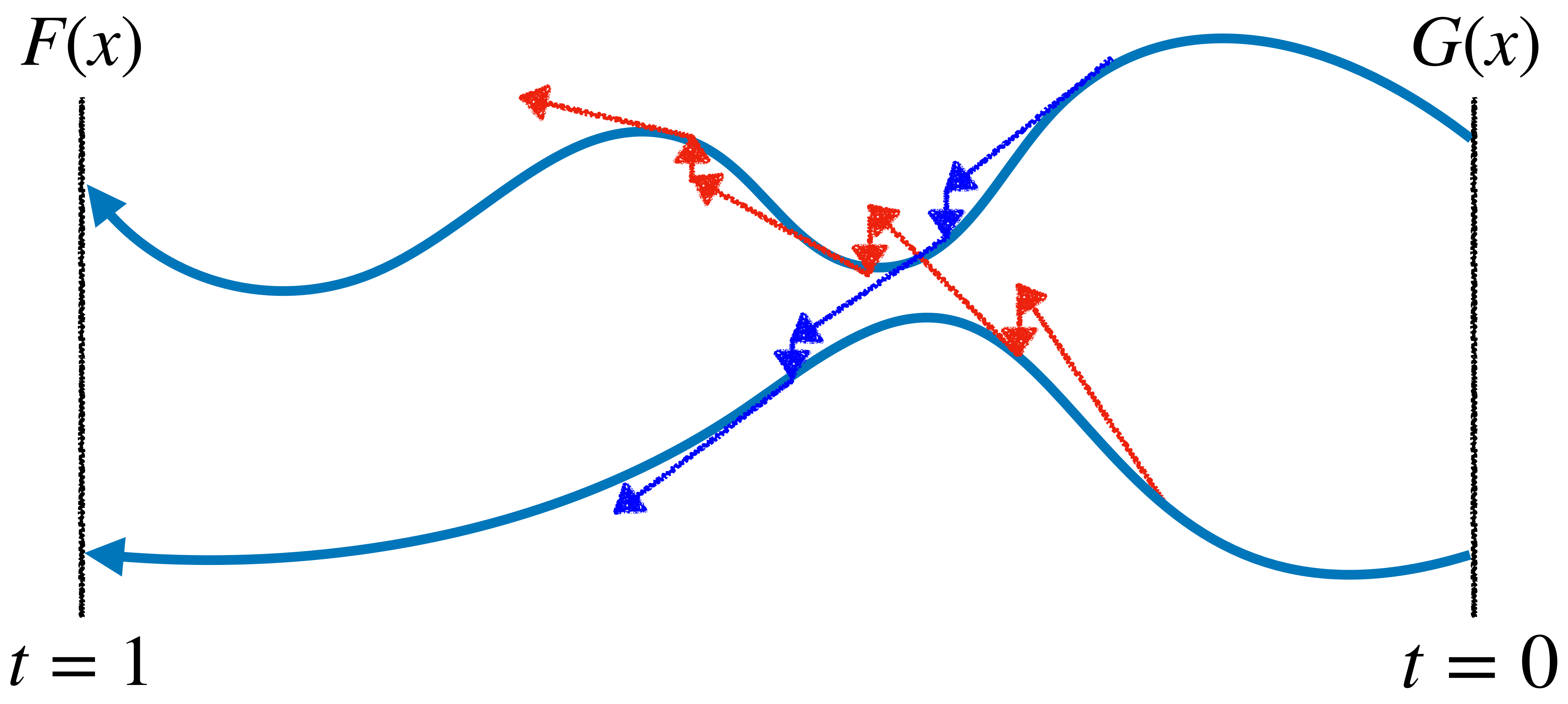
- Scenario 1) Path jumping

  Path jumping may result in false multiple solutions and a lost solution.

- Scenario 2) Path crossing

  Path crossing loses correspondence between the start solution and the target solution.

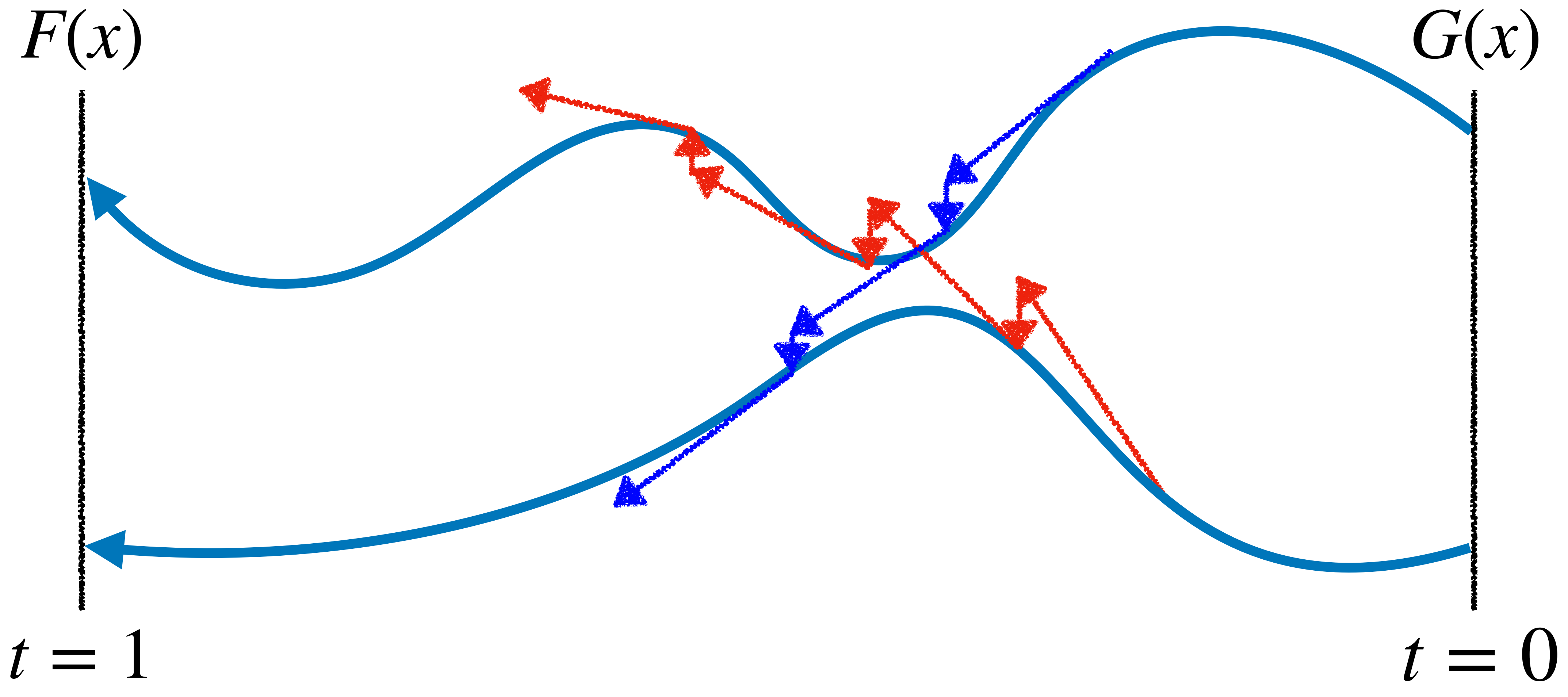
  Crucial for some algorithms (e.g. monodromy), and hard to detect.



Why certified homotopy tracking?

- Certified homotopy guarantees tracking the same path from start to finish without jumping or crossing

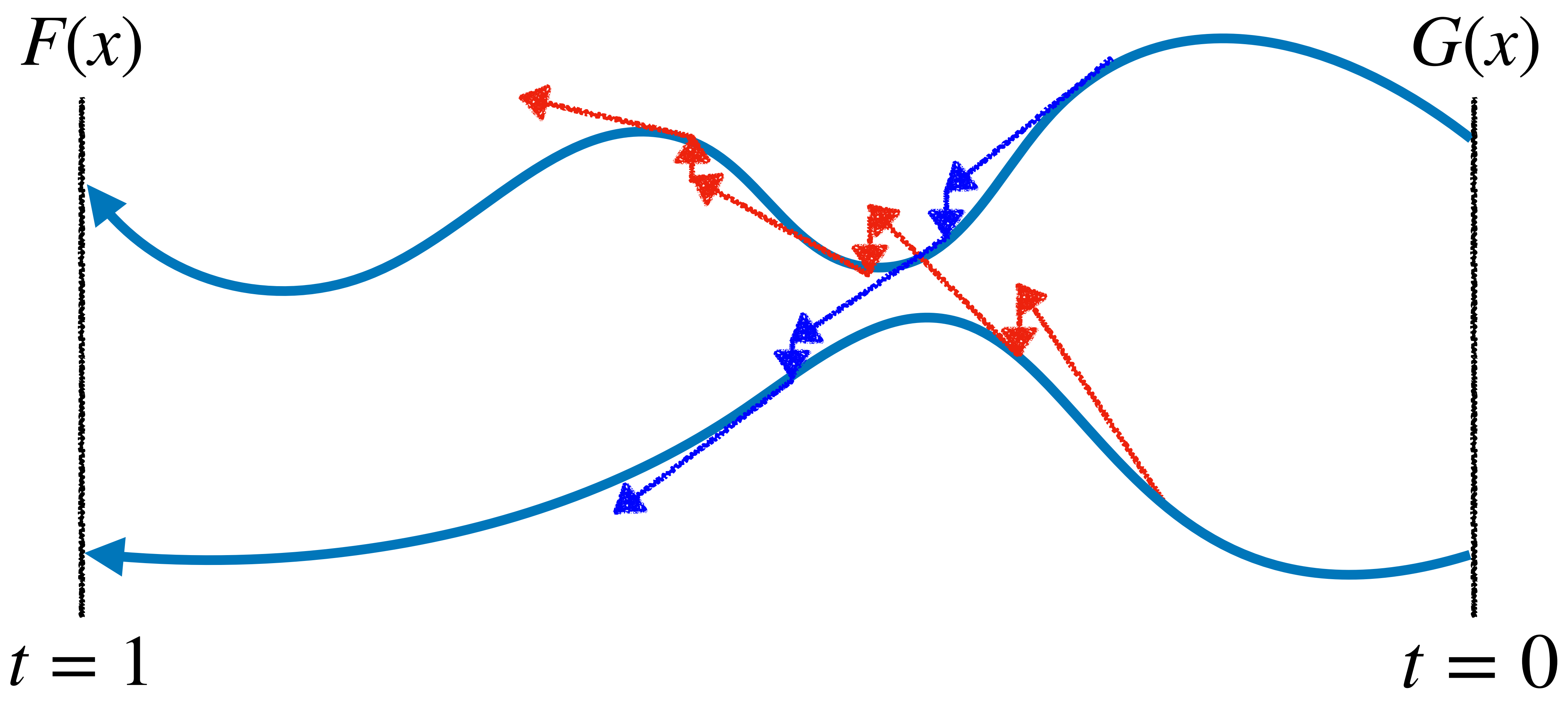
(Beltran-Leykin 2012 2013,
Hauenstein-Haywood-Liddell 2014,
van der Hoeven 2015,
Xu-Burr-Yap 2018,



Why certified homotopy tracking?

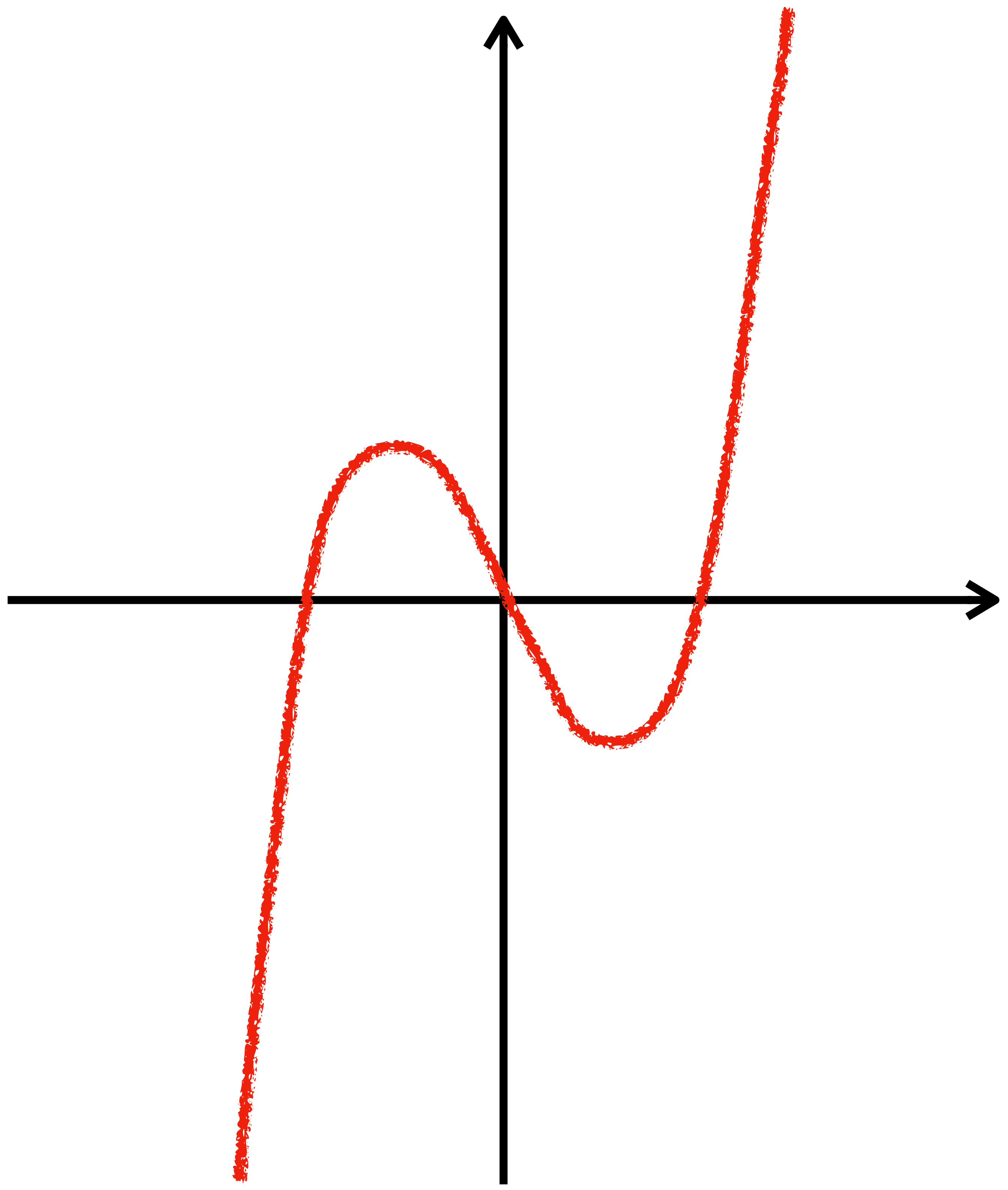
- Certified homotopy guarantees tracking the same path from start to finish without jumping or crossing

(Beltran-Leykin 2012 2013,
Hauenstein-Haywood-Liddell 2014,
van der Hoeven 2015,
Xu-Burr-Yap 2018,
Guillemot-Lairez 2024)



Interval arithmetic

- interval arithmetic



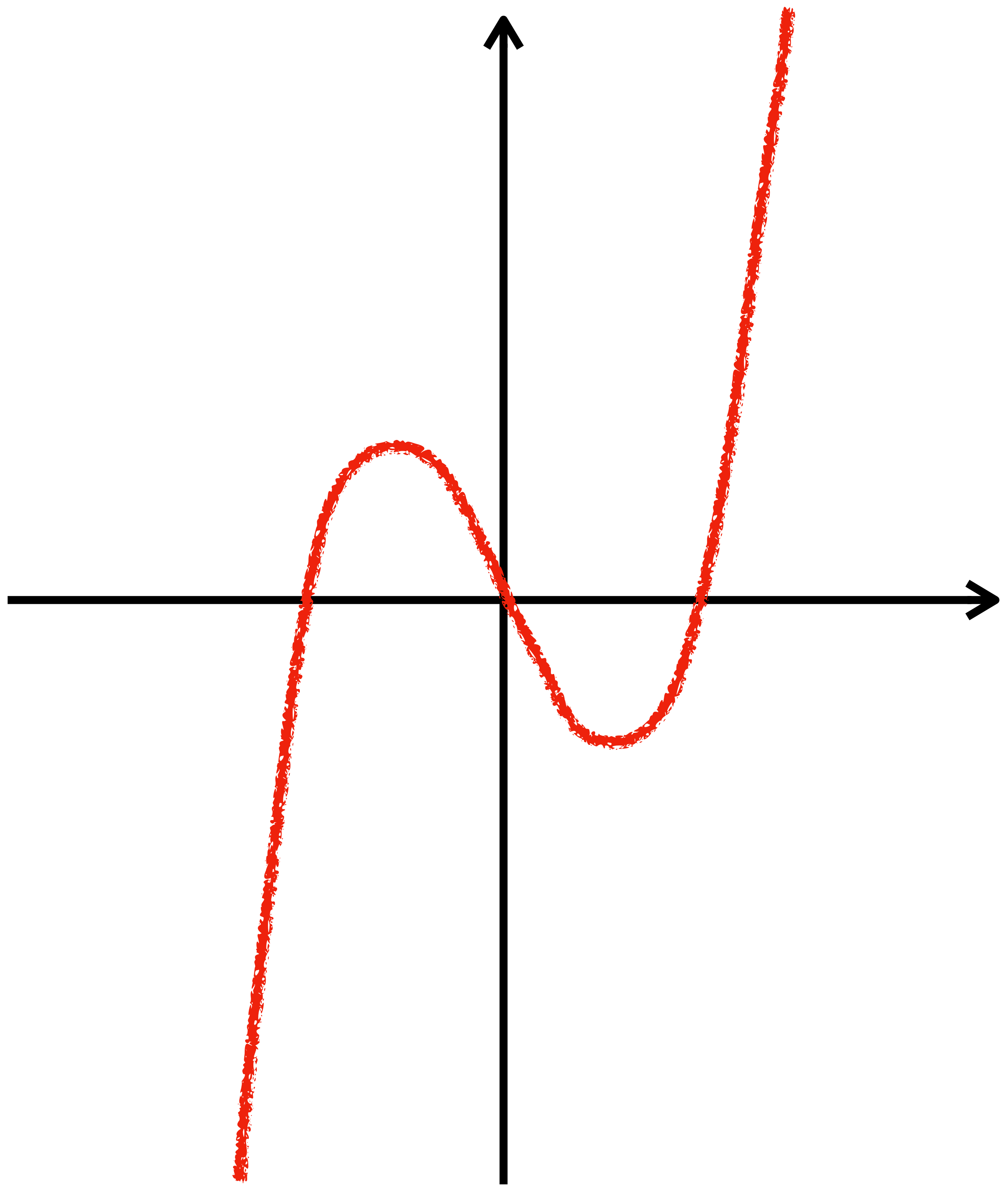
Interval arithmetic

- interval arithmetic

  for an arithmetic operator $\odot$, define

  $$[a, b] \odot [c, d] = \{x \odot y \mid x \in [a, b], y \in [c, d]\}$$

  for conservative and guaranteed computation



Interval arithmetic

- interval arithmetic

  for an arithmetic operator $\odot$, define

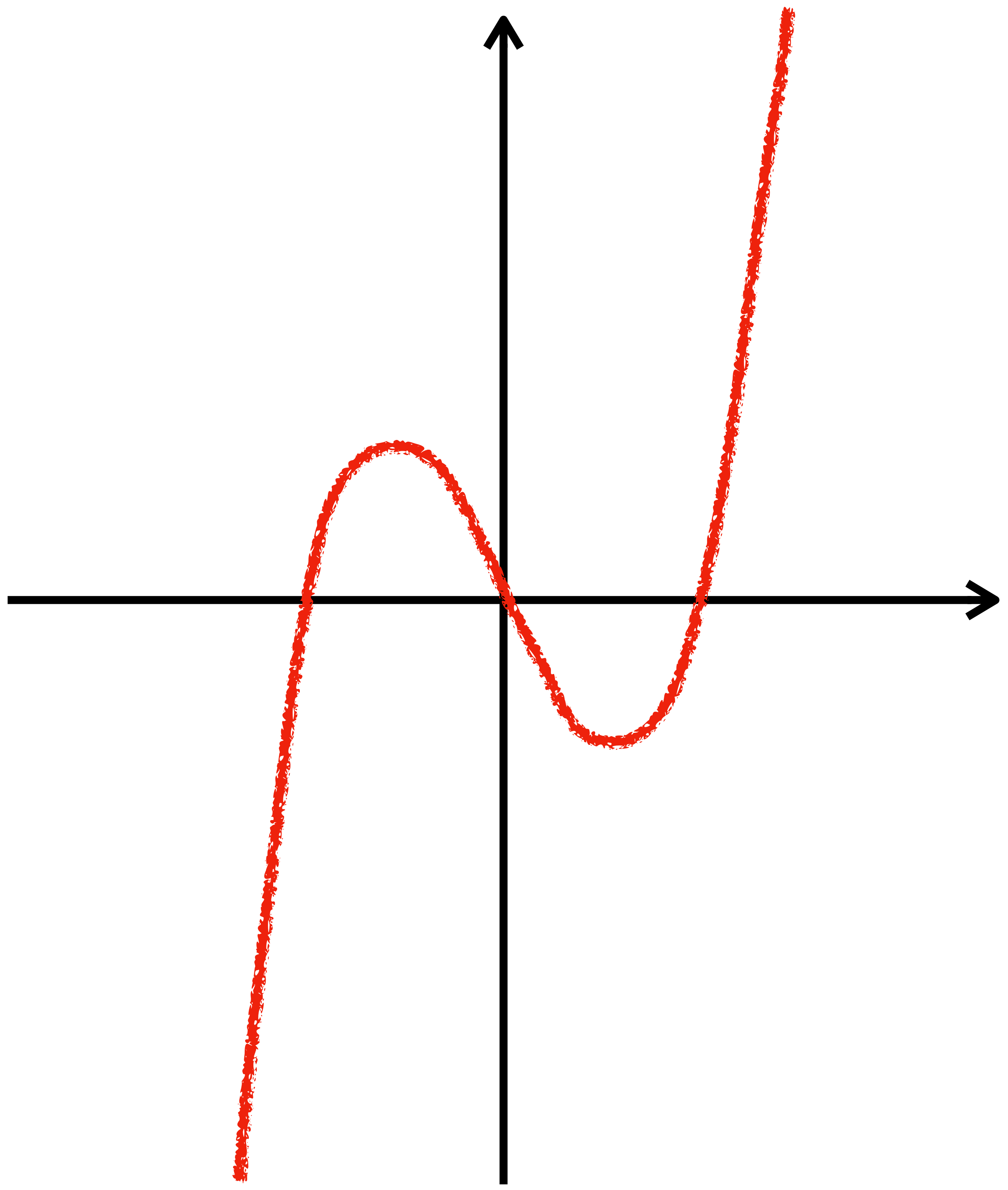
  $$[a, b] \odot [c, d] = \{x \odot y \mid x \in [a, b], y \in [c, d]\}$$

  for conservative and guaranteed computation

  ex) $f(x) = x^3 - 2x$ on $x \in [-2,2]$.

  Then, $f(x) \in [-2,2]^3 - 2 \cdot [-2,2] = [-12,12]$

  (much larger than the actual image $[-4,4]$)



Interval arithmetic

- interval arithmetic

  for an arithmetic operator $\odot$, define

  $[a,b] \odot [c,d] = \{x \odot y \mid x \in [a,b], y \in [c,d]\}$

  for conservative and guaranteed computation

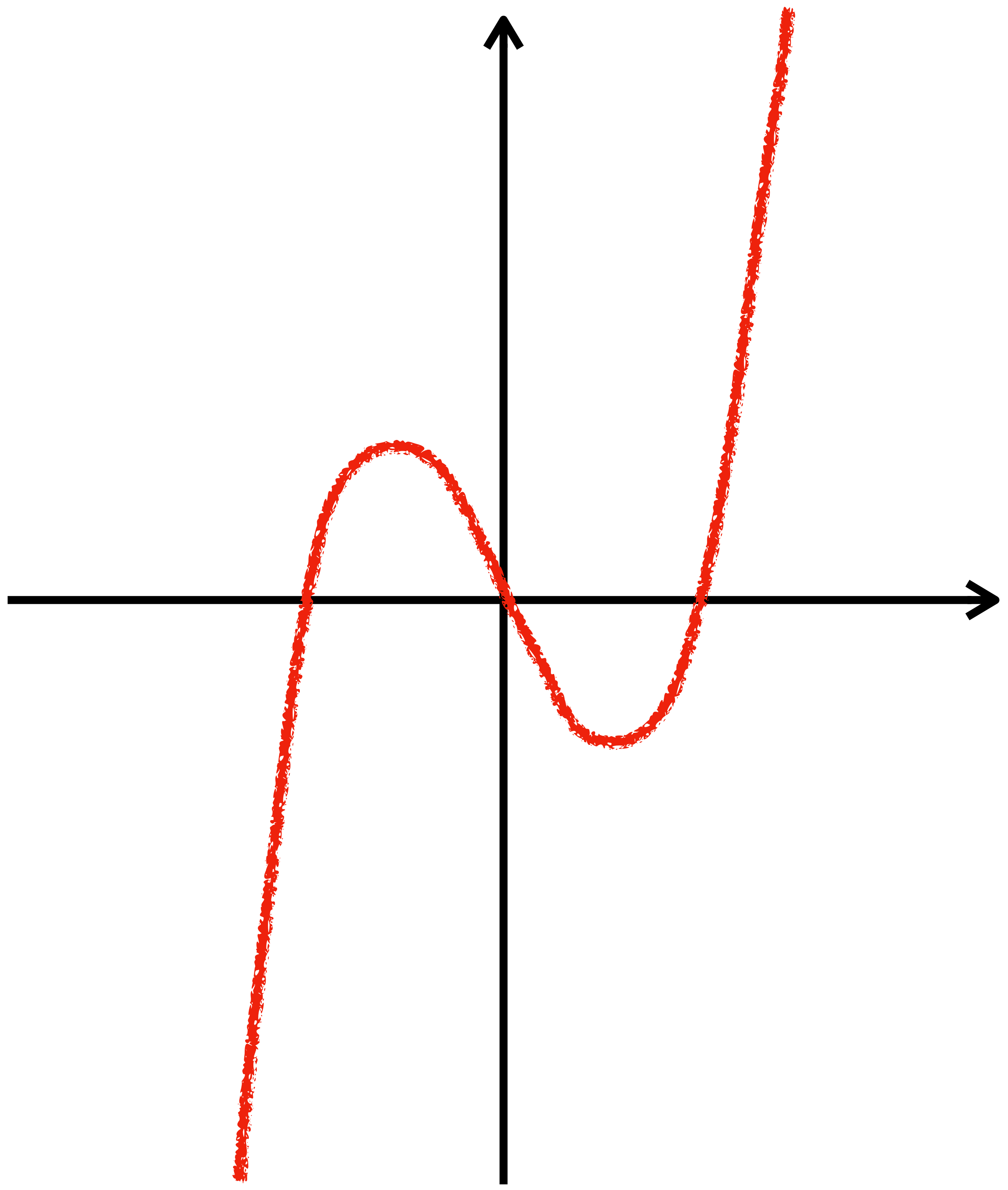
  ex) $f(x) = x^3 - 2x$ on $x \in [-2,2]$.

  Then, $f(x) \in [-2,2]^3 - 2 \cdot [-2,2] = [-12,12]$

  (much larger than the actual image $[-4,4]$)

  If a certain property is true for $[-12,12]$,

  then it is true for any point in the actual image of $f(x)$ over $[-2,2]$



Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y \square \frac{dF}{dx}(I) \right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\square F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y\,\square\frac{dF}{dx}(I)\right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\square F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

Theorem (Krawczyk 1969). For the Krawczyk operator the following hold :

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y \square \frac{dF}{dx}(I) \right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\square F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

Theorem (Krawczyk 1969). For the Krawczyk operator the following hold :

1. If $K_{x,Y}(I) \subset I$, then there is a solution of F in I (**existence** of a solution)

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y\Box\frac{dF}{dx}(I)\right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\Box F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

Theorem (Krawczyk 1969). For the Krawczyk operator the following hold :

1. If $K_{x,Y}(I) \subset I$, then there is a solution of F in I (**existence** of a solution)
2. If I has a solution and $\sqrt{2}\|Id - Y\Box\frac{dF}{dx}(I)\| < 1$, then there is a solution of F in I, and it is unique (**uniqueness** of a solution)

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y \square \frac{dF}{dx}(I) \right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\square F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

ex) $F = \begin{bmatrix} x^2 - y - 1 \\ -\frac{1}{2}x^2 + y^2 \end{bmatrix}$

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y\square\frac{dF}{dx}(I)\right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\square F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

ex) $F = \begin{bmatrix} x^2 - y - 1 \\ -\frac{1}{2}x^2 + y^2 \end{bmatrix}$ has a root $(\sqrt{2},1)$.

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y \square \frac{dF}{dx}(I) \right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\square F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

ex) $F = \begin{bmatrix} x^2 - y - 1 \\ -\frac{1}{2}x^2 + y^2 \end{bmatrix}$ has a root $(\sqrt{2}, 1)$.

(But let us pretend not to know!)

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y \square \frac{dF}{dx}(I) \right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\square F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

ex) $F = \begin{bmatrix} x^2 - y - 1 \\ -\frac{1}{2}x^2 + y^2 \end{bmatrix}$ has a root $(\sqrt{2}, 1)$.

(But let us pretend not to know!)

$$I = \begin{bmatrix} [1.2, 1.6] \\ [0.8, 1.2] \end{bmatrix}$$

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y \square \frac{dF}{dx}(I) \right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\square F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

ex) $F = \begin{bmatrix} x^2 - y - 1 \\ -\frac{1}{2}x^2 + y^2 \end{bmatrix}$ has a root $(\sqrt{2}, 1)$.

(But let us pretend not to know!)

$I = \begin{bmatrix} [1.2, 1.6] \\ [0.8, 1.2] \end{bmatrix}$ with $x = \begin{bmatrix} 1.4 \\ 1 \end{bmatrix}$ and $Y = JF(x)^{-1}$

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y\,\square\frac{dF}{dx}(I)\right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\square F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

ex) $F = \begin{bmatrix} x^2 - y - 1 \\ -\frac{1}{2}x^2 + y^2 \end{bmatrix}$ has a root $(\sqrt{2},1)$.

(But let us pretend not to know!)

$I = \begin{bmatrix} [1.2,1.6] \\ [0.8,1.2] \end{bmatrix}$ with $x = \begin{bmatrix} 1.4 \\ 1 \end{bmatrix}$ and $Y = JF(x)^{-1}$

Then, $K_{x,Y}(I) = \begin{bmatrix} [1.3476\cdots,1.4810\cdots] \\ [.89333\cdots,1.1067\cdots] \end{bmatrix}$

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y\square\frac{dF}{dx}(I)\right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\square F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

ex) $F = \begin{bmatrix} x^2 - y - 1 \\ -\frac{1}{2}x^2 + y^2 \end{bmatrix}$ has a root $(\sqrt{2}, 1)$.

(But let us pretend not to know!)

$I = \begin{bmatrix} [1.2, 1.6] \\ [0.8, 1.2] \end{bmatrix}$ with $x = \begin{bmatrix} 1.4 \\ 1 \end{bmatrix}$ and $Y = JF(x)^{-1}$

Then, $K_{x,Y}(I) = \begin{bmatrix} [1.3476\cdots, 1.4810\cdots] \\ [.89333\cdots, 1.1067\cdots] \end{bmatrix} \subset I.$

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y \square \frac{dF}{dx}(I) \right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\square F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

ex) $F = \begin{bmatrix} x^2 - y - 1 \\ -\frac{1}{2}x^2 + y^2 \end{bmatrix}$ has a root $(\sqrt{2},1)$.

(But let us pretend not to know!)

$I = \begin{bmatrix} [1.2,1.6] \\ [0.8,1.2] \end{bmatrix}$ with $x = \begin{bmatrix} 1.4 \\ 1 \end{bmatrix}$ and $Y = JF(x)^{-1}$

Then, $K_{x,Y}(I) = \begin{bmatrix} [1.3476\cdots,1.4810\cdots] \\ [.89333\cdots,1.1067\cdots] \end{bmatrix} \subset I.$

Therefore, I contains a root of F (which is $(\sqrt{2},1)$),

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I) = x - YF(x) + \left(Id - Y\square\frac{dF}{dx}(I)\right)(I - x)$$

F : a square differentiable system of polynomial equations

I : an interval

$\square F(I) \supset \{F(x) \mid x \in I\}$: an interval extension of F over I

x : a point in I

Y : an invertible matrix

ex) $F = \begin{bmatrix} x^2 - y - 1 \\ -\frac{1}{2}x^2 + y^2 \end{bmatrix}$ has a root $(\sqrt{2}, 1)$.

(But let us pretend not to know!)

$I = \begin{bmatrix} [1.2, 1.6] \\ [0.8, 1.2] \end{bmatrix}$ with $x = \begin{bmatrix} 1.4 \\ 1 \end{bmatrix}$ and $Y = JF(x)^{-1}$

Then, $K_{x,Y}(I) = \begin{bmatrix} [1.3476\cdots, 1.4810\cdots] \\ [.89333\cdots, 1.1067\cdots] \end{bmatrix} \subset I.$

Therefore, I contains a root of F (which is $(\sqrt{2}, 1)$), uniquely.

Q. Can we use the Krawczyk method for certified homotopy tracking?

Q. Can we use the Krawczyk method for certified homotopy tracking?

A. Consider the Krawczyk method **“for a system with parameters”**

Krawczyk method

Combine interval arithmetic and Newton's method.
Define the **Krawczyk operator**

$$K_{x,Y}(I,T) = x - Y \square H(x,T) + \left(Id - Y \square \frac{\partial H(I,T)}{\partial x} \right)(I - x)\,.$$

$H(x,t)$: a homotopy with a parameter t from $\mathbb{C}^n \times [0,1]$ to $\mathbb{C}^n$

(e.g. $H(x,t) = t\gamma G(x) + (1-t)F(x)$)

I : an interval

T : an interval contained in $[0,1]$

Krawczyk method

Combine interval arithmetic and Newton's method. Define the **Krawczyk operator**

$$K_{x,Y}(I,T) = x - Y \,\square\, H(x,T) + \left(Id - Y \,\square\, \frac{\partial H(I,T)}{\partial x}\right)(I - x)\,.$$

$H(x,t)$: a homotopy with a parameter t from $\mathbb{C}^n \times [0,1]$ to $\mathbb{C}^n$

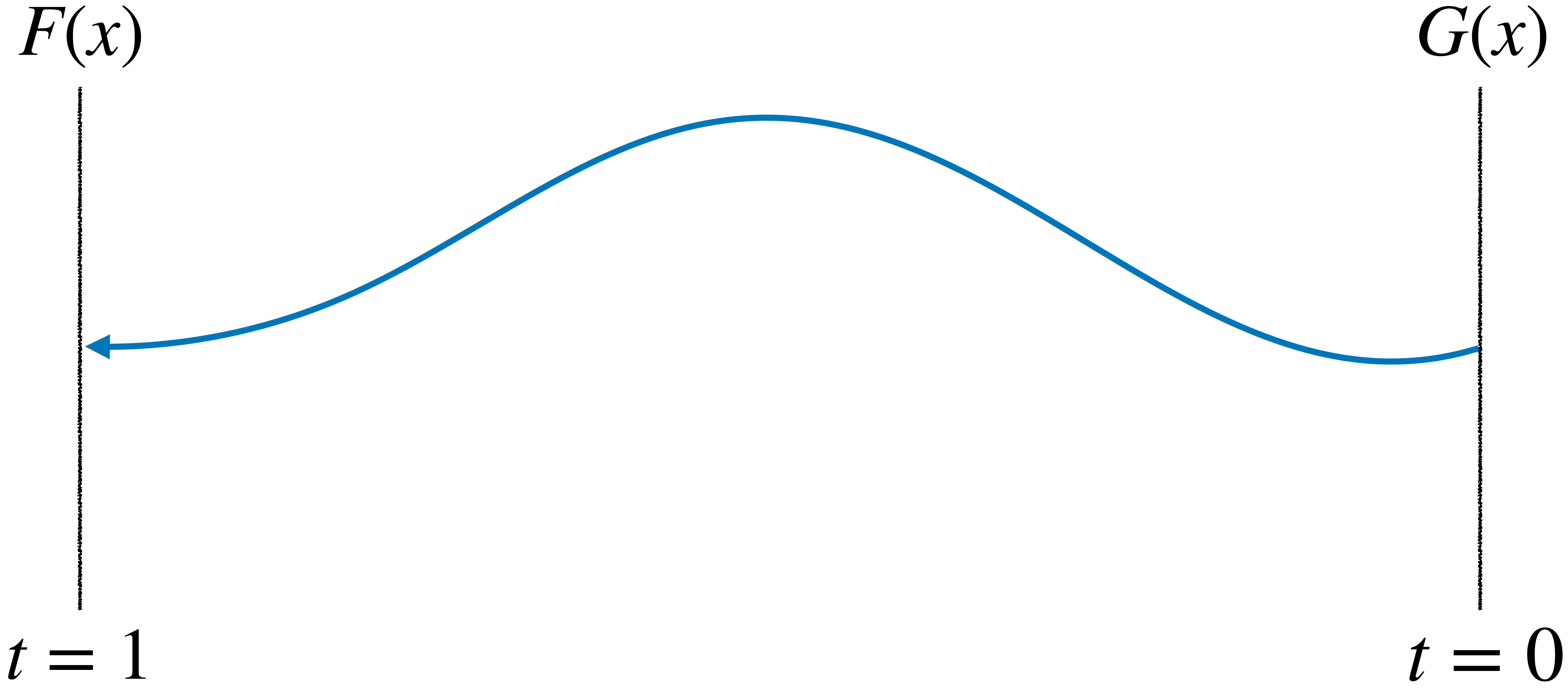
(e.g. $H(x,t) = t\gamma G(x) + (1-t)F(x)$)

I : an interval

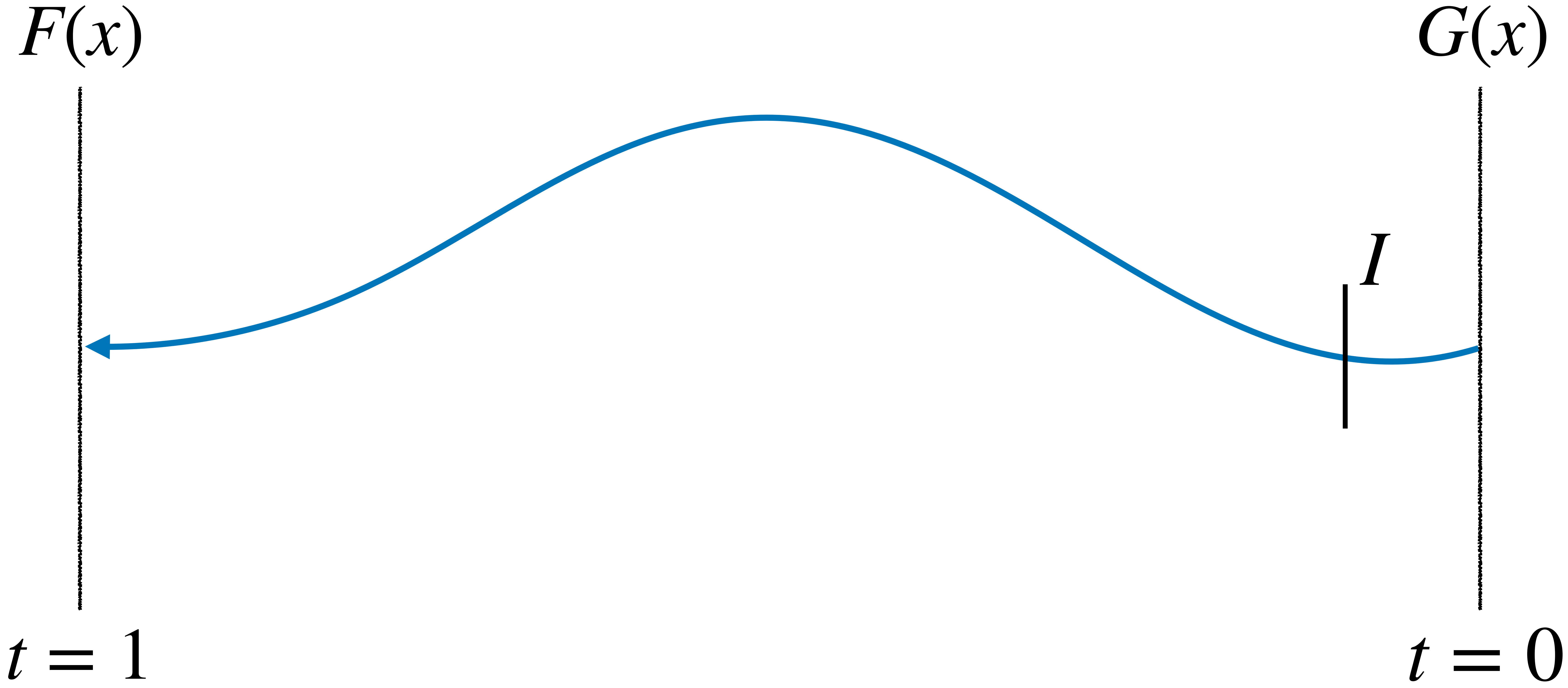
T : an interval contained in $[0,1]$

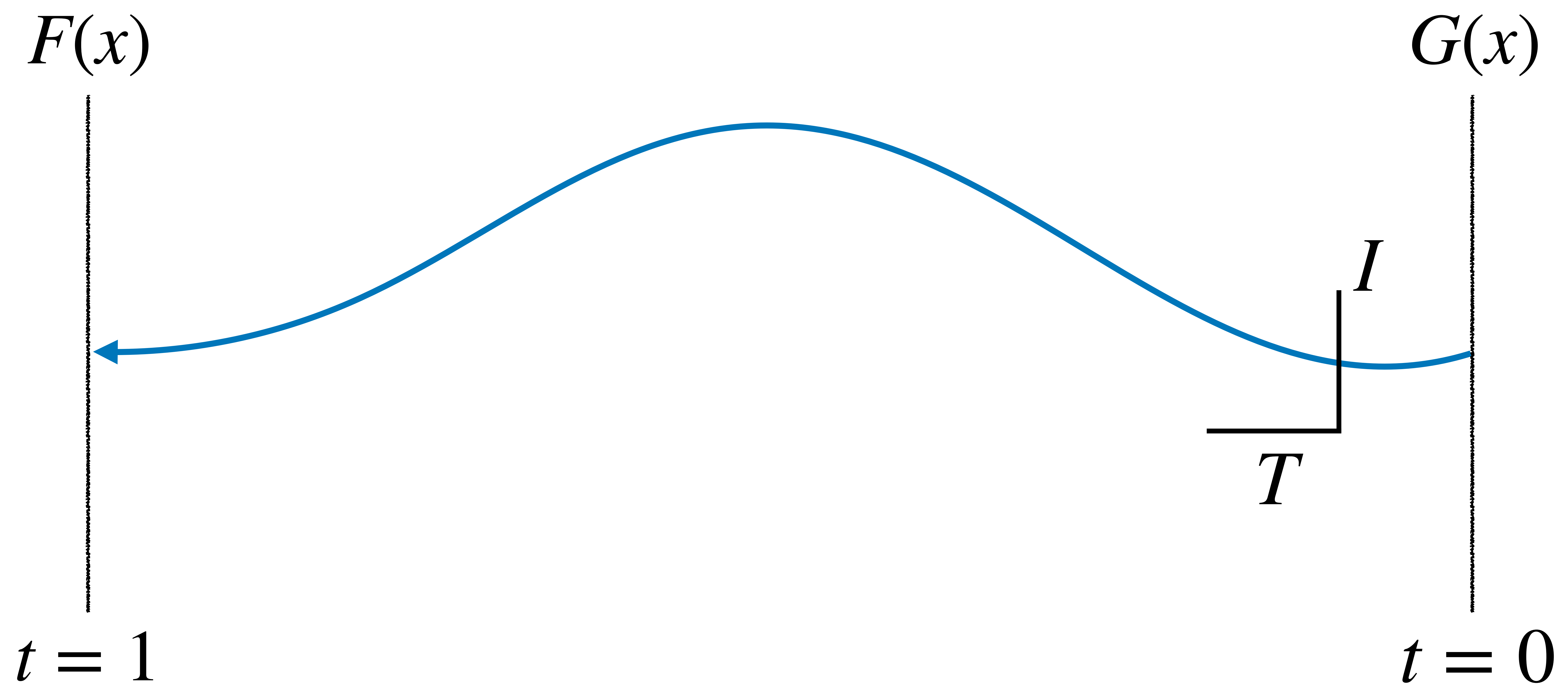
Theorem (Duff-L.) For the Krawczyk operator, the following hold :

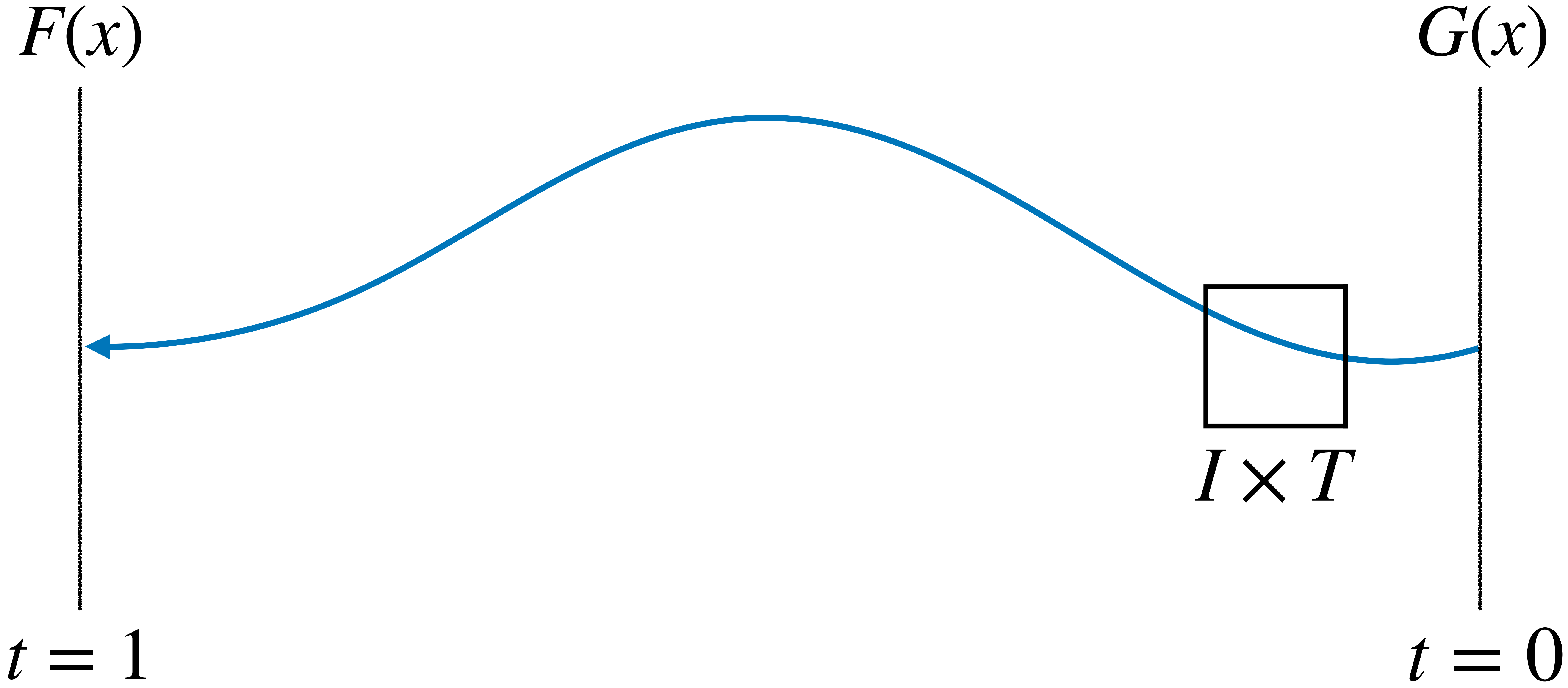
1. If $K_{x,Y}(I,T) \subset I$, then there is a solution of $H(x,t)$ in I for all $t \in T$ (**existence** of a solution path)
2. If I has a solution and $$\sqrt{2}\left\| Id - Y \,\square\, \frac{\partial H(I,T)}{\partial x} \right\| < 1,$$ then for all $t \in T$, there is a solution of F in I, and it is unique (**uniqueness** of a solution path)

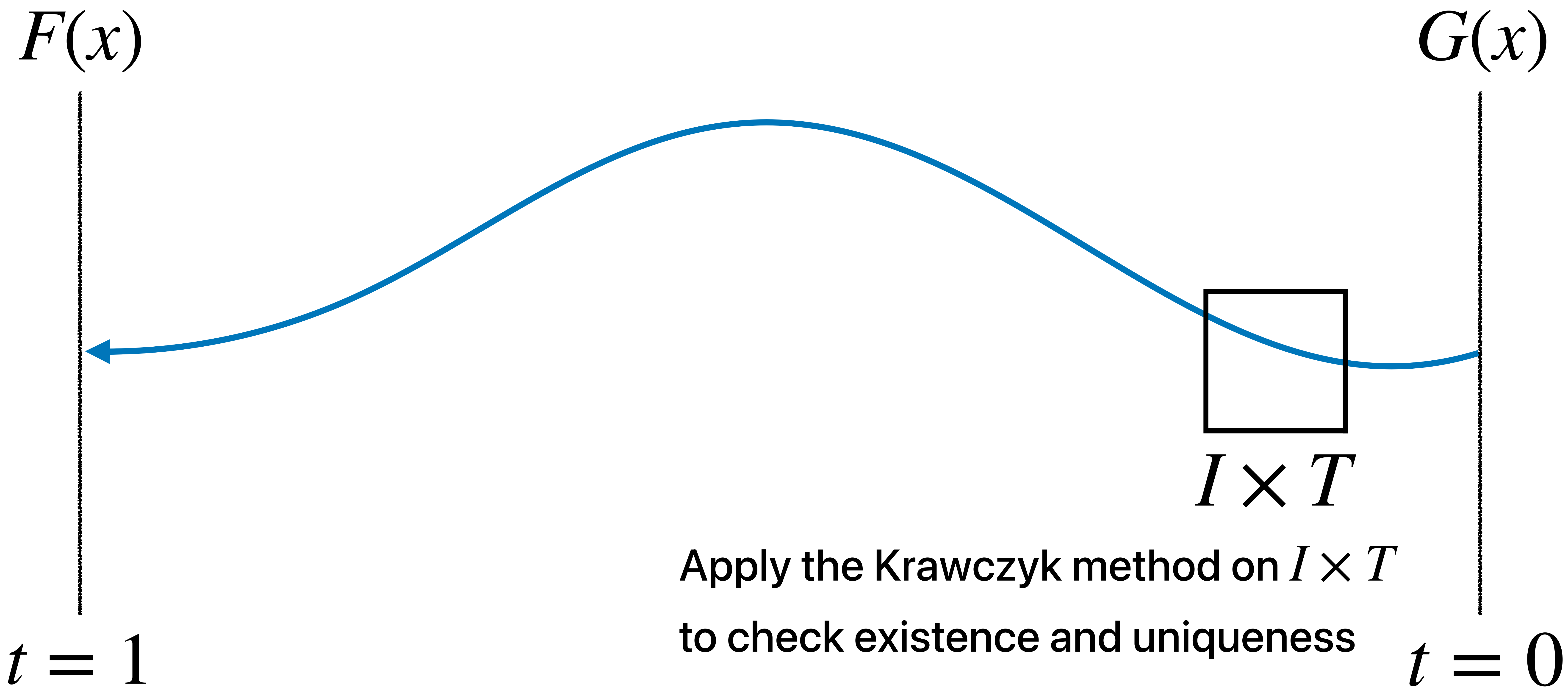


$F(x)$
$G(x)$
$t = 1$
$t = 0$

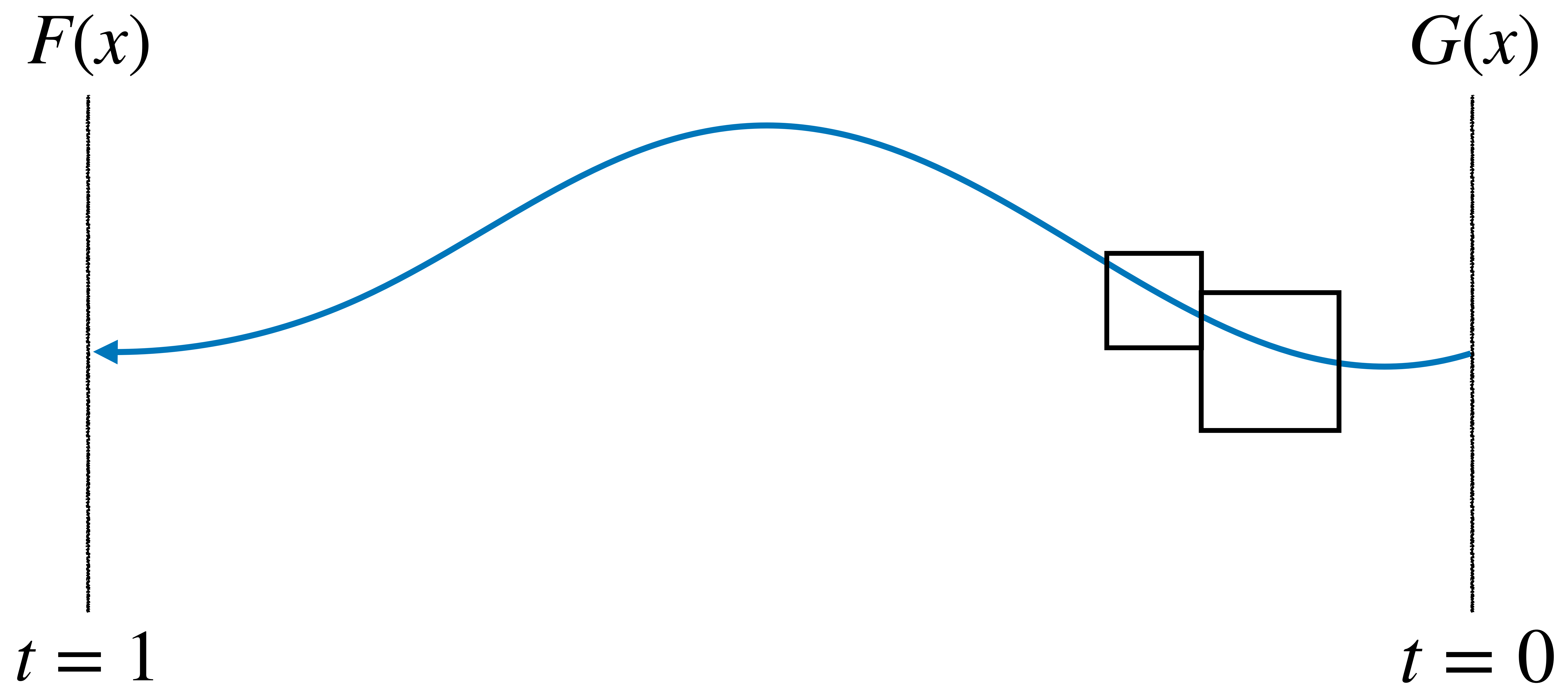

$F(x)$
$G(x)$
I
$t = 1$
$t = 0$


$F(x)$
$G(x)$
I
T
$t = 1$
$t = 0$

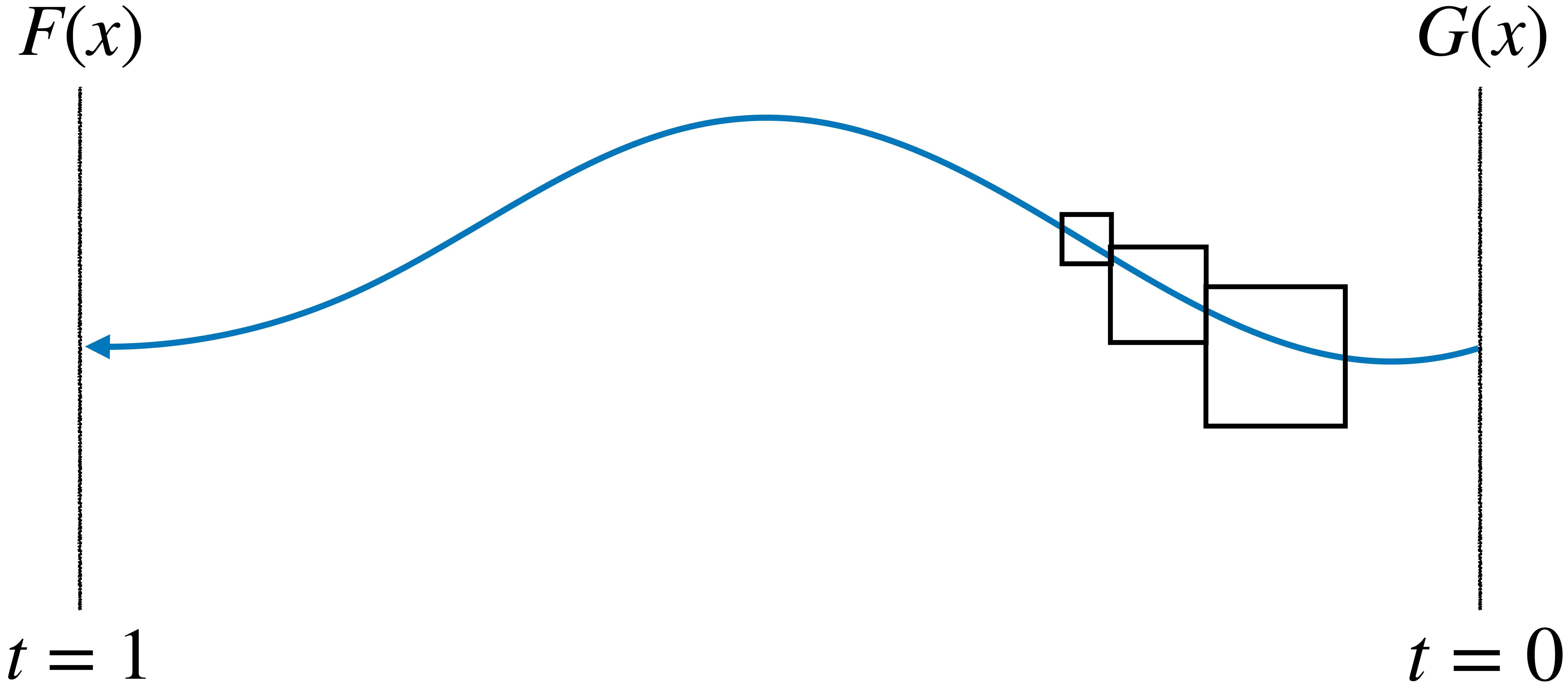

$F(x)$
$G(x)$
$I \times T$
$t = 1$
$t = 0$

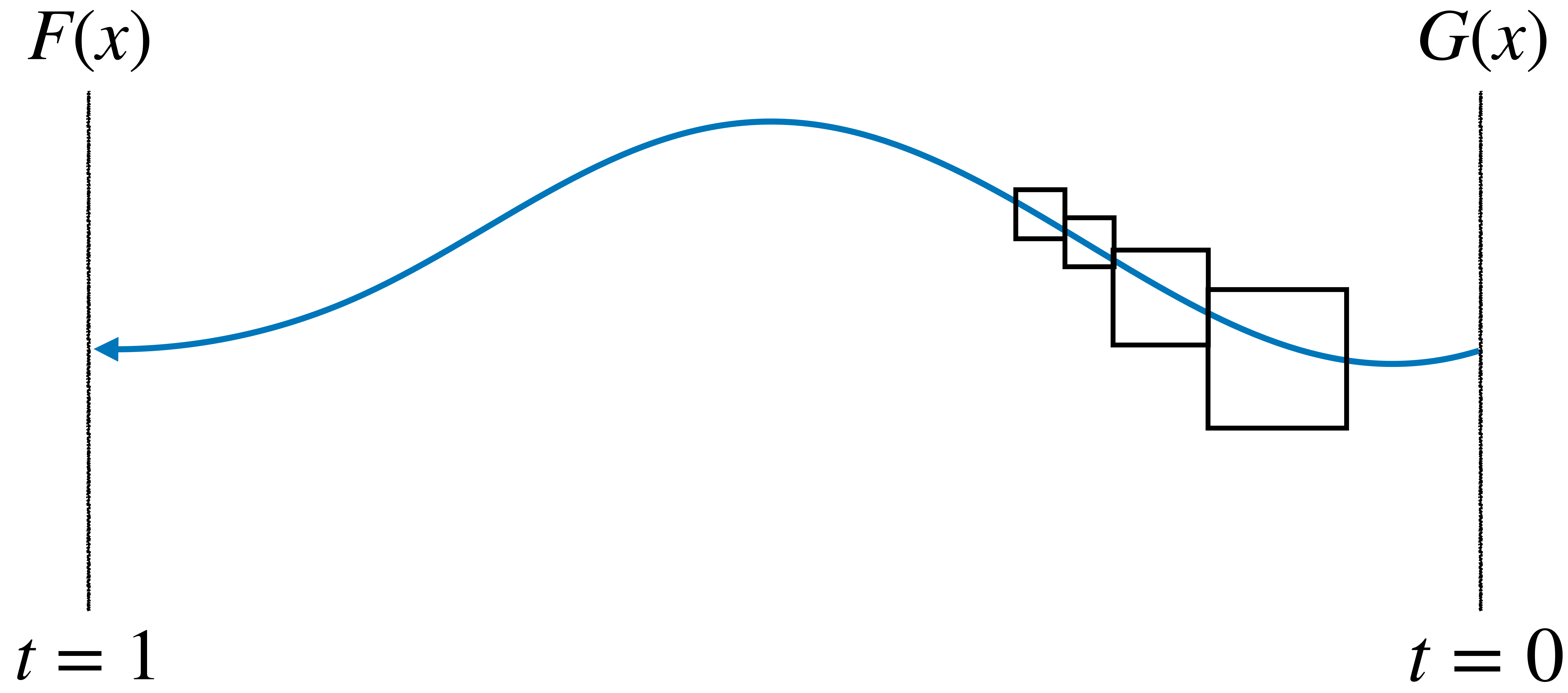


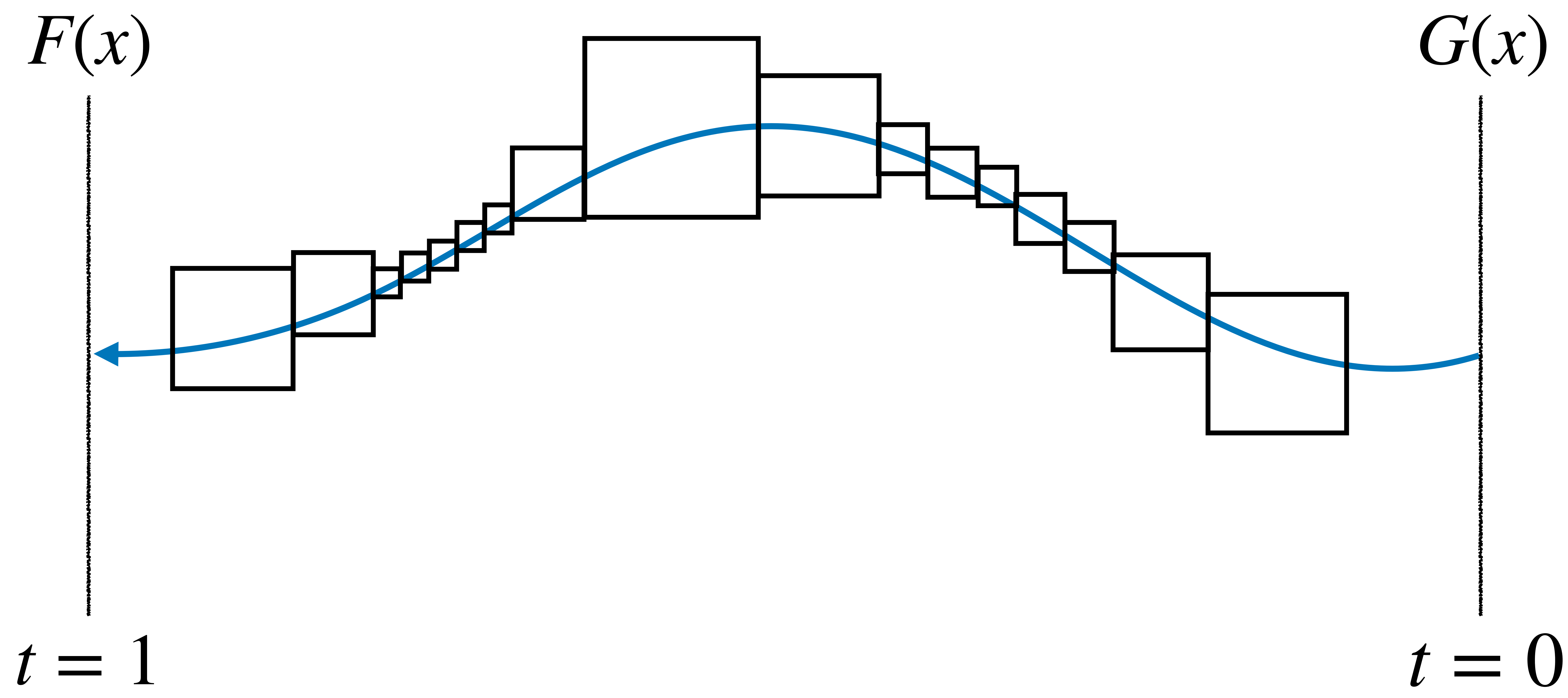
$F(x)$
$G(x)$
$I \times T$
Apply the Krawczyk method on $I \times T$
to check existence and uniqueness
$t = 1$
$t = 0$

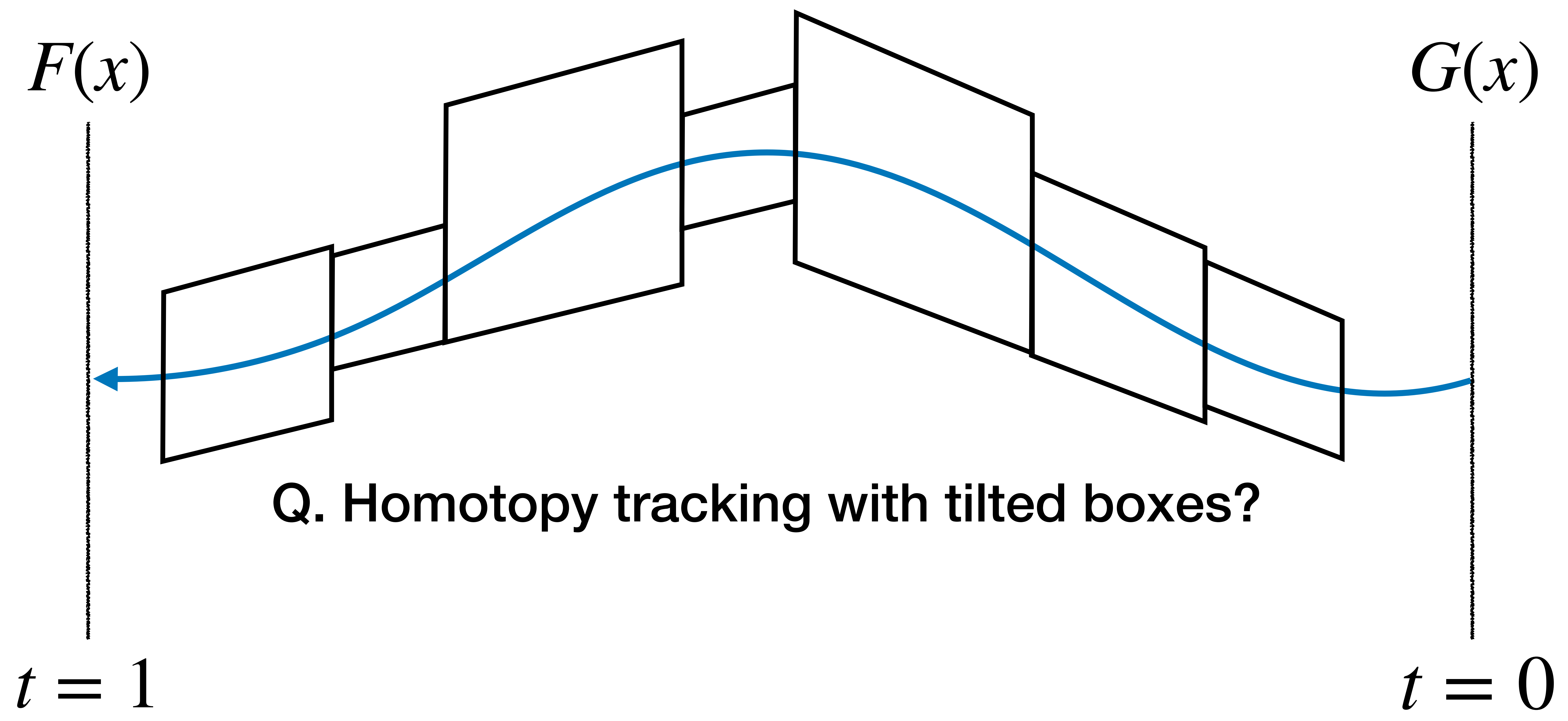


$F(x)$
$G(x)$
$t = 1$
$t = 0$


$F(x)$
$G(x)$
$t = 1$
$t = 0$


$F(x)$
$G(x)$
$t = 1$
$t = 0$

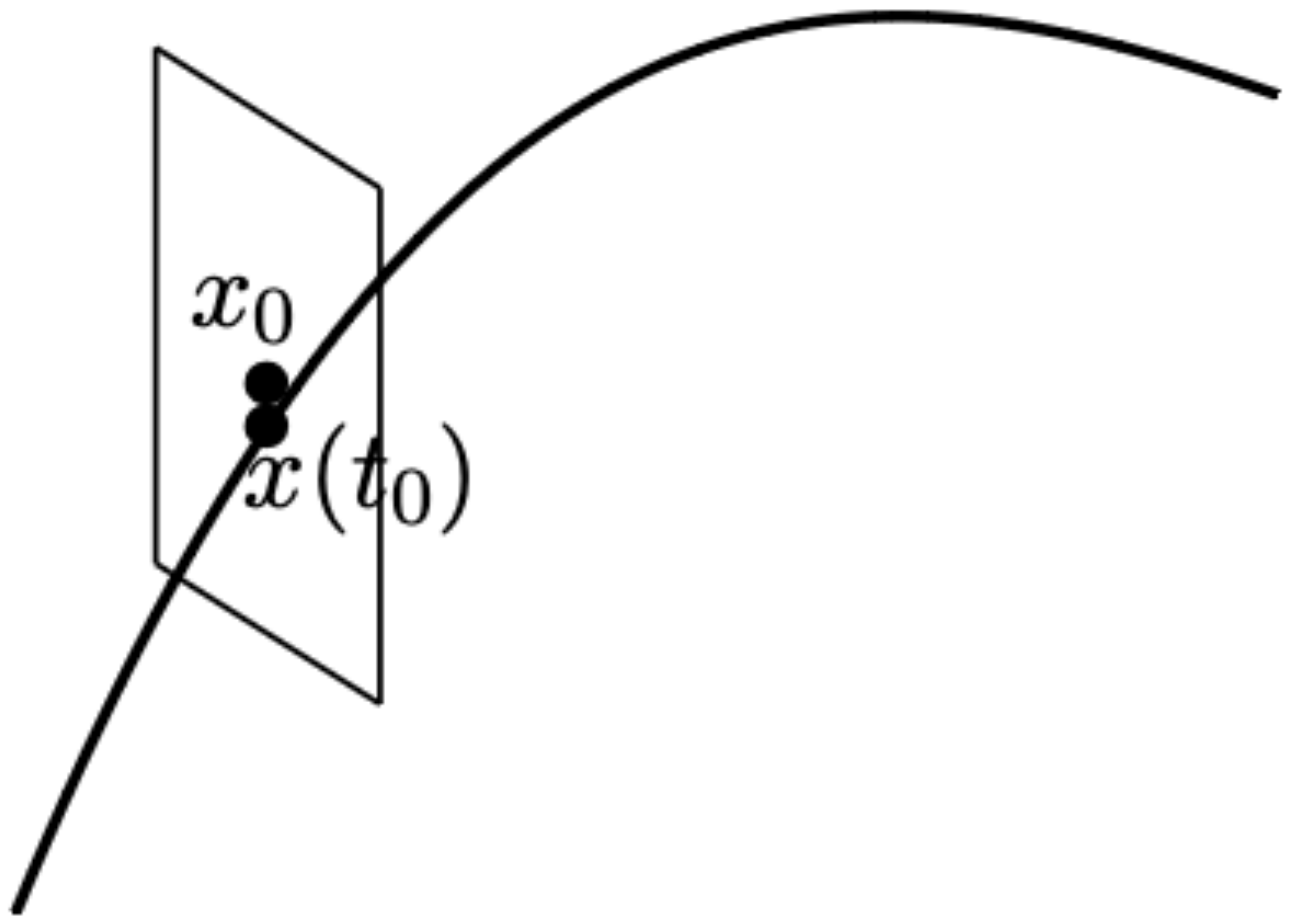

$F(x)$
$G(x)$
$t = 1$
$t = 0$



$F(x)$
$G(x)$
Q. Homotopy tracking with tilted boxes?
$t = 1$
$t = 0$

Preconditioning

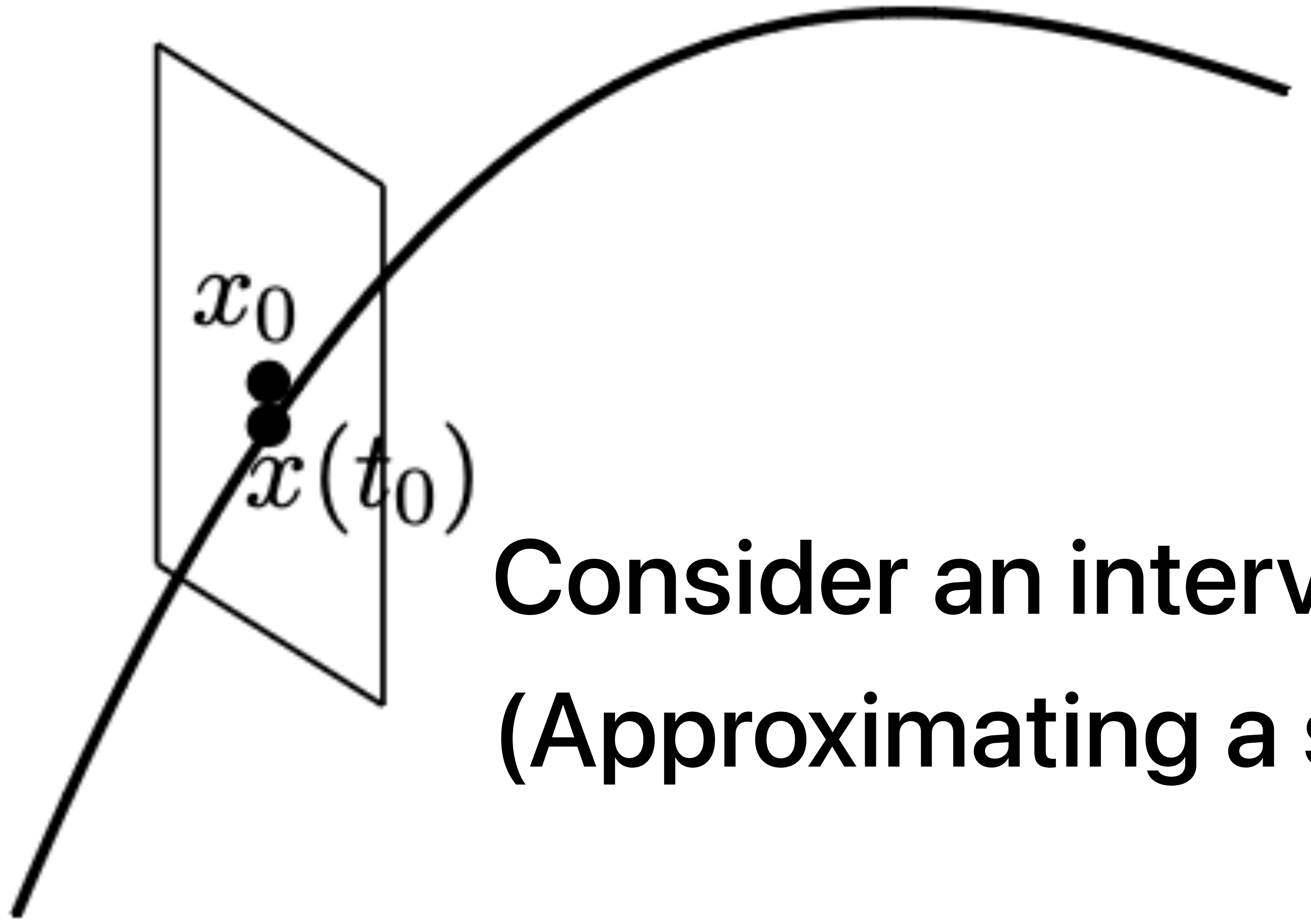
Constructing a tilted interval box



Solution path $x(t)$
from $t = 0$ to $t = 1$

Preconditioning

Constructing a tilted interval box



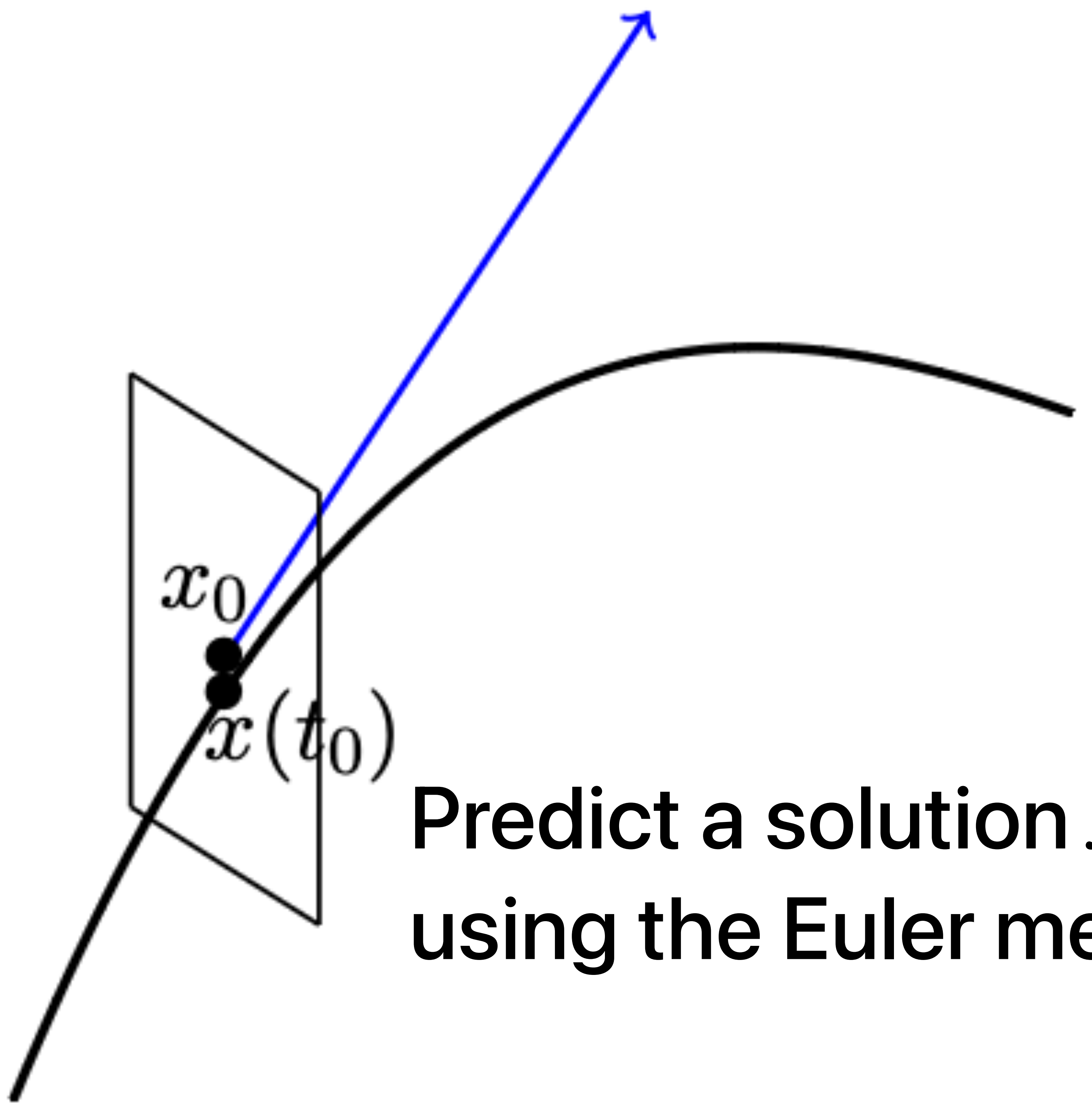
Solution path $x(t)$
from $t = 0$ to $t = 1$

Consider an interval box I centered at x_0

(Approximating a solution at $t = t_0$)

Preconditioning

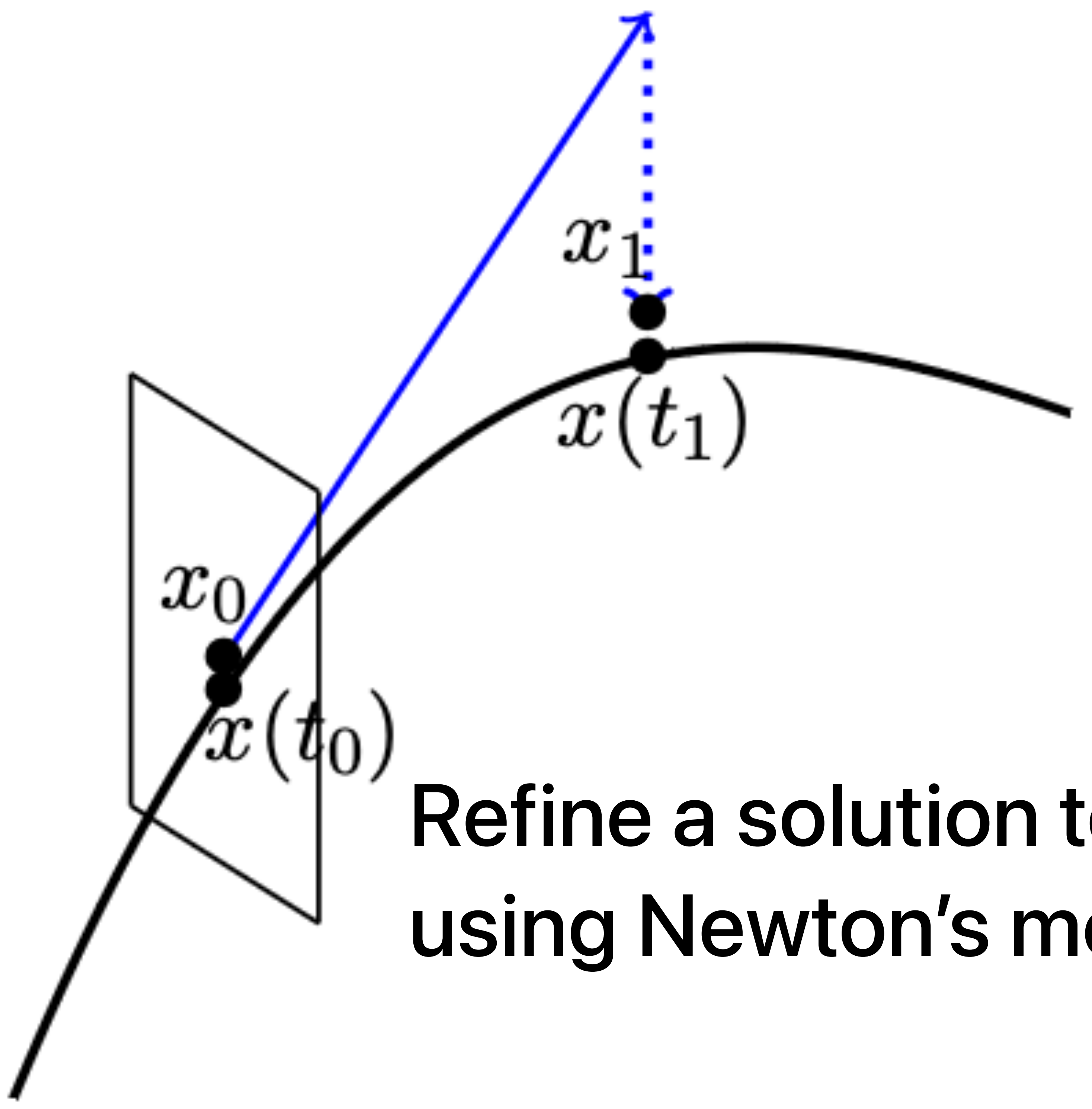
Constructing a tilted interval box



Predict a solution $x(t_1)$
using the Euler method for example

Preconditioning

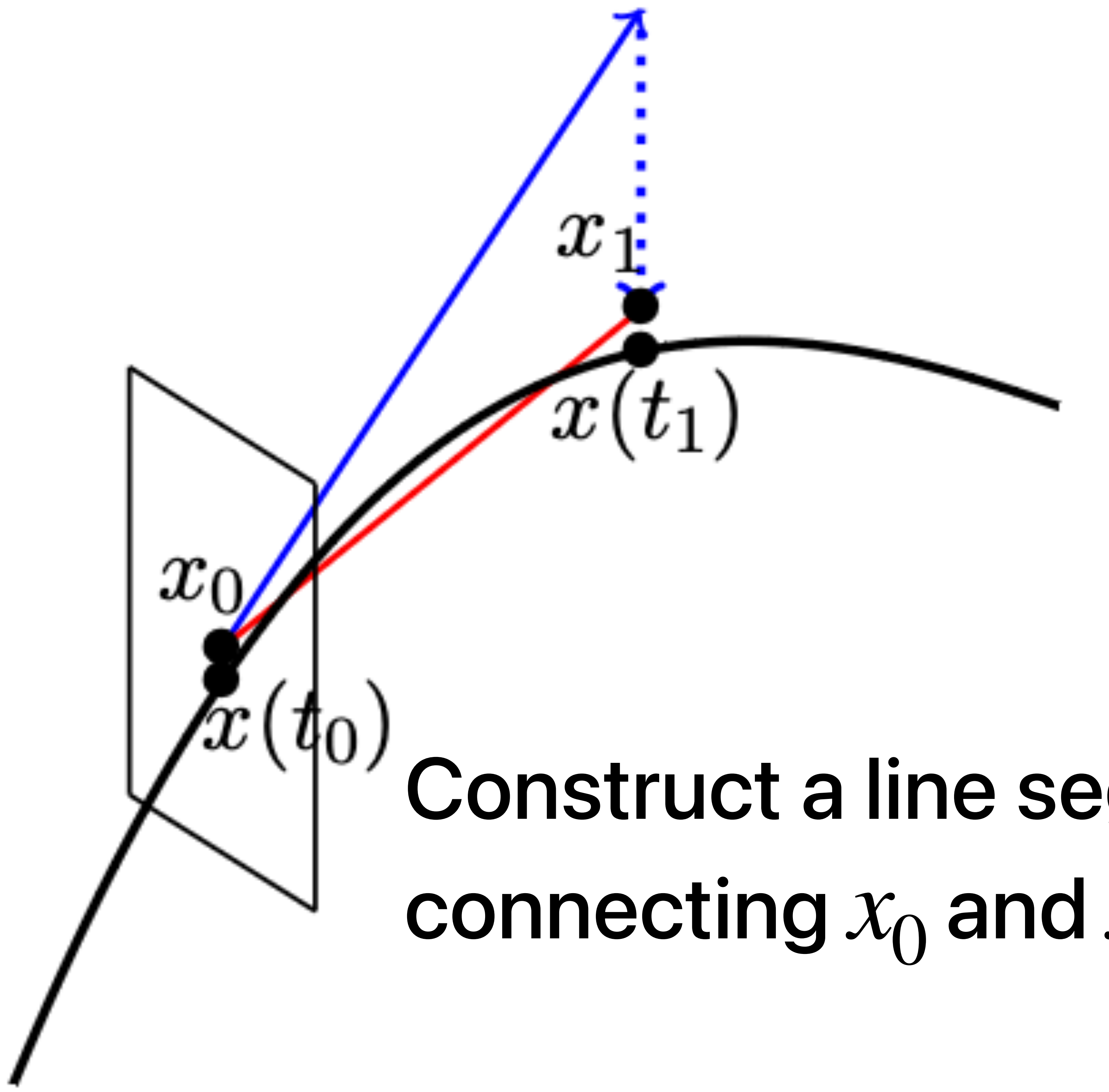
Constructing a tilted interval box



Refine a solution to get x_1
using Newton's method

Preconditioning

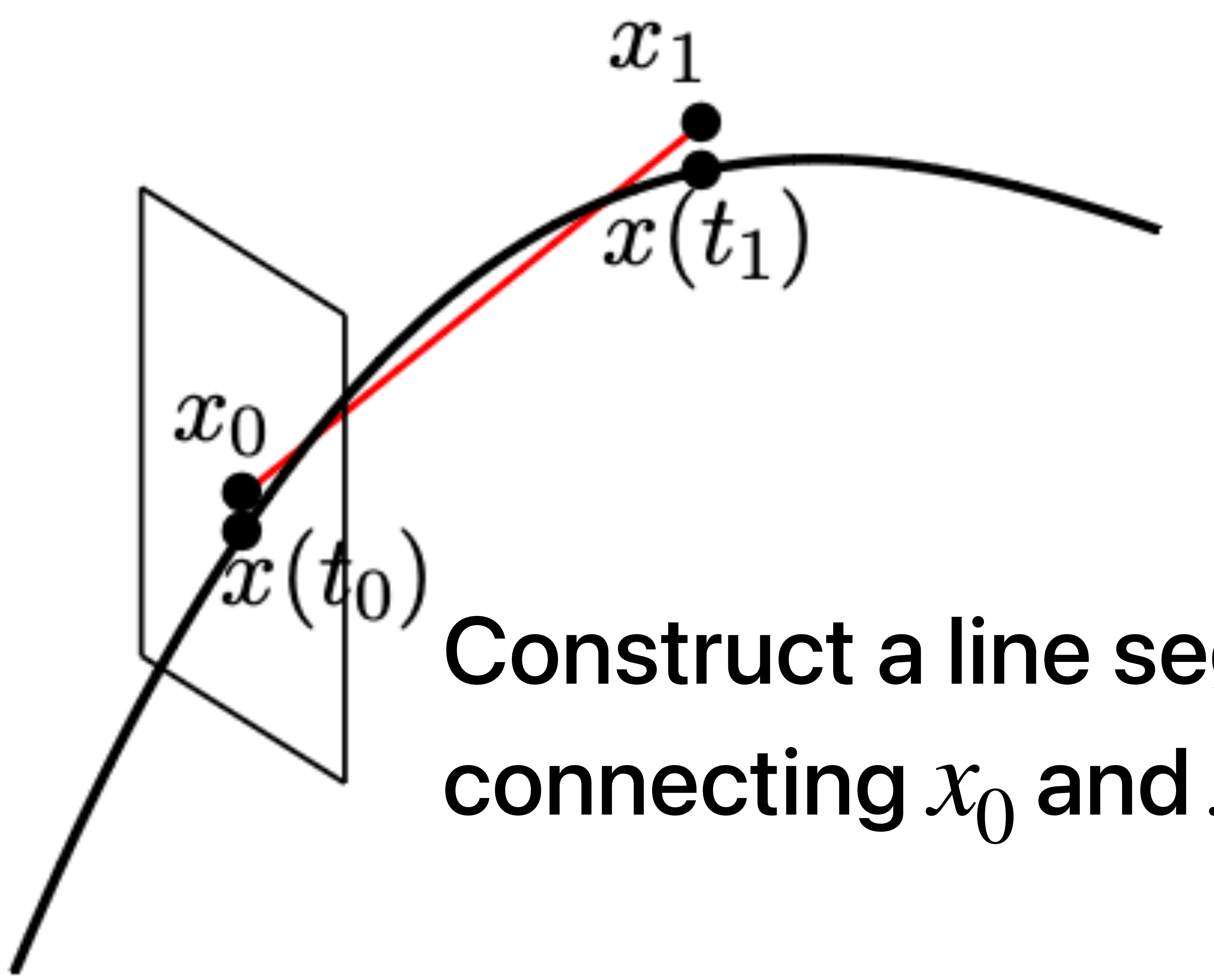
Constructing a tilted interval box



Construct a line segment connecting x_0 and x_1

Preconditioning

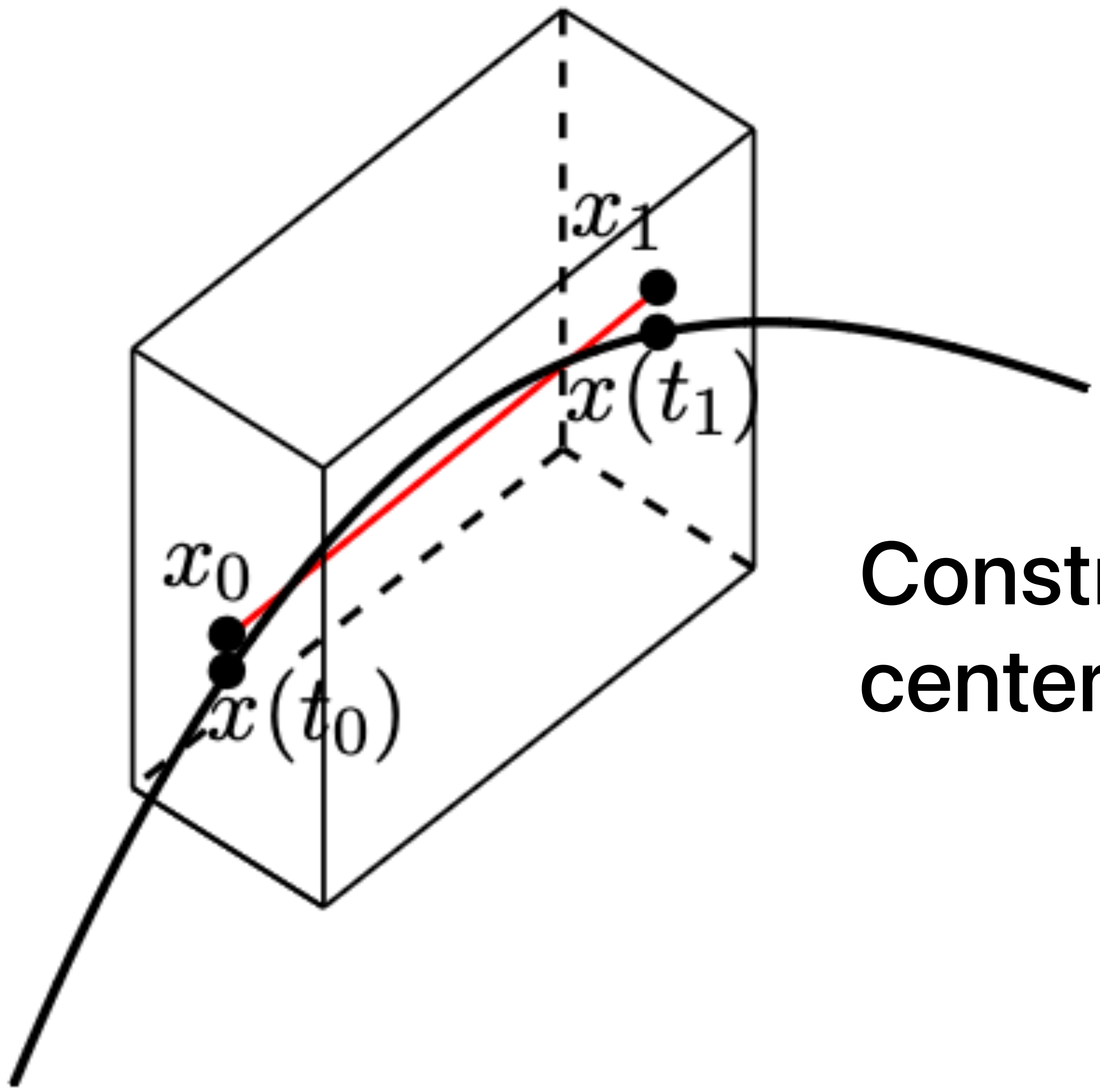
Constructing a tilted interval box



Construct a line segment connecting x_0 and x_1

Preconditioning

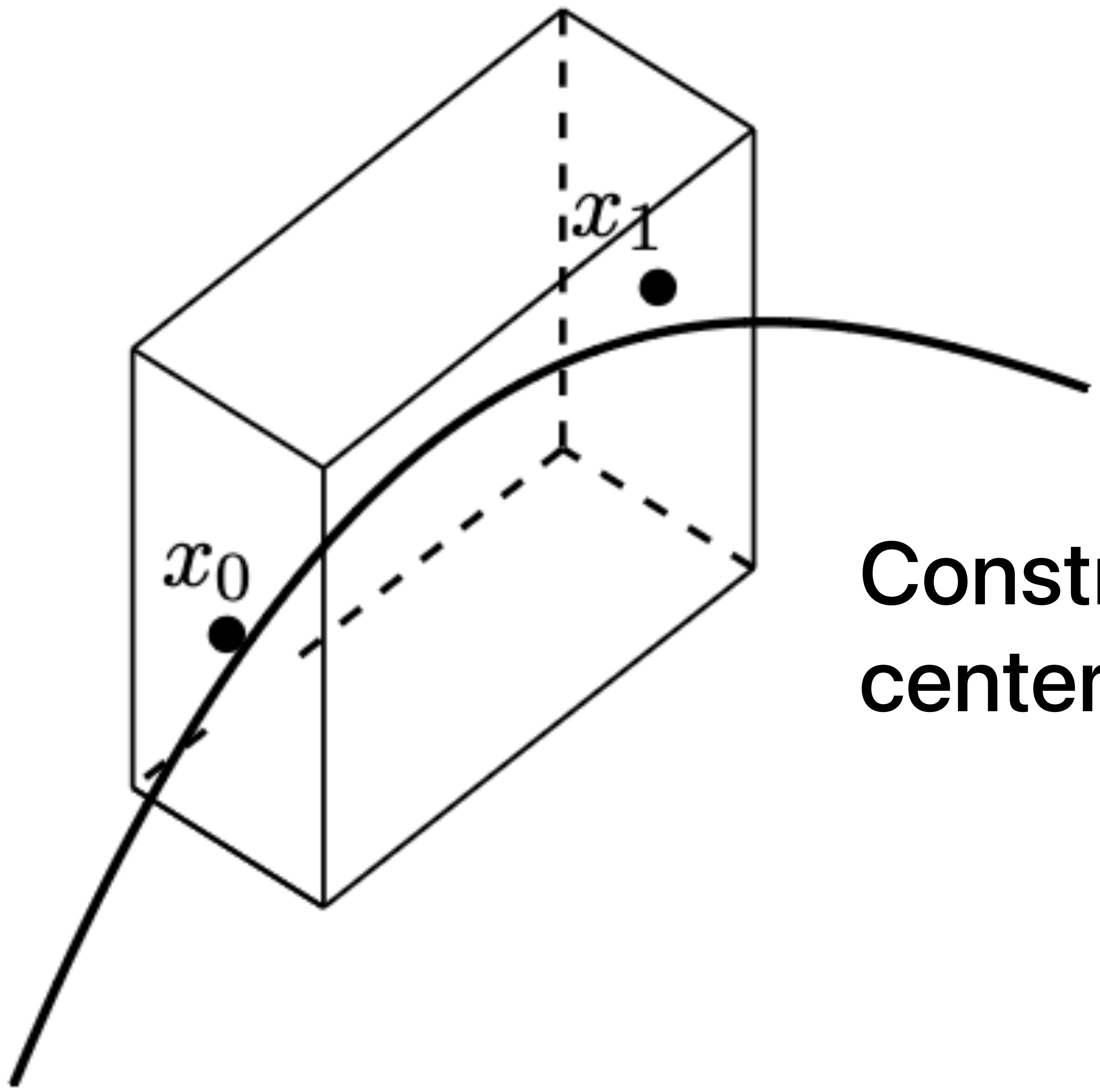
Constructing a tilted interval box



Construct a tilted interval box
centered at the line segment

Preconditioning

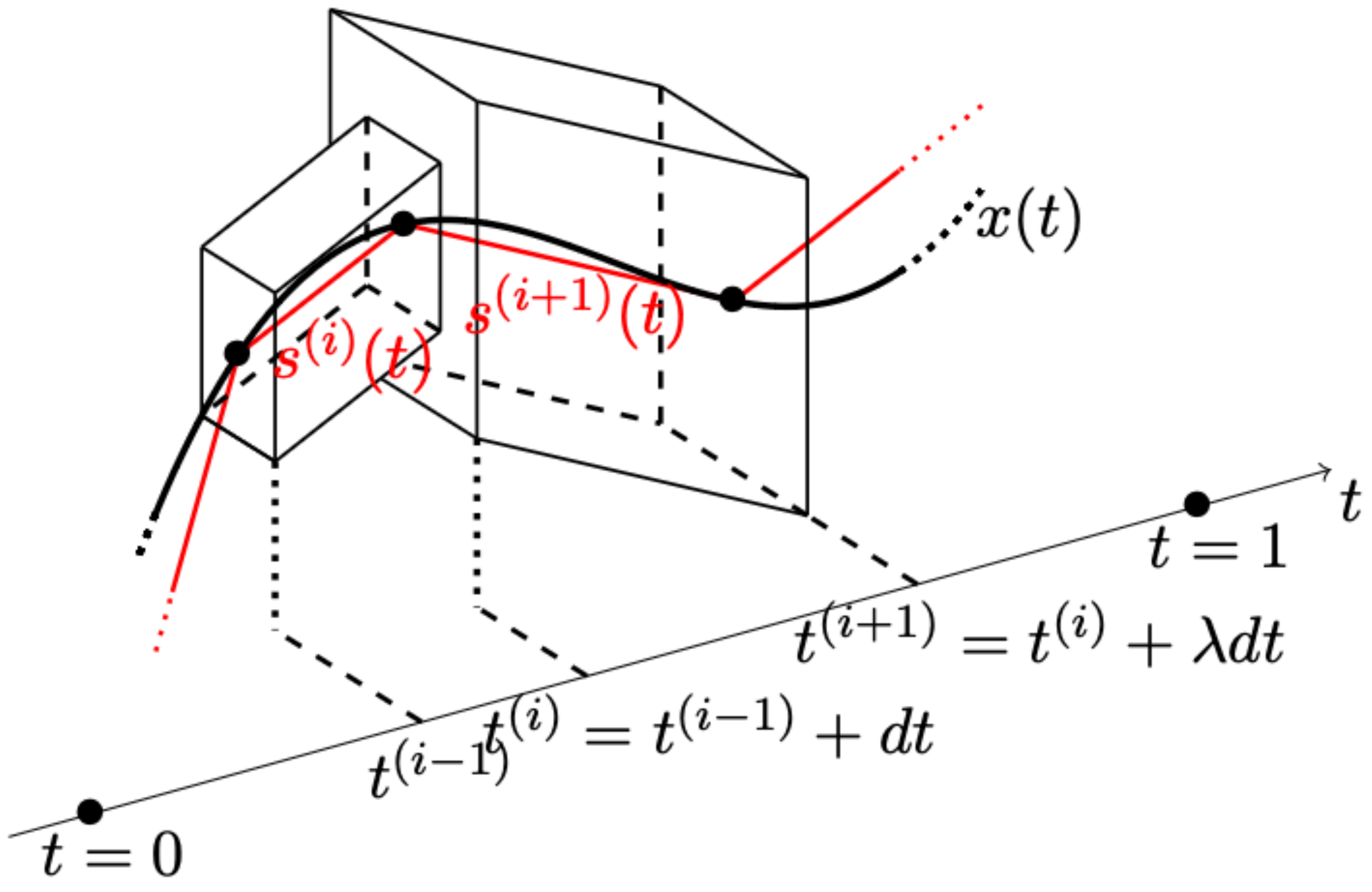
Constructing a tilted interval box



Construct a tilted interval box centered at the line segment

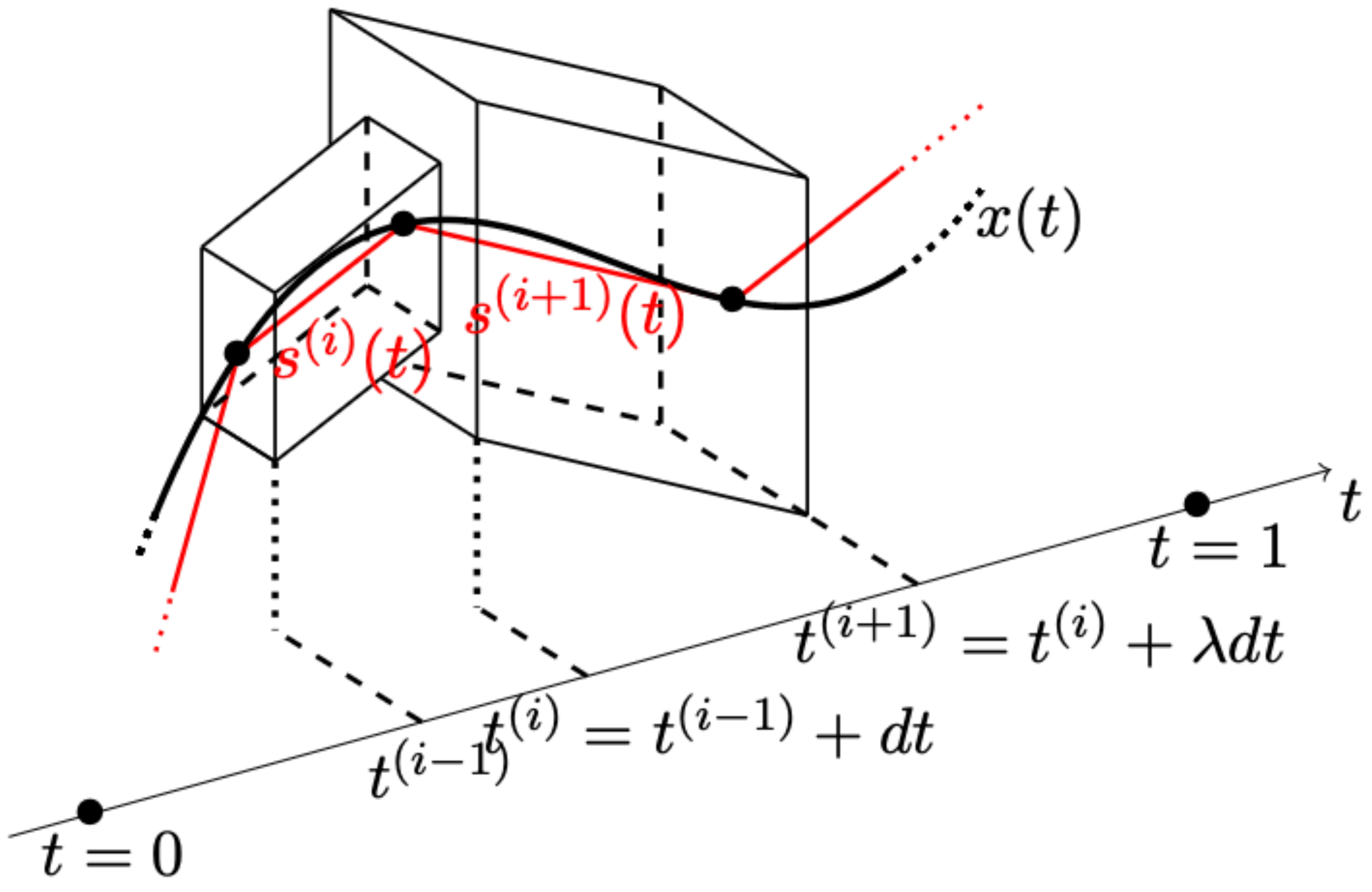
Preconditioning

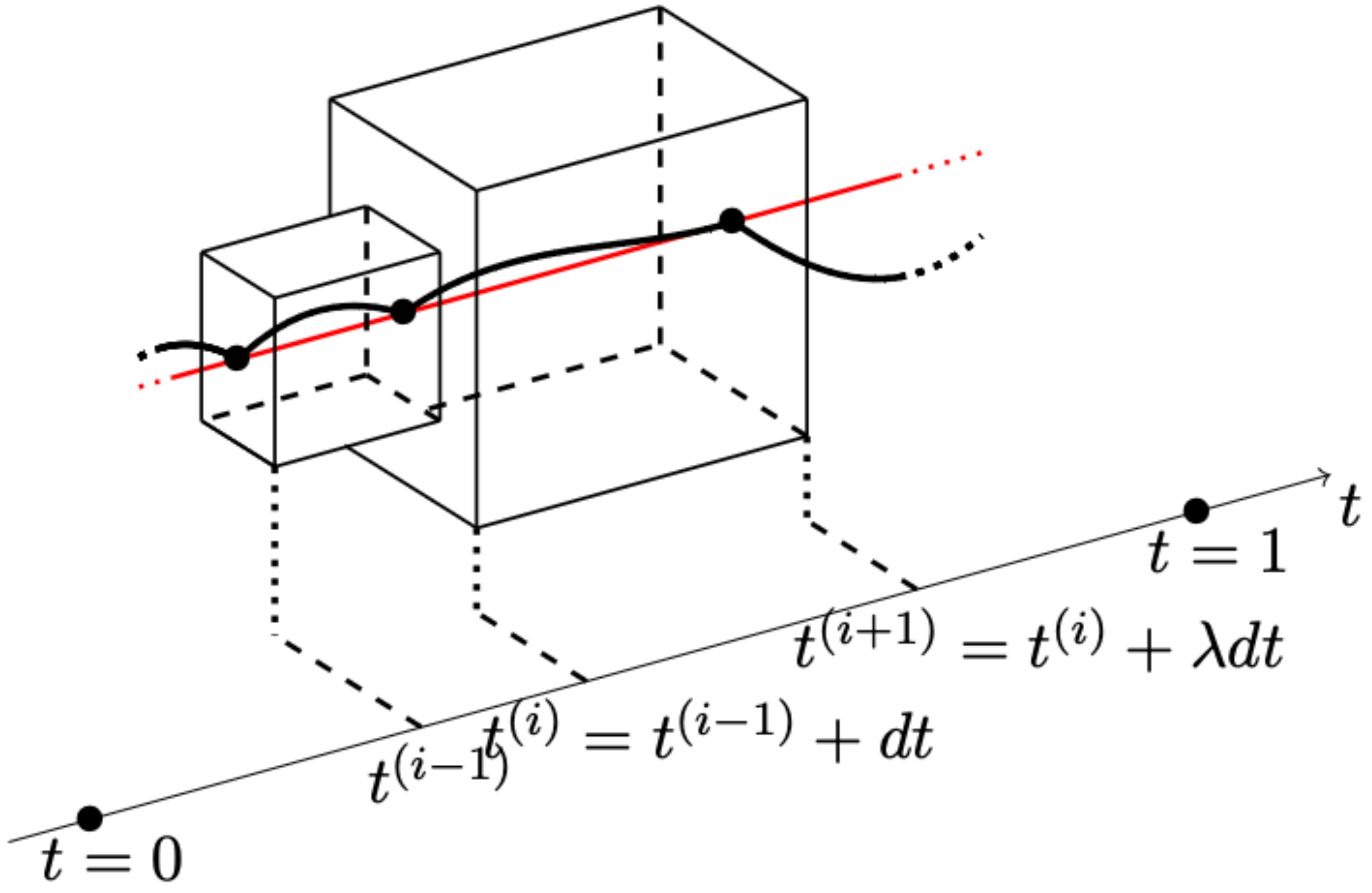
Constructing a tilted interval box



Preconditioning

Constructing a tilted interval box





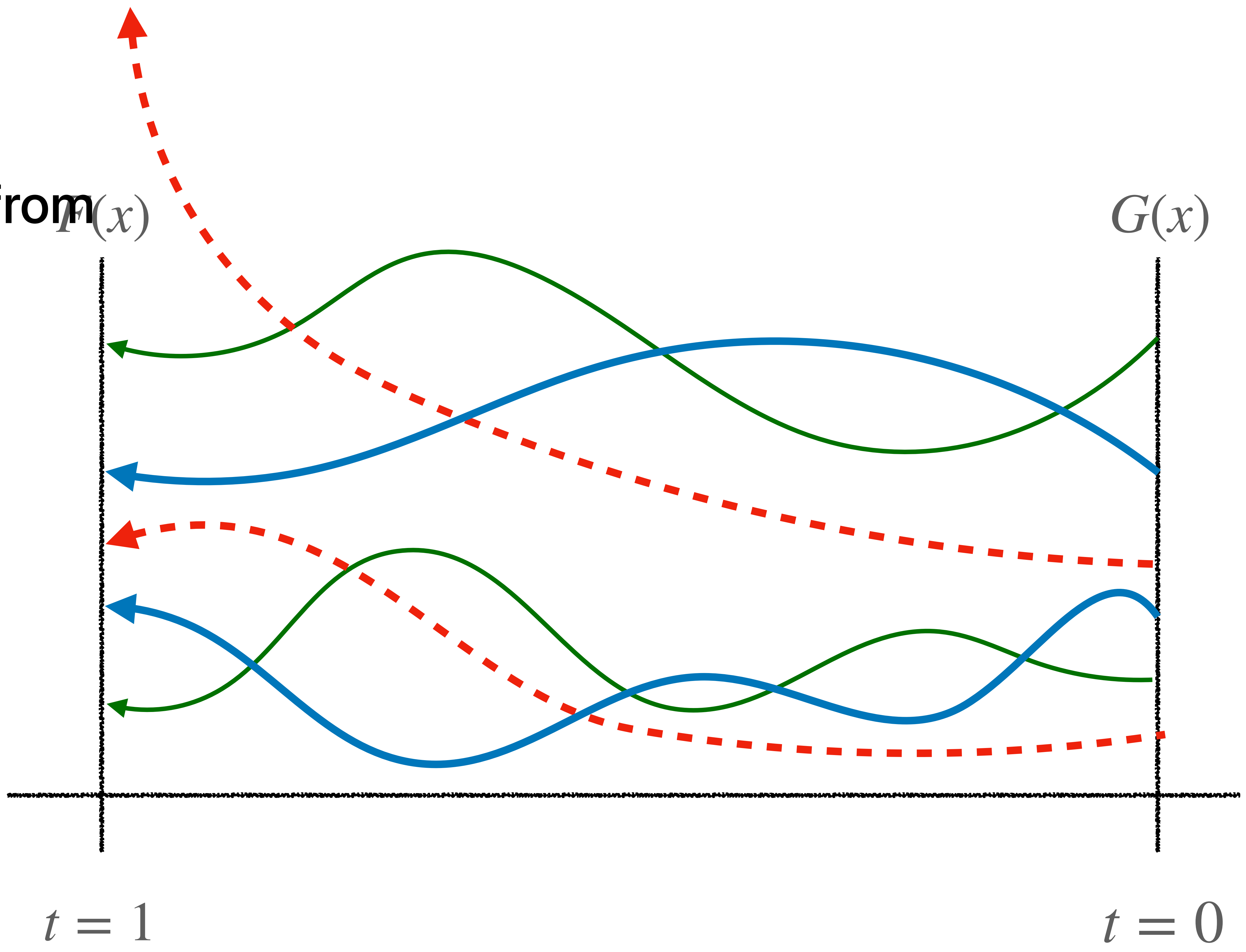
After the construction, apply the shearing map to make the rectangular interval box to avoid overestimation from interval arithmetic

Algorithm

Algorithm

Input :

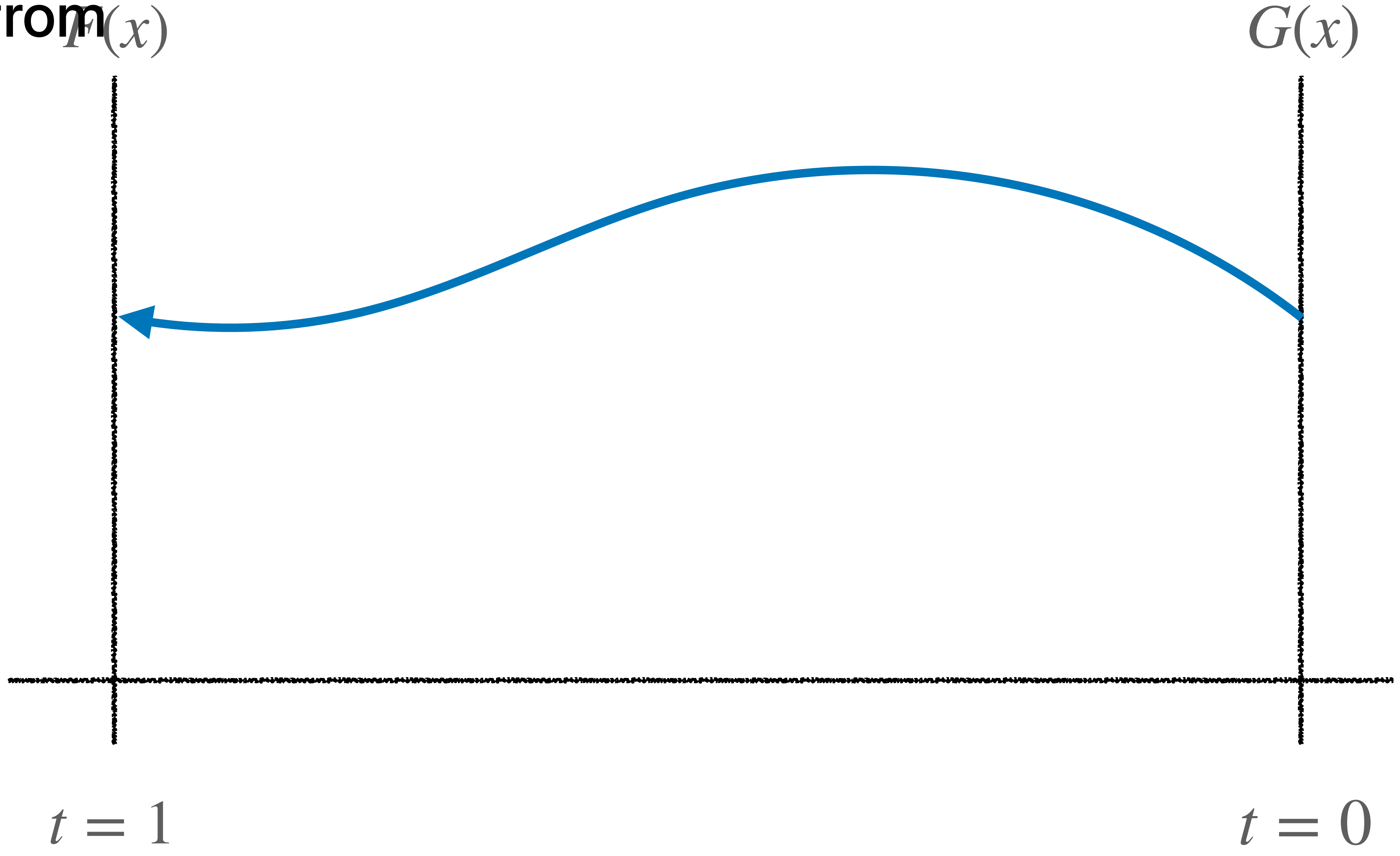
- $H(x,t) = F(x;p(t))$ a parameter homotopy from $t=0$ to $t=1$



Algorithm

Input :

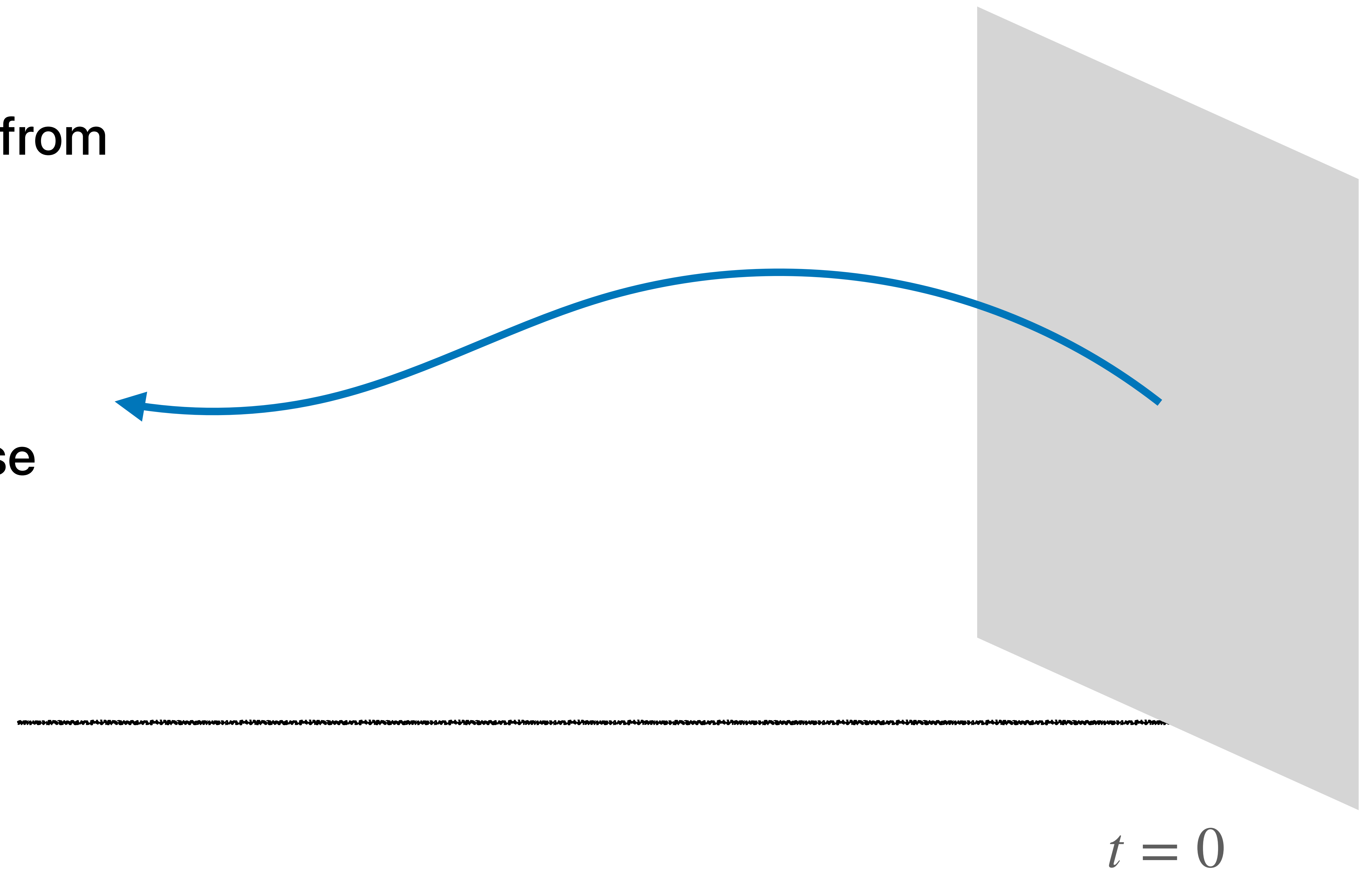
- $H(x,t) = F(x;p(t))$ a parameter homotopy from $t=0$ to $t=1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$



Algorithm

Input :

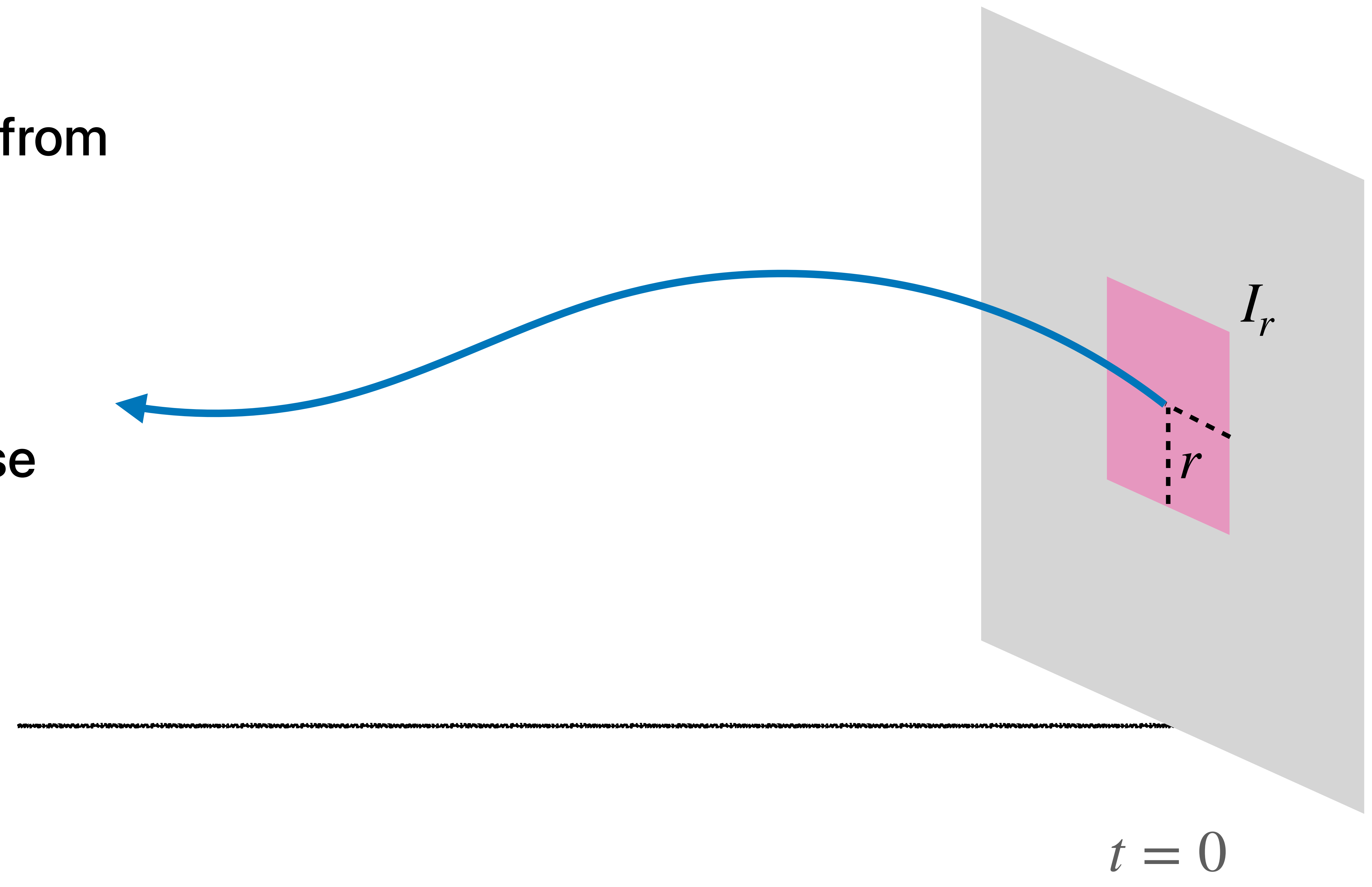
- $H(x,t) = F(x; p(t))$ a parameter homotopy from $t=0$ to $t=1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$



Algorithm

Input :

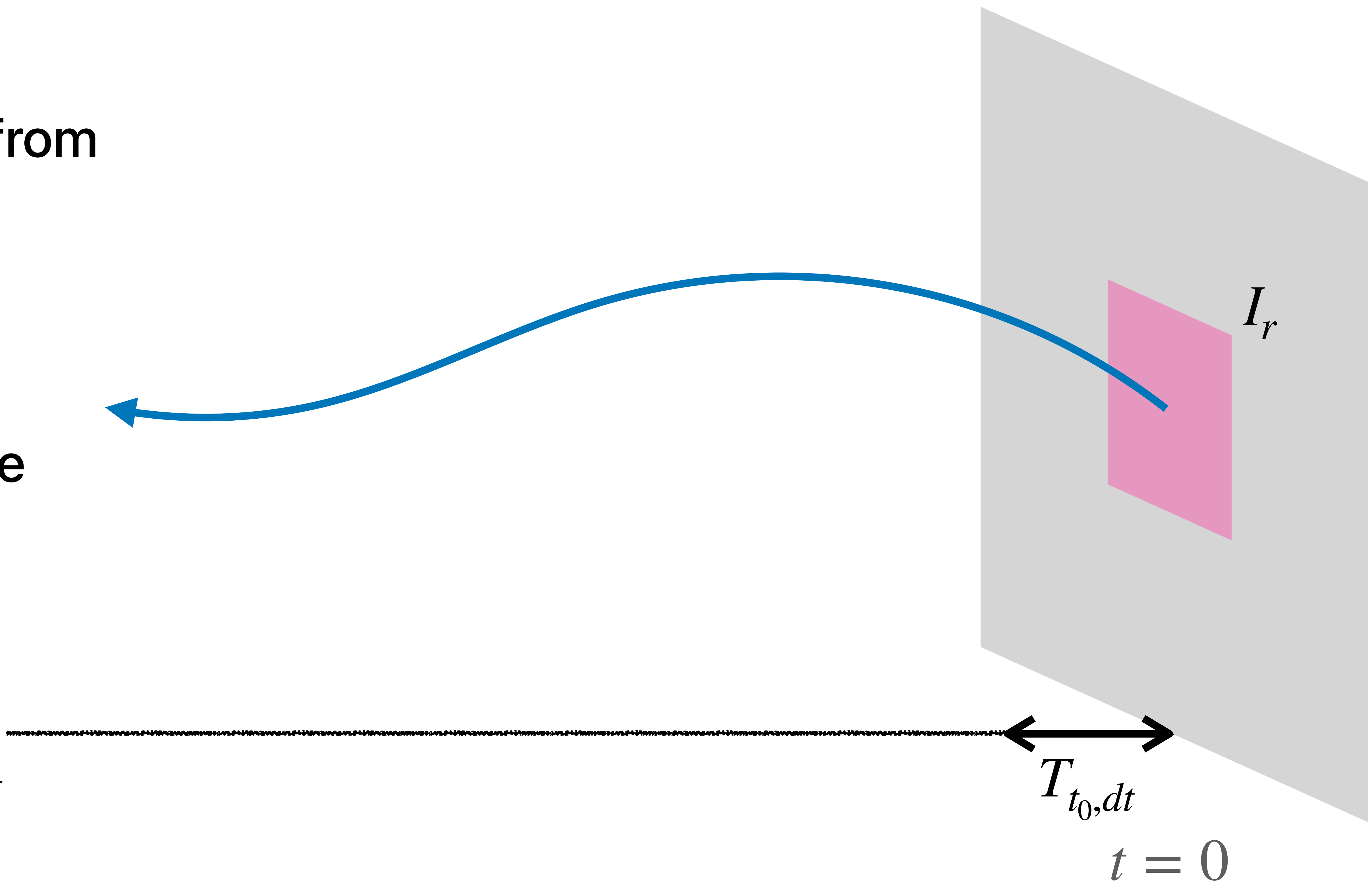
- $H(x, t) = F(x; p(t))$ a parameter homotopy from $t = 0$ to $t = 1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$



Algorithm

Input :

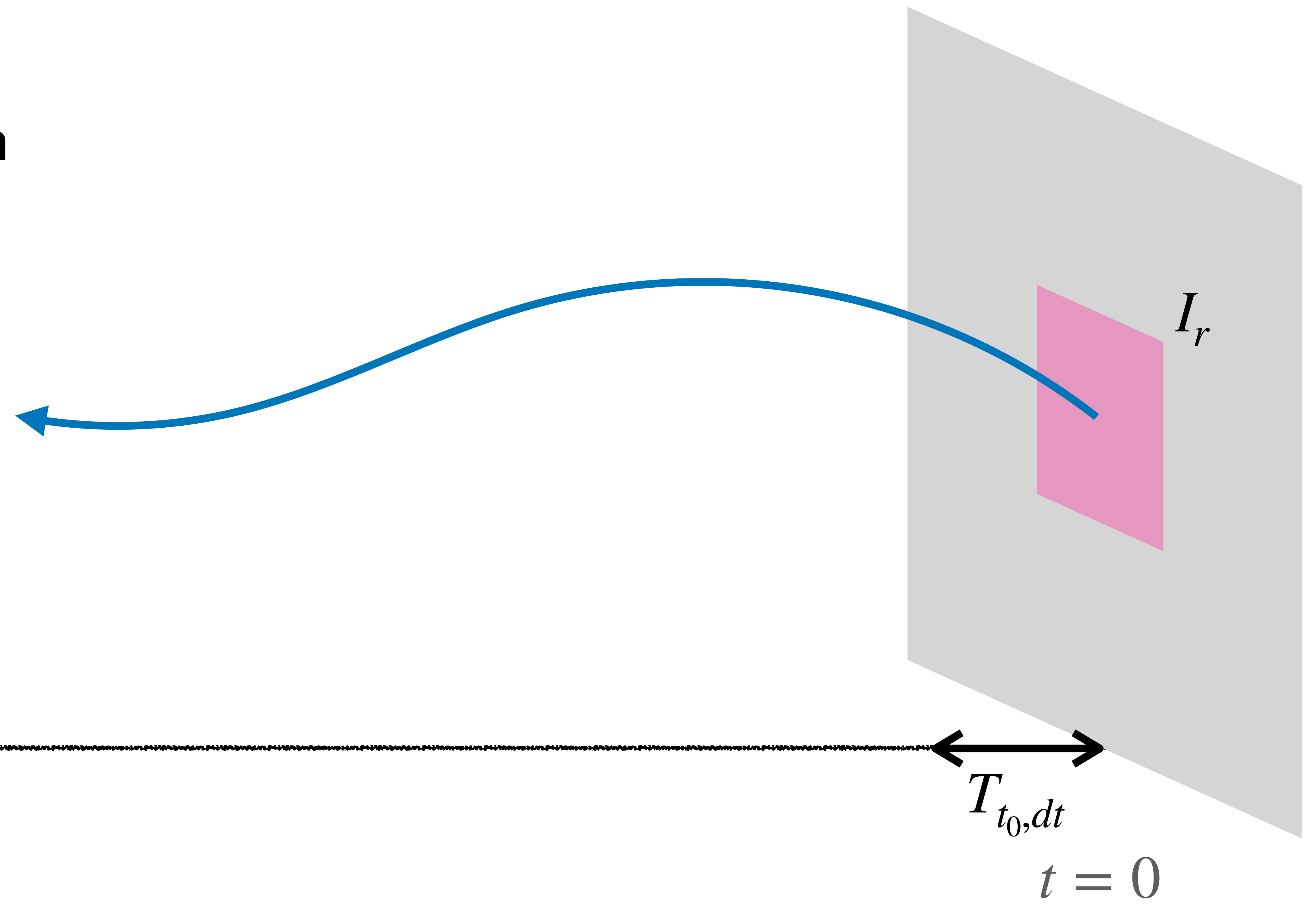
- $H(x, t) = F(x; p(t))$ a parameter homotopy from $t = 0$ to $t = 1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$
- $dt \in (0,1)$: the initial step size
- $T_{t_0,dt} = [t_0, t_1]$ where $t_0 = 0$ and $t_1 = t_0 + dt$



Algorithm

Input :

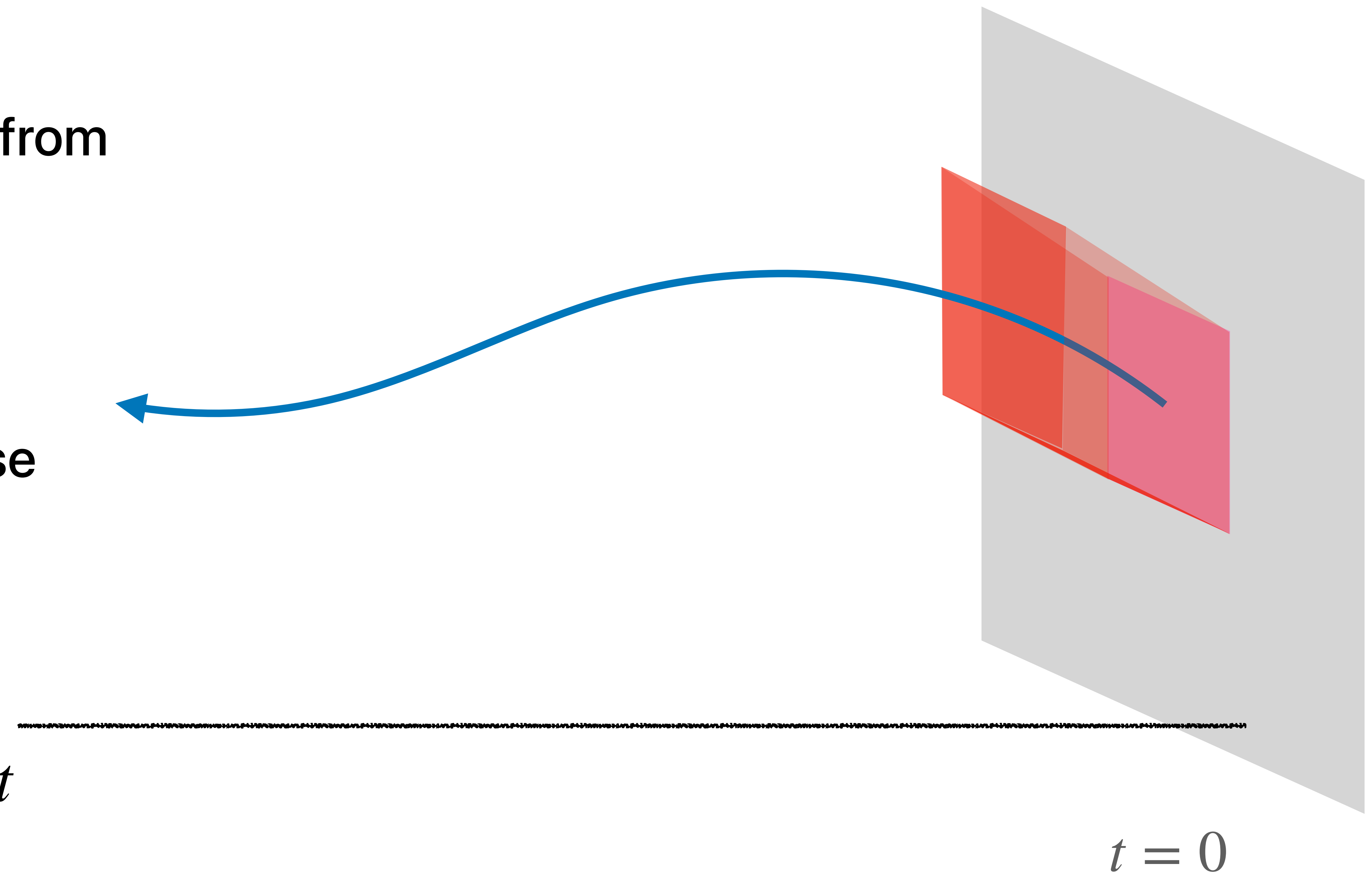
- $H(x,t) = F(x;p(t))$ a parameter homotopy from $t=0$ to $t=1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$
- $dt \in (0,1)$: the initial step size
- $T_{t_0,dt} = [t_0, t_1]$ where $t_0 = 0$ and $t_1 = t_0 + dt$
- $\lambda > 1$: a scaling constant



Algorithm

Input :

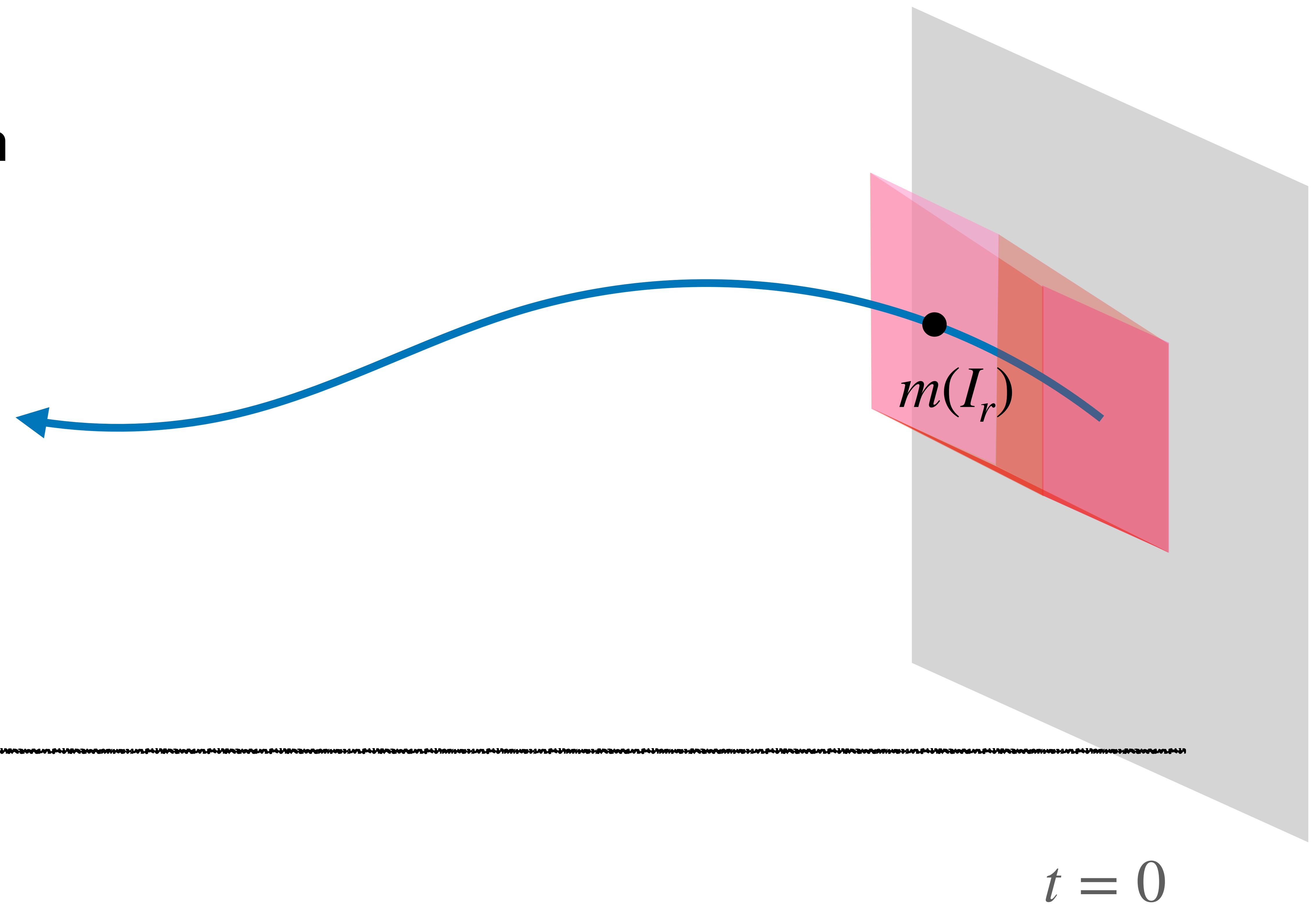
- $H(x,t) = F(x; p(t))$ a parameter homotopy from $t = 0$ to $t = 1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$
- $dt \in (0,1)$: the initial step size
- $T_{t_0,dt} = [t_0, t_1]$ where $t_0 = 0$ and $t_1 = t_0 + dt$
- $\lambda > 1$: a scaling constant



Algorithm

Input :

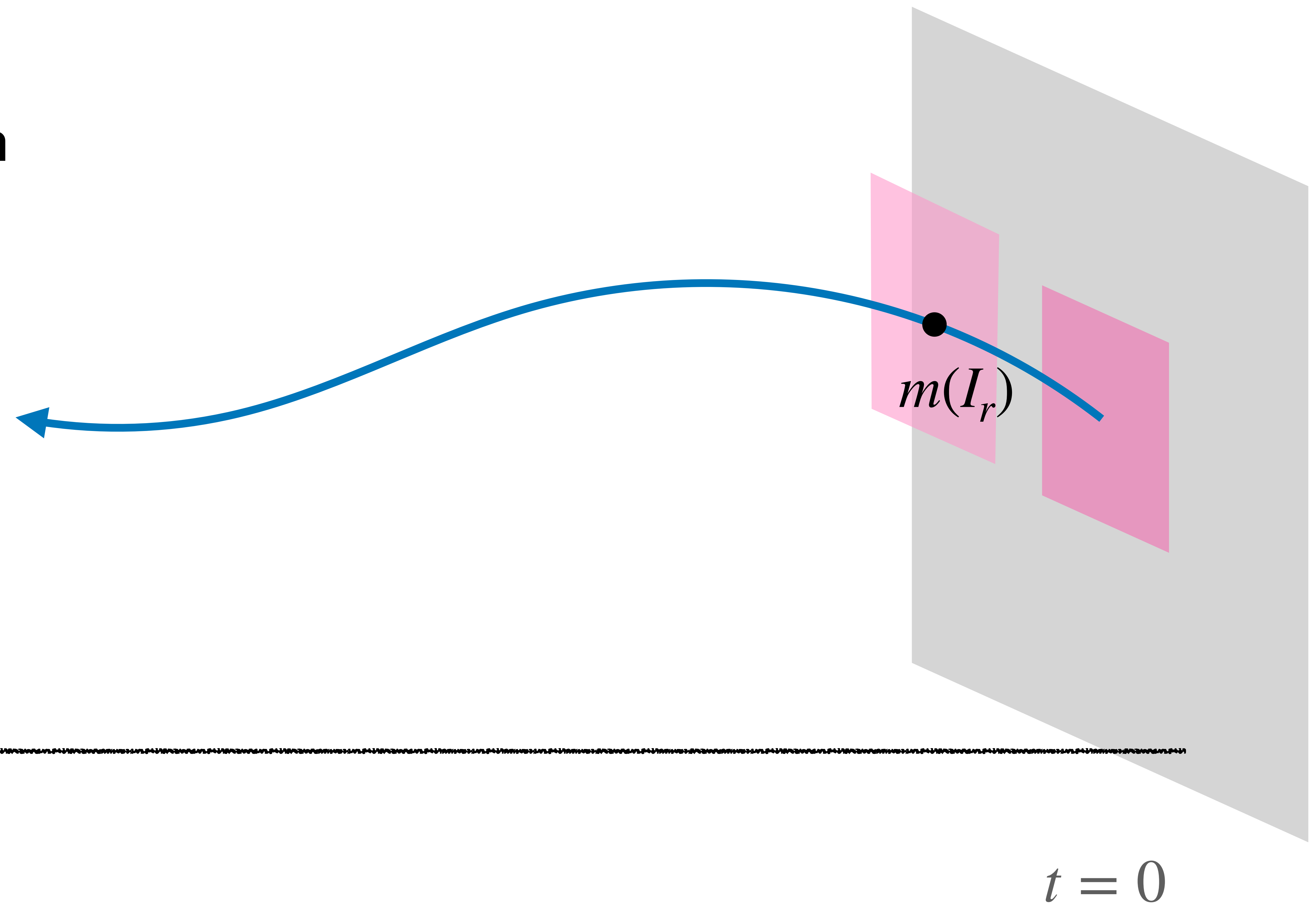
- $H(x,t) = F(x;p(t))$ a parameter homotopy from $t = 0$ to $t = 1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$
- $dt \in (0,1)$: the initial step size
- $T_{t_0,dt} = [t_0, t_1]$ where $t_0 = 0$ and $t_1 = t_0 + dt$
- $\lambda > 1$: a scaling constant



Algorithm

Input :

- $H(x,t) = F(x;p(t))$ a parameter homotopy from $t = 0$ to $t = 1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$
- $dt \in (0,1)$: the initial step size
- $T_{t_0,dt} = [t_0, t_1]$ where $t_0 = 0$ and $t_1 = t_0 + dt$
- $\lambda > 1$: a scaling constant



Algorithm

Input :

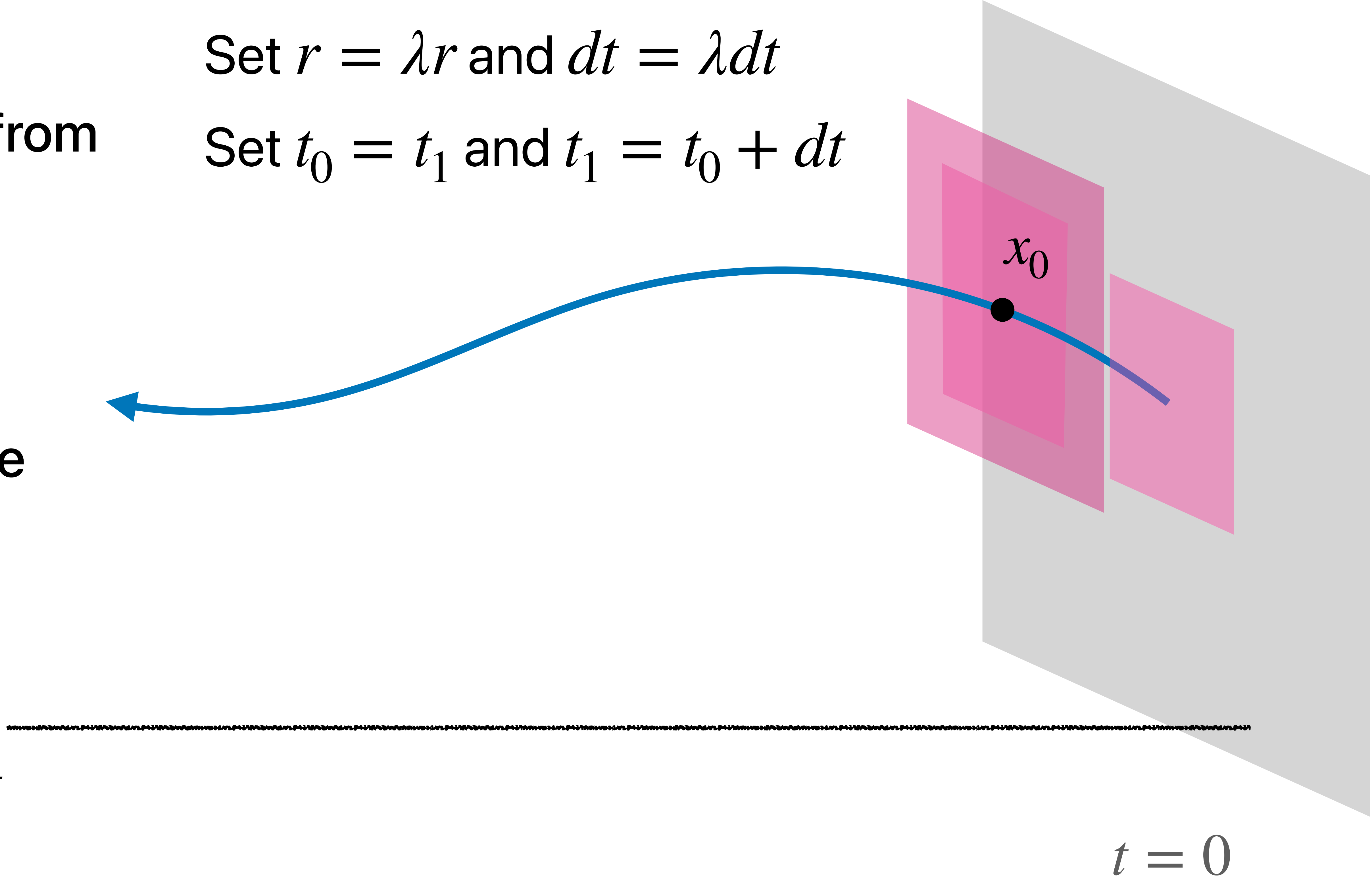
- $H(x,t) = F(x;p(t))$ a parameter homotopy from $t=0$ to $t=1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$
- $dt \in (0,1)$: the initial step size
- $T_{t_0,dt} = [t_0, t_1]$ where $t_0 = 0$ and $t_1 = t_0 + dt$
- $\lambda > 1$: a scaling constant

While $t_0 < 1$ **do**

If Krawczyk test passed **then**

Set $r = \lambda r$ and $dt = \lambda dt$

Set $t_0 = t_1$ and $t_1 = t_0 + dt$



Algorithm

Input :

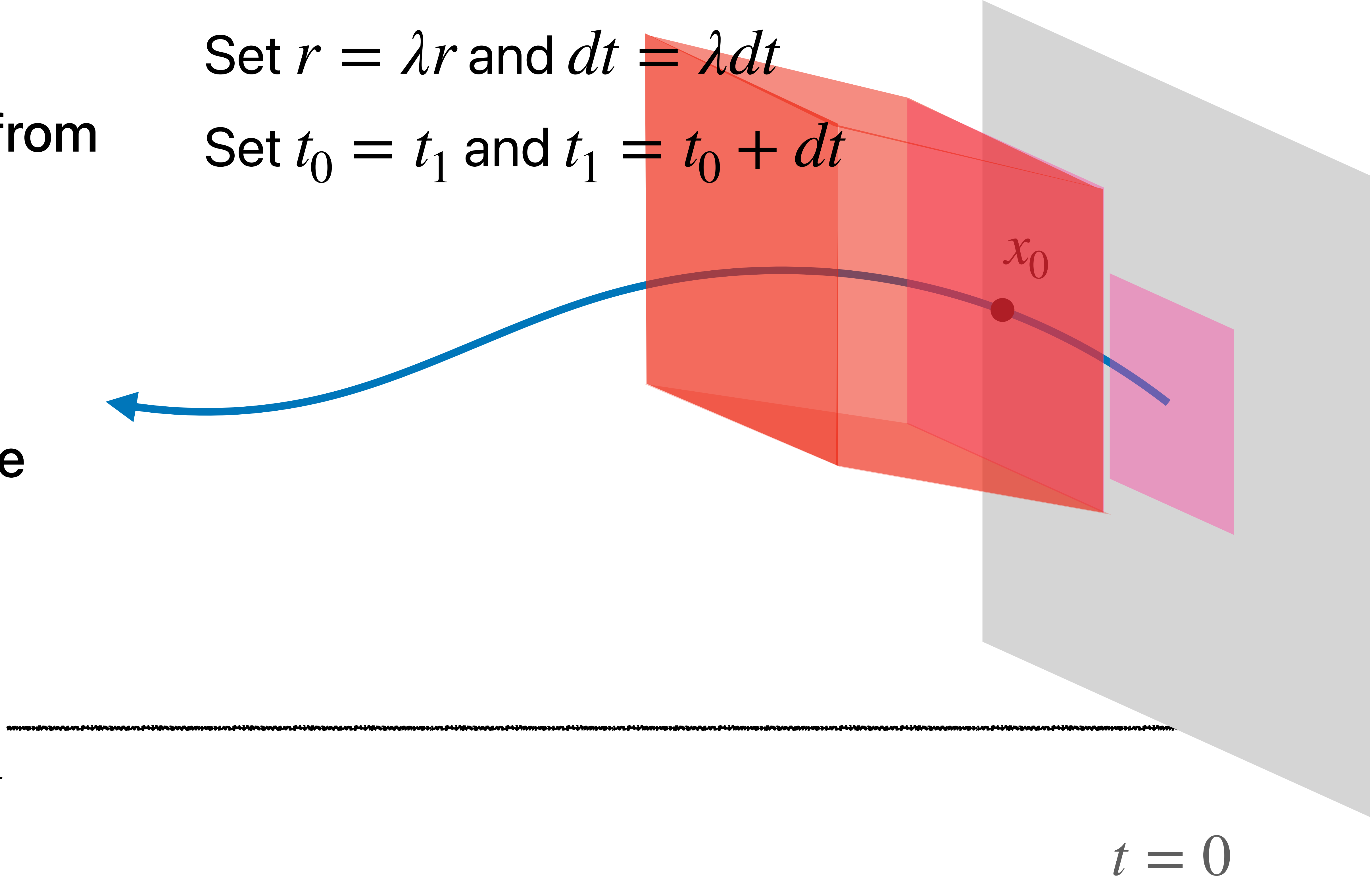
- $H(x,t) = F(x; p(t))$ a parameter homotopy from $t = 0$ to $t = 1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$
- $dt \in (0,1)$: the initial step size
- $T_{t_0,dt} = [t_0, t_1]$ where $t_0 = 0$ and $t_1 = t_0 + dt$
- $\lambda > 1$: a scaling constant

While $t_0 < 1$ **do**

If Krawczyk test passed **then**

Set $r = \lambda r$ and $dt = \lambda dt$

Set $t_0 = t_1$ and $t_1 = t_0 + dt$



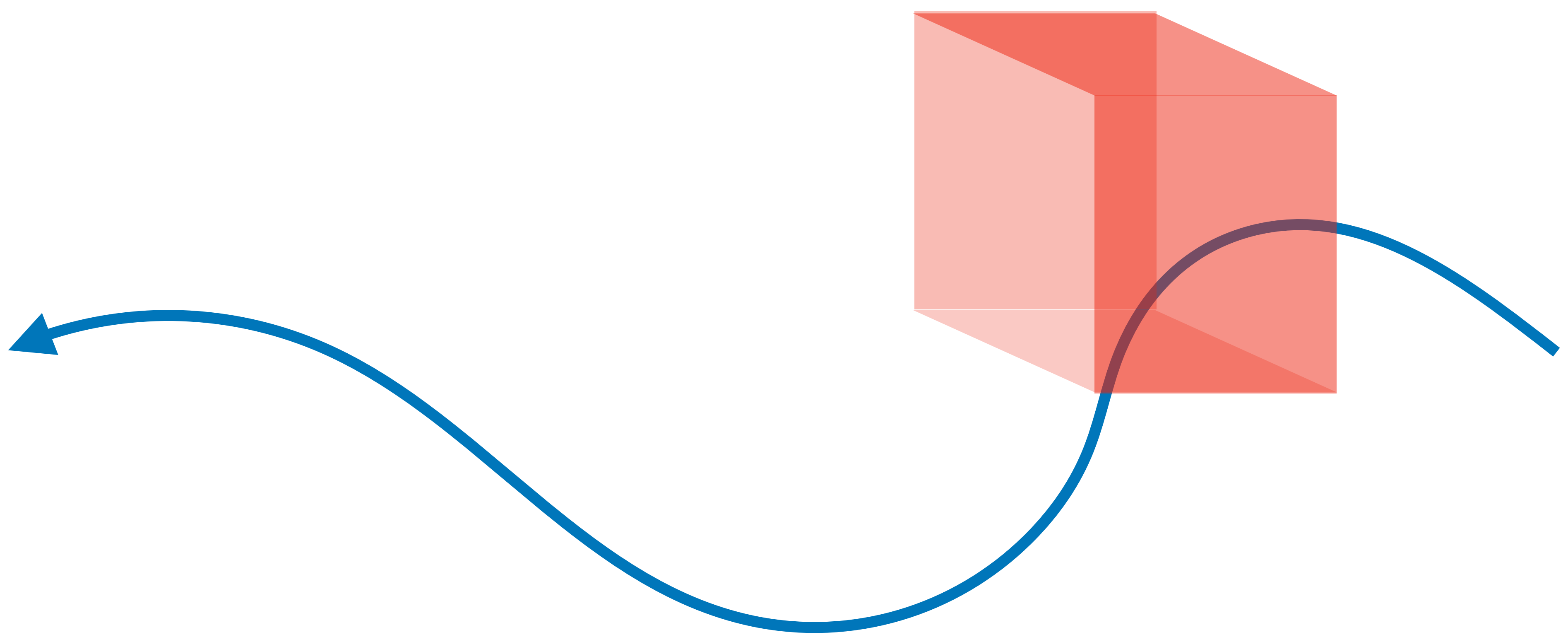
Algorithm

Input :

- $H(x, t) = F(x; p(t))$ a parameter homotopy from $t = 0$ to $t = 1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$
- $dt \in (0,1)$: the initial step size
- $T_{t_0,dt} = [t_0, t_1]$ where $t_0 = 0$ and $t_1 = t_0 + dt$
- $\lambda > 1$: a scaling constant

While $t_0 < 1$ **do**

If Krawczyk test failed **then**



Algorithm

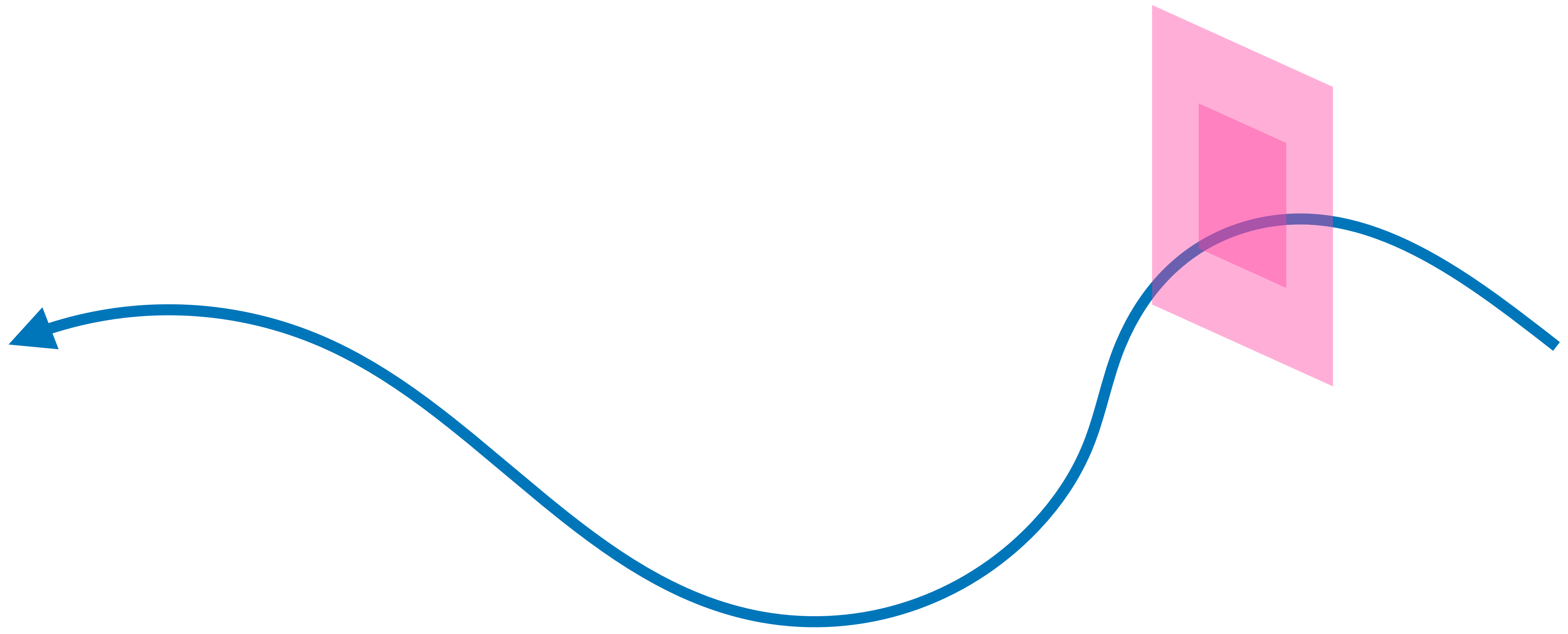
Input :

- $H(x, t) = F(x; p(t))$ a parameter homotopy from $t = 0$ to $t = 1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$
- $dt \in (0,1)$: the initial step size
- $T_{t_0,dt} = [t_0, t_1]$ where $t_0 = 0$ and $t_1 = t_0 + dt$
- $\lambda > 1$: a scaling constant

While $t_0 < 1$ **do**

If Krawczyk test failed **then**

Set $r = \dfrac{1}{\lambda} r$ and $dt = \dfrac{1}{\lambda} dt$



Algorithm

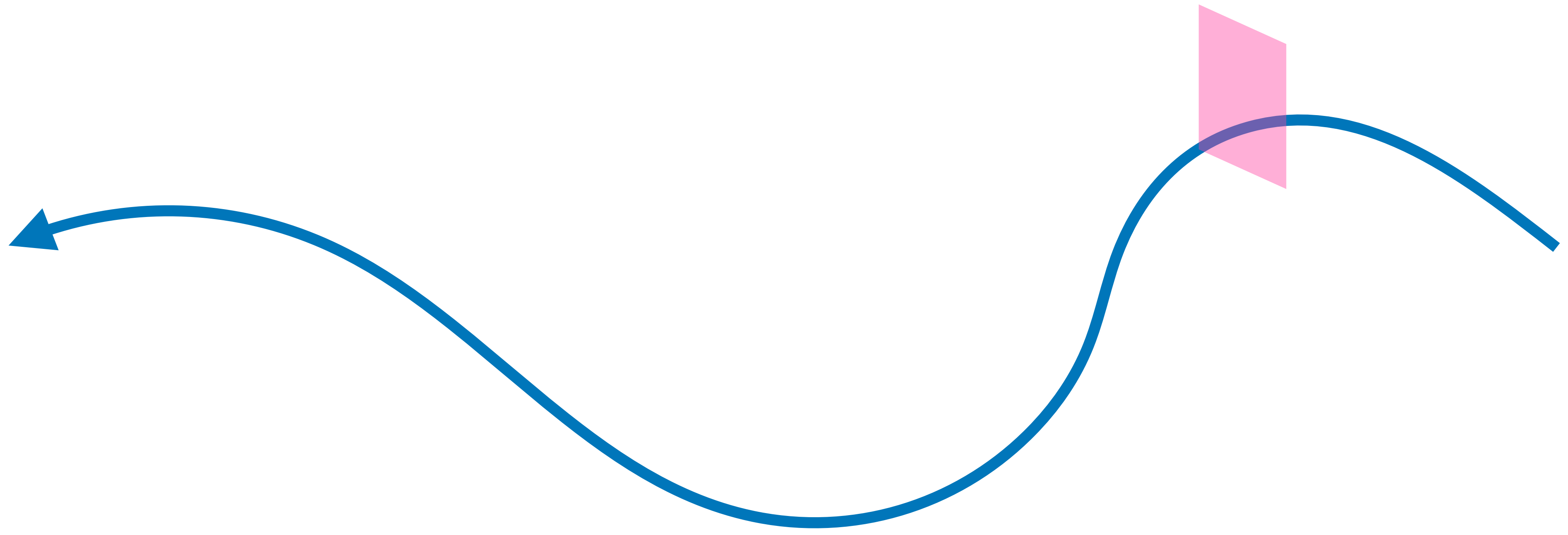
Input :

- $H(x,t) = F(x;p(t))$ a parameter homotopy from $t=0$ to $t=1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$
- $dt \in (0,1)$: the initial step size
- $T_{t_0,dt} = [t_0, t_1]$ where $t_0 = 0$ and $t_1 = t_0 + dt$
- $\lambda > 1$: a scaling constant

While $t_0 < 1$ **do**

If Krawczyk test failed **then**

Set $r = \dfrac{1}{\lambda} r$ and $dt = \dfrac{1}{\lambda} dt$



Algorithm

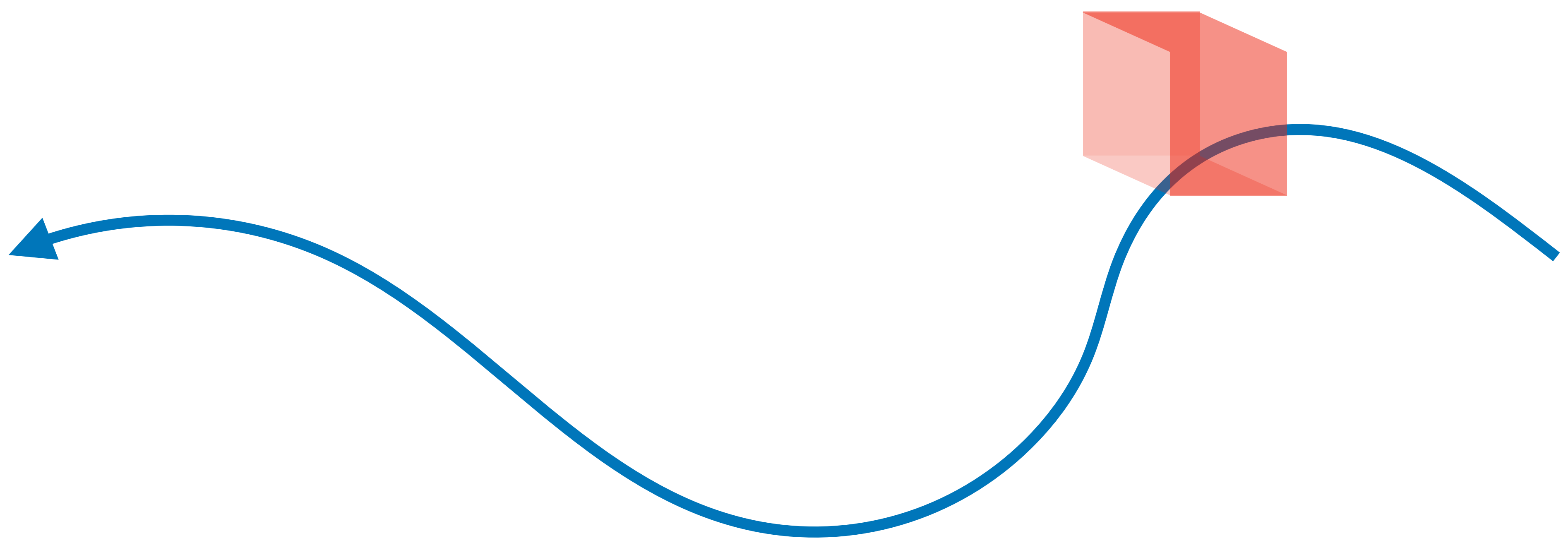
Input :

- $H(x, t) = F(x; p(t))$ a parameter homotopy from $t = 0$ to $t = 1$
- $x(t)$: a nonsingular solution path in $\mathbb{C}^n$
- I_r : a square interval box with a radius r whose midpoint $x_0 = m(I_r)$ approximates $x(0)$
- $dt \in (0,1)$: the initial step size
- $T_{t_0,dt} = [t_0, t_1]$ where $t_0 = 0$ and $t_1 = t_0 + dt$
- $\lambda > 1$: a scaling constant

While $t_0 < 1$ **do**

If Krawczyk test failed **then**

Set $r = \dfrac{1}{\lambda} r$ and $dt = \dfrac{1}{\lambda} dt$



Algorithm

Algorithm 3 Krawczyk homotopy (tilted)

Input:
- A parameter homotopy $H(x,t) = F(x; p(t)) : \mathbb{C}^n \times [0,1] \to \mathbb{C}^n$ analytic in x and linear in p,
- a point x_0 approximating $x(0)$, for some nonsingular solution path $x(t) : [0,1] \to \mathbb{C}^n$ such that $H(x(t), t) = 0$,
- a positive number $r > 0$ for the initial radius,
- a time-step size $dt \in (0,1)$, and
- a scaling constant $\lambda > 1$.

Output: A certified approximation of $x(1)$.

Set $t_0 = 0$ and $t_1 = dt$.
Run **Preconditioning**$(H(x,t), r, t_0, t_1, x_0)$ to compute $x_1, \hat{H}(x,t), I_r$ and $T_{t_0,dt}$.
Compute $Y := \partial_x \hat{H}(0, t_0)^{-1}$.
while $t_0 < 1$ **do**
 Run Krawczyk test with $K_{0,Y}(I_r, T_{t_0,dt})$.
 if Krawczyk test passed **then**
  Set $r = \lambda r$ and $dt = \lambda dt$.
  Refine x_1
  Set $x_0 = x_1, t_0 = t_1$, and $t_1 = t_0 + dt$.
  Run **Preconditioning**$(H(x,t), r, t_0, t_1, x_0)$ to compute $x_1, \hat{H}(x,t), I_r$ and $T_{t_0,dt}$.
  Compute $Y := \partial_x \hat{H}(0, t_0)^{-1}$.
 else
  Set $r = \frac{1}{\lambda} r$ and $dt = \frac{1}{\lambda} dt$.
  Run **Preconditioning**$(H(x,t), r, t_0, t_1, x_0)$ to compute $x_1, \hat{H}(x,t), I_r$ and $T_{t_0,dt}$.
 end if
end while
Find a point x_1 by refining $s(1)$ with the system $H(x,1)$.
Return x_1.

Termination

Theorem (Duff-L.). Let $H(x,t) = t\gamma G(x) + (1-t)F(x)$ be twice differentiable with an exact solution $x^\star$ to $H(x,t_0)$.

Termination

Theorem (Duff-L.). Let $H(x,t) = t\gamma G(x) + (1-t)F(x)$ be twice differentiable with an exact solution $x^\star$ to $H(x,t_0)$. Consider a fixed constant $L \geq \left\| \square\, \partial_x^2 H(I_1, [0,1]) \right\|$.

Termination

Theorem (Duff-L.). Let $H(x,t) = t\gamma G(x) + (1-t)F(x)$ be twice differentiable with an exact solution $x^\star$ to $H(x,t_0)$. Consider a fixed constant $L \geq \left\| \square\, \partial_x^2 H(I_1, [0,1]) \right\|$.
Assuming that $x(t)$ is nonsingular at any $t \in [0,1]$,

Termination

Theorem (Duff-L.). Let $H(x,t) = t\gamma G(x) + (1-t)F(x)$ be twice differentiable with an exact solution $x^\star$ to $H(x,t_0)$. Consider a fixed constant $L \geq \left\| \square \partial_x^2 H(I_1, [0,1]) \right\|$.
Assuming that $x(t)$ is nonsingular at any $t \in [0,1]$, if dt and r satisfy

$$1 - \|Y\| \cdot \frac{dt}{r} \cdot \|G(x^\star) - F(x^\star)\| > 0 \quad \text{and} \quad \sqrt{nr^2 + dt^2} < \frac{1}{\sqrt{2} \cdot \|Y\| \cdot L}$$

where $Y = \partial_x H(x^\star, t_0)^{-1}$, then the Krawczyk method succeeds.

Termination

Theorem (Duff-L.). Let $H(x,t) = t\gamma G(x) + (1-t)F(x)$ be twice differentiable with an exact solution $x^\star$ to $H(x,t_0)$. Consider a fixed constant $L \geq \left\| \square\, \partial_x^2 H(I_1, [0,1]) \right\|$.
Assuming that $x(t)$ is nonsingular at any $t \in [0,1]$, if dt and r satisfy

$$1 - \|Y\| \cdot \frac{dt}{r} \cdot \|G(x^\star) - F(x^\star)\| > 0 \quad \text{and} \quad \sqrt{nr^2 + dt^2} < \frac{1}{\sqrt{2} \cdot \|Y\| \cdot L}$$

where $Y = \partial_x H(x^\star, t_0)^{-1}$, then the Krawczyk method succeeds.

Termination

Theorem (Duff-L.). Let $H(x,t) = t\gamma G(x) + (1-t)F(x)$ be twice differentiable with an exact solution $x^\star$ to $H(x,t_0)$. Consider a fixed constant $L \geq \left\| \square\, \partial_x^2 H(I_1, [0,1]) \right\|$.

Assuming that $x(t)$ is nonsingular at any $t \in [0,1]$, if dt and r satisfy

$$1 - \|Y\| \cdot \frac{dt}{r} \cdot \|G(x^\star) - F(x^\star)\| > 0 \quad \text{and} \quad \sqrt{nr^2 + dt^2} < \frac{1}{\sqrt{2} \cdot \|Y\| \cdot L}$$

where $Y = \partial_x H(x^\star, t_0)^{-1}$, then the Krawczyk method succeeds.

Corollary (Duff-L.). Assuming that $x(t)$ is nonsingular at any $t \in [0,1]$, the tracking terminates in finitely many steps for the homotopy $H(x,t)$.

Implementation detail

Implementation in Macaulay2

Not the optimal version yet due to incomplete implementation of interval arithmetic in M2 (a proof-of-concept implementation)

A scaling constant $\lambda = 3$ is used

Experiments

- Comparison with the known method

  The certified homotopy **(Beltran-Leykin 2012 2013)** is based on alpha theory. Compare the number of steps for each solution path for benchmark problems. The initial radius and step-size, $r = 0.4$ and $dt = 0.1$ were used

Experiments

- Comparison with the known method

  The certified homotopy **(Beltran-Leykin 2012 2013)** is based on alpha theory. Compare the number of steps for each solution path for benchmark problems. The initial radius and step-size, $r = 0.4$ and $dt = 0.1$ were used

System	# roots	Average	Max	Min	Average(BL 2012)
Random (2,2,2)	8	317.75	509	203	198.5
Random (2,2,2,2)	16	563.25	1257	211	813.812
Random (2,2,2,2,2)	32	675.3125	5119	209	1542.5
Random (2,2,2,2,2,2)	64	1166.421875	5415	267	2211.58

- In many examples, Krawczyk homotopy requires less number of steps in average than **BL 2012**

Experiments

- Performance according to changes in step size and radius

  Solved the random (2,2,2) problem by changing the step size dt and the radius r, and measured the iterations.

Experiments

- Performance according to changes in step size and radius

Solved the random (2,2,2) problem by changing the step size dt and the radius r, and measured the iterations.

	Average	Max	Min
$(dt, r) = (.2, .4)$	134.75	181	83
$(.4, .4)$	96.5	170	44
$(.6, .4)$	98.625	166	32
$(.8, .4)$	96.75	194	24
$(.9, .4)$	100.5	216	23

	Average	Max	Min
$(.02, .04)$	135.625	189	85
$(.04, .04)$	99.5	175	48
$(.06, .04)$	98.875	171	34
$(.08, .04)$	99.5	195	28
$(.09, .04)$	101.375	215	24

Experiments

- Performance according to changes in step size and radius

Solved the random (2,2,2) problem by changing the step size dt and the radius r, and measured the iterations.

	Average	Max	Min
$(dt, r) = (.2,.4)$	134.75	181	83
$(.4,.4)$	96.5	170	44
$(.6,.4)$	98.625	166	32
$(.8,.4)$	96.75	194	24
$(.9,.4)$	100.5	216	23

	Average	Max	Min
$(.02,.04)$	135.625	189	85
$(.04,.04)$	99.5	175	48
$(.06,.04)$	98.875	171	34
$(.08,.04)$	99.5	195	28
$(.09,.04)$	101.375	215	24

Experiments

- Performance according to changes in step size and radius

Solved the random (2,2,2) problem by changing the step size dt and the radius r, and measured the iterations.

	Average	Max	Min
$(dt, r) = (.2, .4)$	134.75	181	83
$(.4, .4)$	96.5	170	44
$(.6, .4)$	98.625	166	32
$(.8, .4)$	96.75	194	24
$(.9, .4)$	100.5	216	23

	Average	Max	Min
$(.02, .04)$	135.625	189	85
$(.04, .04)$	99.5	175	48
$(.06, .04)$	98.875	171	34
$(.08, .04)$	99.5	195	28
$(.09, .04)$	101.375	215	24

- The performance of the algorithm depends on the ratio between dt and r (rather than their specific values)

Summary

- Certified homotopy tracking guarantees the correctness of homotopy tracking without path jumping or crossing
- Certified tracking assisted by interval arithmetic may be more competitive to alpha theory-based known methods
- Robust implementation to deal with larger size examples is expected

Thanks For Your Attention!

(https://arxiv.org/abs/2402.07053)