Nonlinear differential algorithm to compute all the zeros of a generic polynomial

Titas Geryba

The Background

Def. A polynomial is a mathematical expression involving a sum of powers in one or more variables multiplied by coefficients.

$$P_N(z) = c_N z^N + c_{N-1} z^{N-1} + c_{N-2} z^{N-2} + \ldots + c_1 z + c_0$$

We can factor a polynomial and rewrite it in the form:

$$P_N(z) = c_N (z - x_1)(z - x_2) \ldots (z - x_{N-1})(z - x_N)$$

The Background

Def. A polynomial is a mathematical expression involving a sum of powers in one or more variables multiplied by coefficients.

$$P_N(z) = c_N z^N + c_{N-1} z^{N-1} + c_{N-2} z^{N-2} + \ldots + c_1 z + c_0$$

We can factor a polynomial and rewrite it in the form:

$$P_N(z) = c_N (z - x_1)(z - x_2) \ldots (z - x_{N-1})(z - x_N)$$

$$P_N(z) = 0$$

The Algorithm

We will consider monic polynomials as follows:

$$P_N\left(z;\vec{c},\underline{x}\right) = z^N + \sum_{m=1}^{N}\left(c_m\, z^{N-m}\right) = \prod_{n=1}^{N}\left(z - x_n\right)$$

The Algorithm

We will consider monic polynomials as follows:

$$P_N\left(z; \vec{c}, \underline{x}\right) = z^N + \sum_{m=1}^{N} \left(c_m\, z^{N-m}\right) = \prod_{n=1}^{N} \left(z - x_n\right)$$

We then introduce that coefficients and zeros can change over a variable t :

$$p_N\left(z; \vec{\gamma}\,(t)\,, \underline{y}\,(t)\right) = z^N + \sum_{m=1}^{N} \left[\gamma_m\,(t)\;\, z^{N-m}\right] = \prod_{n=1}^{N} \left[z - y_n\,(t)\right]$$

The Algorithm

$$z^N + \sum_{m=1}^{N} \left[\gamma_m(t)\ z^{N-m}\right] = \prod_{n=1}^{N} \left[z - y_n(t)\right]$$

The Algorithm

$$z^N + \sum_{m=1}^{N} \left[\gamma_m(t)\ z^{N-m}\right] = \prod_{n=1}^{N} \left[z - y_n(t)\right]$$

$$\sum_{m=1}^{N} \left[\dot{y}_m(t) \prod_{l=1, l\neq m}^{N} [z - y_l(t)]\right] = -\sum_{m=1}^{N} [\dot{\gamma}_m(t)[z]^{N-m}]$$

The Algorithm

$$z^N + \sum_{m=1}^{N} \left[\gamma_m(t)\ z^{N-m}\right] = \prod_{n=1}^{N} \left[z - y_n(t)\right]$$

$$\sum_{m=1}^{N} \left[\dot{y}_m(t) \prod_{l=1, l\neq m}^{N} [z - y_l(t)]\right] = -\sum_{m=1}^{N} [\dot{\gamma}_m(t)[z]^{N-m}]$$

$$Let\ z = y_n(t)$$

The Algorithm

$$z^N + \sum_{m=1}^{N} \left[\gamma_m(t)\; z^{N-m}\right] = \prod_{n=1}^{N} \left[z - y_n(t)\right]$$

$$\sum_{m=1}^{N} \left[\dot{y}_m(t) \prod_{l=1, l\neq m}^{N} [z - y_l(t)]\right] = -\sum_{m=1}^{N} [\dot{\gamma}_m(t)[z]^{N-m}]$$

$$Let\ z = y_n(t)$$

$$\dot{y}_n(t)\left[\prod_{l=1, l\neq m}^{N} [y_n(t) - y_l(t)]\right] = -\sum_{m=1}^{N} [\dot{\gamma}_m(t)[y_n(t)]^{N-m}]$$

The Algorithm

$$z^N + \sum_{m=1}^{N} \left[\gamma_m(t)\; z^{N-m}\right] = \prod_{n=1}^{N} \left[z - y_n(t)\right]$$

$$\sum_{m=1}^{N} \left[\dot{y}_m(t) \prod_{l=1, l\neq m}^{N} [z - y_l(t)]\right] = -\sum_{m=1}^{N} [\dot{\gamma}_m(t)[z]^{N-m}]$$

$$Let\; z = y_n(t)$$

$$\dot{y}_n(t)\left[\prod_{l=1, l\neq m}^{N} [y_n(t) - y_l(t)]\right] = -\sum_{m=1}^{N} [\dot{\gamma}_m(t)[y_n(t)]^{N-m}]$$

$$\dot{y}_n(t) = -\left\{\prod_{l=1, l\neq m}^{N} [y_n(t) - y_l(t)]^{-1}\right\} \sum_{m=1}^{N} [\dot{\gamma}_m(t)[y_n(t)]^{N-m}]$$

The Algorithm

$$\dot{y}_n(t) = -\left\{ \prod_{\ell=1,\ \ell \neq n}^{N} \left[y_n(t) - y_\ell(t) \right]^{-1} \right\} \sum_{m=1}^{N} \left\{ \dot{\gamma}_m(t) \left[y_n(t) \right]^{N-m} \right\}$$

The Algorithm

$$\dot{y}_n(t) = -\left\{ \prod_{\ell=1,\ \ell\neq n}^{N} [y_n(t) - y_\ell(t)]^{-1} \right\} \sum_{m=1}^{N} \left\{ \dot{\gamma}_m(t)\ [y_n(t)]^{N-m} \right\}$$

$$\gamma_m(t) = \gamma_m(0) + \left[\frac{f(t) - f(0)}{f(T) - f(0)} \right] [c_m - \gamma_m(0)]$$

The Algorithm

$$\dot{y}_n(t) = -\left\{ \prod_{\ell=1,\ \ell \neq n}^{N} [y_n(t) - y_\ell(t)]^{-1} \right\} \sum_{m=1}^{N} \left\{ \dot{\gamma}_m(t)\ [y_n(t)]^{N-m} \right\}$$

$$\gamma_m(t) = \gamma_m(0) + \left[\frac{f(t) - f(0)}{f(T) - f(0)} \right] [c_m - \gamma_m(0)]$$

$$\gamma_m(T) = c_m$$

$$\gamma_m(0) = \gamma_m(0)$$

The Algorithm

$$\dot{y}_n(t) = -\left\{ \prod_{\ell=1,\ \ell\neq n}^{N} \left[y_n(t) - y_\ell(t) \right]^{-1} \right\} \sum_{m=1}^{N} \left\{ \dot{\gamma}_m(t) \left[y_n(t) \right]^{N-m} \right\}$$

$$\gamma_m(t) = \gamma_m(0) + \left[\frac{f(t) - f(0)}{f(T) - f(0)} \right] \left[c_m - \gamma_m(0) \right]$$

$$\gamma_m(T) = c_m$$

$$\gamma_m(0) = \gamma_m(0)$$

$$\dot{\gamma}_m(t) = g(t) \left[c_m - \gamma_m(0) \right]$$

$$g(t) = \frac{\dot{f}(t)}{f(T) - f(0)} \text{ implying } \int_0^T dt\, g(t) = 1$$

The Algorithm

$$\dot{y}_n(t) = -\left\{\prod_{\ell=1,\ \ell\neq n}^{N} [y_n(t) - y_\ell(t)]^{-1}\right\} \sum_{m=1}^{N} \left\{\dot{\gamma}_m(t)\ [y_n(t)]^{N-m}\right\}$$

$$\gamma_m(t) = \gamma_m(0) + \left[\frac{f(t) - f(0)}{f(T) - f(0)}\right] [c_m - \gamma_m(0)]$$

$$\gamma_m(T) = c_m$$

$$\gamma_m(0) = \gamma_m(0)$$

$$\dot{\gamma}_m(t) = g(t)\ [c_m - \gamma_m(0)]$$

$$g(t) = \frac{\dot{f}(t)}{f(T) - f(0)} \quad \text{implying} \int_0^T dt\, g(t) = 1$$

The Algorithm

$$\dot{y}_n(t) = -g(t) \left\{ \prod_{\ell=1,\ \ell\neq n}^{N} [y_n(t) - y_\ell(t)]^{-1} \right\} \sum_{m=1}^{N} \left\{ [c_m - \gamma_m(0)]\ [y_n(t)]^{N-m} \right\}$$

$$g(t) = \frac{\dot{f}(t)}{f(T) - f(0)} \quad \text{implying} \int_0^T dt\, g(t) = 1\,.$$

The Algorithm

$$g(t) = \frac{\dot{f}(t)}{f(T) - f(0)} \text{ implying } \int_0^T dt\, g(t) = 1$$

$$T = 1$$

$$f(t) = t \Rightarrow g(t) = \frac{1}{f(1) - f(0)} = 1$$

The Algorithm

$$g(t) = \frac{\dot{f}(t)}{f(T) - f(0)} \text{ implying } \int_0^T dt\, g(t) = 1$$

$$T = 1$$

$$f(t) = t \Rightarrow g(t) = \frac{1}{f(1) - f(0)} = 1$$

$$\int_0^T dt\, g(t) = \int_0^1 dt\, 1 = 1$$

The Algorithm

$$\dot{y}_n(t) = -g(t) \left\{ \prod_{\ell=1,\ \ell \neq n}^{N} [y_n(t) - y_\ell(t)]^{-1} \right\} \sum_{m=1}^{N} \left\{ [c_m - \gamma_m(0)] \, [y_n(t)]^{N-m} \right\}$$

$$g(t) = 1, \qquad T = 1$$

The Algorithm

$$\dot{y}_n(t) = -g(t) \left\{ \prod_{\ell=1,\ \ell\neq n}^{N} [y_n(t) - y_\ell(t)]^{-1} \right\} \sum_{m=1}^{N} \left\{ [c_m - \gamma_m(0)]\ [y_n(t)]^{N-m} \right\}$$

$$g(t) = 1, \qquad T = 1$$

$$F(t) = \begin{cases} \dot{y}_1(t) \\ \dot{y}_2(t) \\ \vdots \\ \dot{y}_N(t) \end{cases}$$

The Algorithm

1. Generate the initial guesses for the polynomial's zeros at t = 0

$$y_1(0), y_2(0), \ldots, y_N(0)$$

2. Calculate the time dependent coefficients' values

$$\gamma_m(0) = (-1)^m \sum_{n_1 > n_2 > \cdots > n_m = 1}^{N} \left[y_{n_1}(0)\; y_{n_2}(0) \cdots y_{n_m}(0) \right]$$

3. Integrate the system of equations numerically

$$\dot{y}_n(t) = -g(t) \left\{ \prod_{\ell=1,\ \ell \neq n}^{N} \left[y_n(t) - y_\ell(t) \right]^{-1} \right\} \sum_{m=1}^{N} \left\{ \left[c_m - \gamma_m(0) \right] \left[y_n(t) \right]^{N-m} \right\}$$

$$F(t) = \begin{cases} \dot{y}_1(t) \\ \dot{y}_2(t) \\ \vdots \\ \dot{y}_n(t) \end{cases}$$

$$g(t) = 1, \qquad T = 1$$

Example 1

$$\begin{aligned} &x^3 - 15x^2 - 150x + 1000 \\ &= (x-20)(x+20)(x-10) \end{aligned}$$

Example 1

$$x^3 - 15x^2 - 150x + 1000$$
$$= (x-20)(x+20)(x-10)$$

$$-100 \le Re(y_i(0)) \le 100$$
$$-100 \le Im(y_i(0)) \le 100$$
$$h = 0.01$$

Example 1

$$x^3 - 15x^2 - 150x + 1000$$
$$= (x - 20)(x + 20)(x - 10)$$

$$-100 \leq Re(y_i(0)) \leq 100$$
$$-100 \leq Im(y_i(0)) \leq 100$$

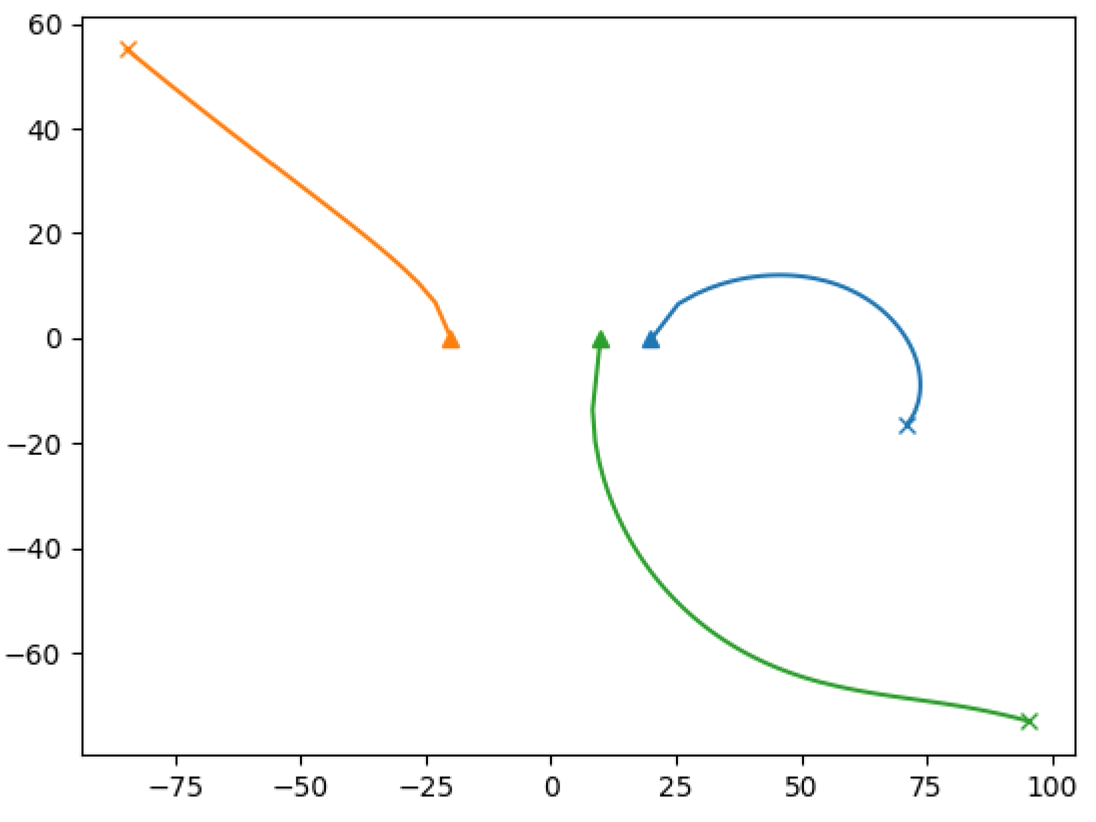
Initial Value	After Iteration
71.1406749978-16.2823174277i	
-84.7349639578+55.126702689i	
95.5650319505-73.1030055883i	

Example 1

$$x^3 - 15x^2 - 150x + 1000$$
$$= (x-20)(x+20)(x-10)$$

$$-100 \leq Re(y_i(0)) \leq 100$$
$$-100 \leq Im(y_i(0)) \leq 100$$
$$time\ step\ h = 0.01$$

Iteration 1

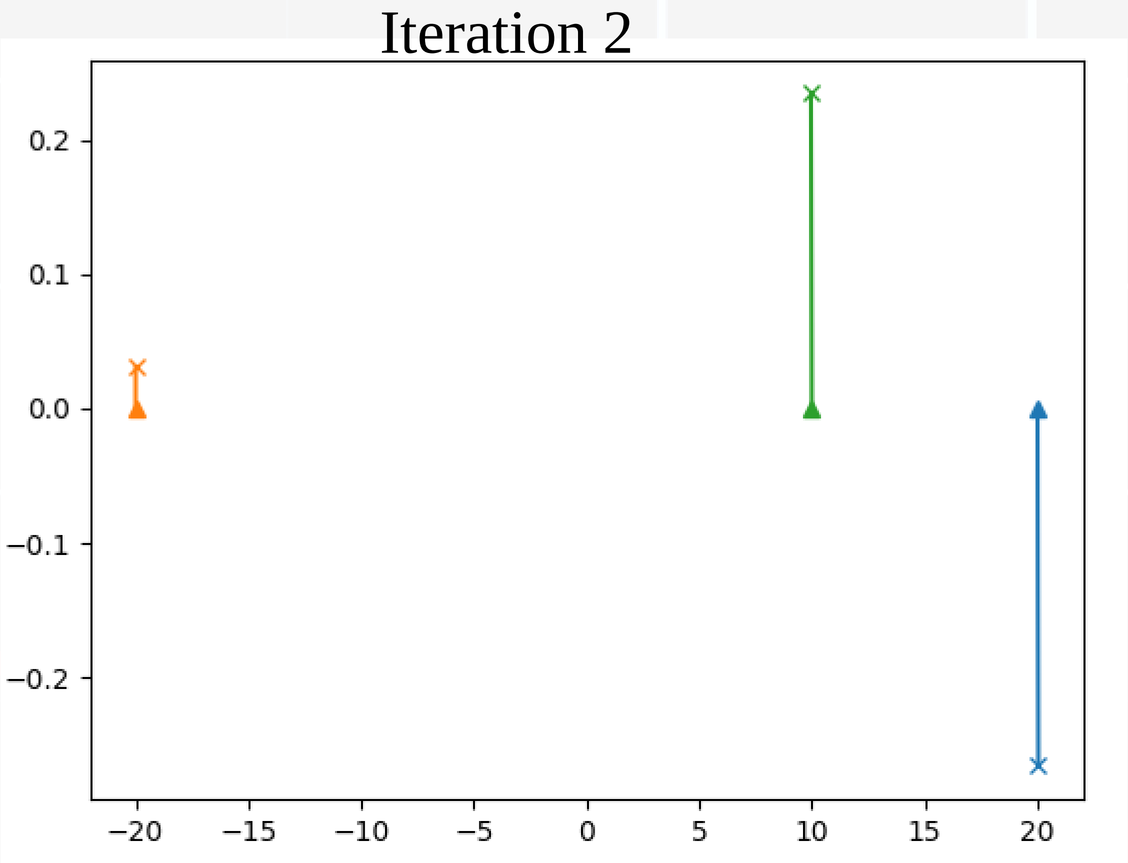


Initial Value	After Iteration
71.1406749978-16.2823174277i	20.0370960114-0.266170648667i
-84.7349639578+55.126702689i	-19.9976154713+0.0311076576057i
95.5650319505-73.1030055883i	9.9605194599+0.235062991061i

Example 1

$$x^3 - 15x^2 - 150x + 1000$$
$$= (x-20)(x+20)(x-10)$$

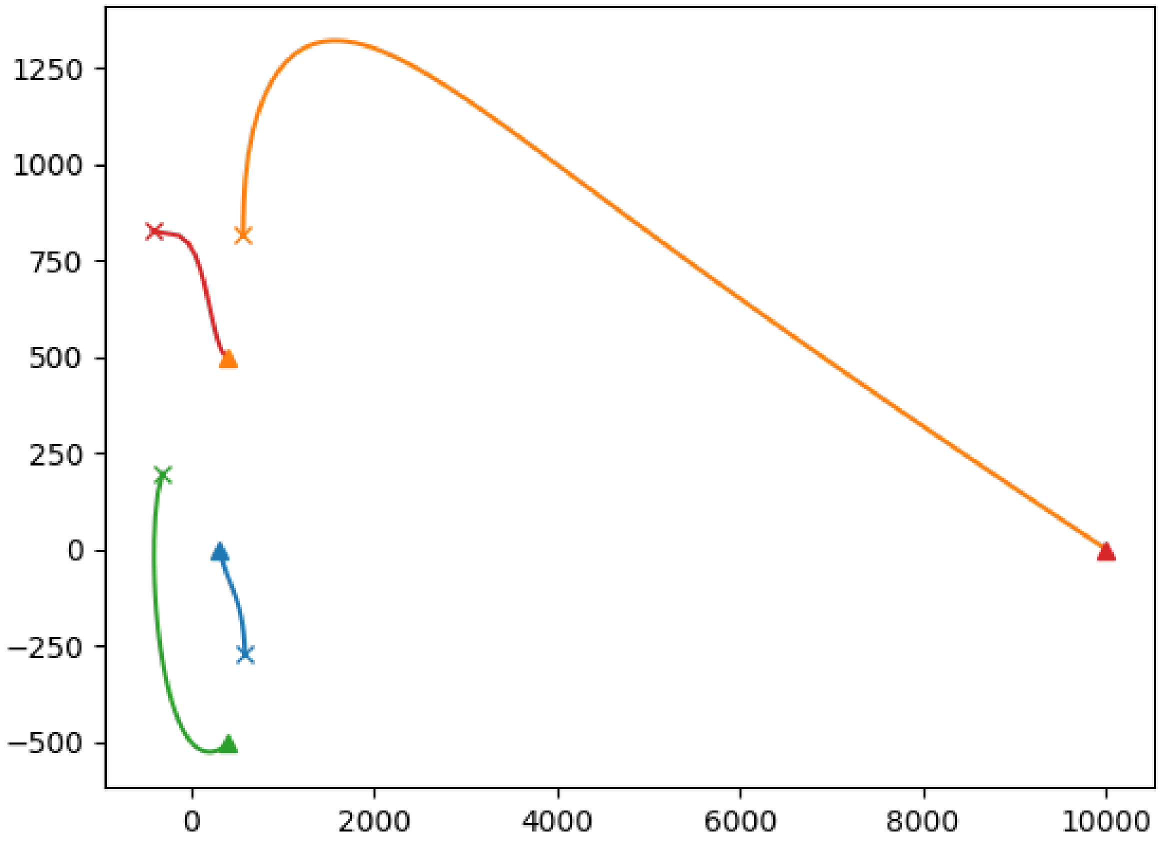
$$-100 \leq Re(y_i(0)) \leq 100$$
$$-100 \leq Im(y_i(0)) \leq 100$$
$$time\ step\ h = 0.01$$



Initial Value	After Iteration
20.0370960114-0.266170648667i	20+1.75207071074e-16i
-19.9976154713+0.0311076576057i	-20-2.21719344273e-17i
9.9605194599+0.235062991061i	10+1.95156391047e-17i

Example 2

$$(z-300)(z+400+500i)(z+400-500i)(z-9999)$$



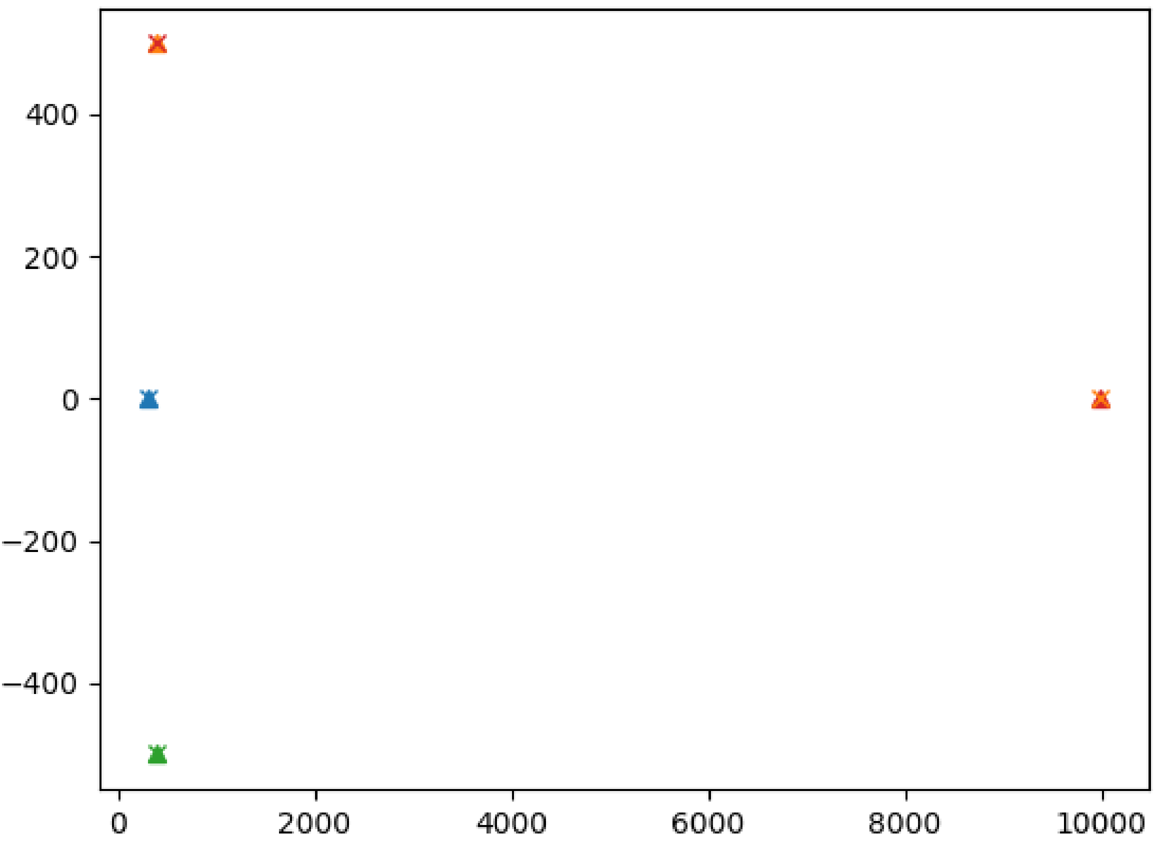
Iteration 1

Initial Value	After Iteration
585.655713746-269.955507343i	300.001015787+0.000710725589028i
564.148717961+816.963408841j	9998.99930425+0.000165669875692i
-314.12545548+197.401514585i	400.000338707-500.000649204i
-426.088792188+825.795241733i	399.999341253+499.999772808i

$$-1000 \leq Re(y_i(0)) \leq 1000$$
$$-1000 \leq Im(y_i(0)) \leq 1000$$
$$h = 0.01$$

Example 2

$$(z-300)(z+400+500i)(z+400-500i)(z-9999)$$



Iteration 2

Initial Value	After Iteration
300.001015787+0.000710725589028i	300.00000000000011-8.6230971161845894e-14j
9998.99930425+0.000165669875692i	9998.9999999999854+7.464394626691454e-15i
400.000338707-500.000649204i	400-499.99999999999983i
399.999341253+499.999772808i	400+499.99999999999994i

$$-1000 \leq Re(y_i(0)) \leq 1000$$
$$-1000 \leq Im(y_i(0)) \leq 1000$$
$$h = 0.01$$

$$z^{N} + \sum_{m=1}^{N} \left[\gamma_{m}(t)\; z^{N-m} \right] = \prod_{n=1}^{N} \left[z - y_{n}(t) \right]$$

$$z^N + \sum_{m=1}^{N} \left[\gamma_m(t)\; z^{N-m}\right] = \prod_{n=1}^{N} \left[z - y_n(t)\right]$$

1. Generate N random zeros
2. Expand the zeros to create a polynomial
3. Solve for zeros numerically
4. Compare the original zeros with the approximated zeros

$$z^N + \sum_{m=1}^{N} \left[\gamma_m(t)\ z^{N-m}\right] = \prod_{n=1}^{N} \left[z - y_n(t)\right]$$

1. Generate N random zeros
2. Expand the zeros to create a polynomial
3. Solve for zeros numerically
4. Compare the original zeros with the approximated zeros

$-10 \leq Re(y_i(0)) \leq 10$

-10 $\leq Im(y_i(0)) \leq 10$

-2 $\leq Re(x_i(0)) \leq 2$

$-2 \leq Im(x_i(0)) \leq 2$

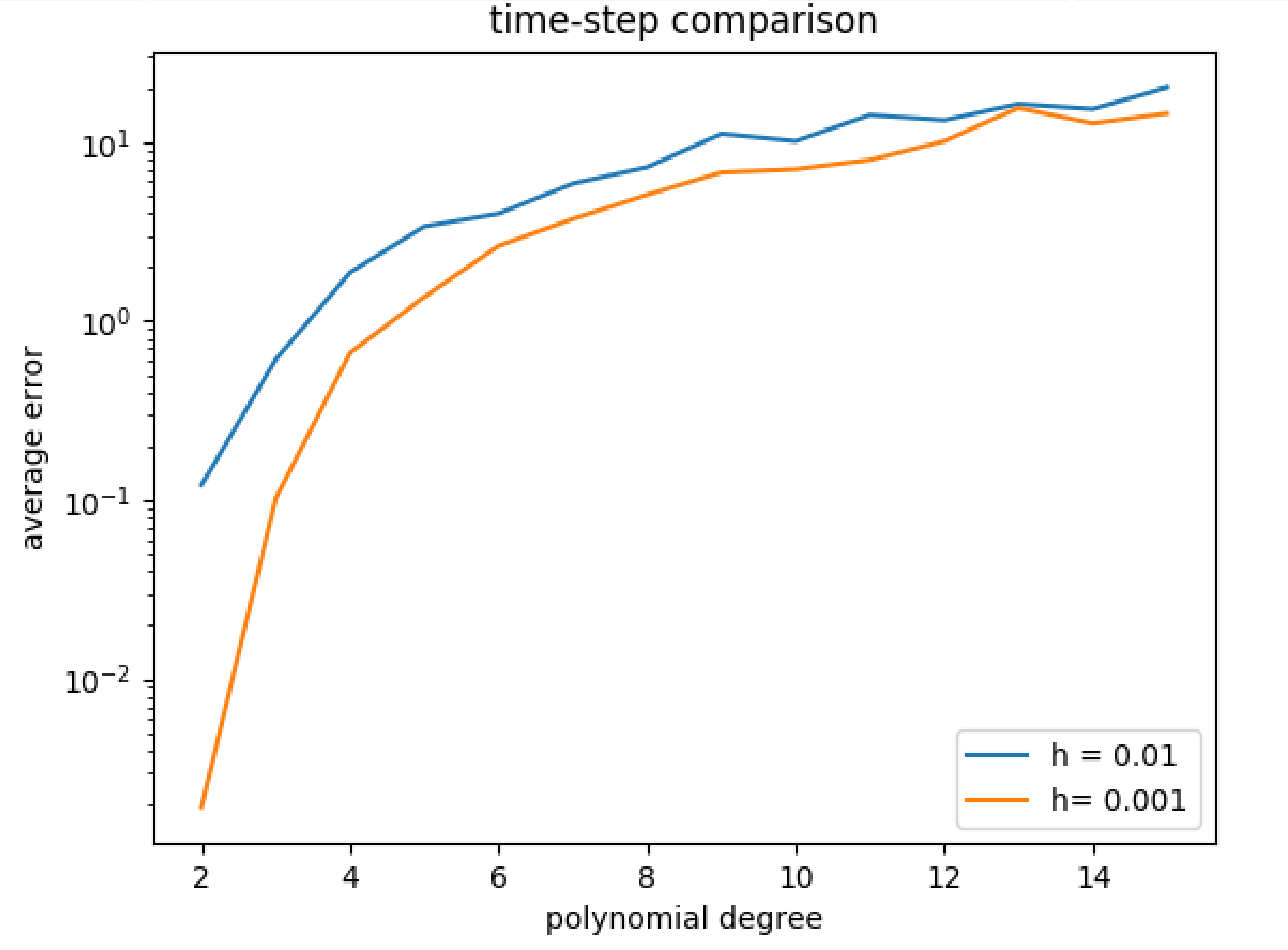
Factors I considered

1. Time step
2. Number of iterations

Time Step

1.

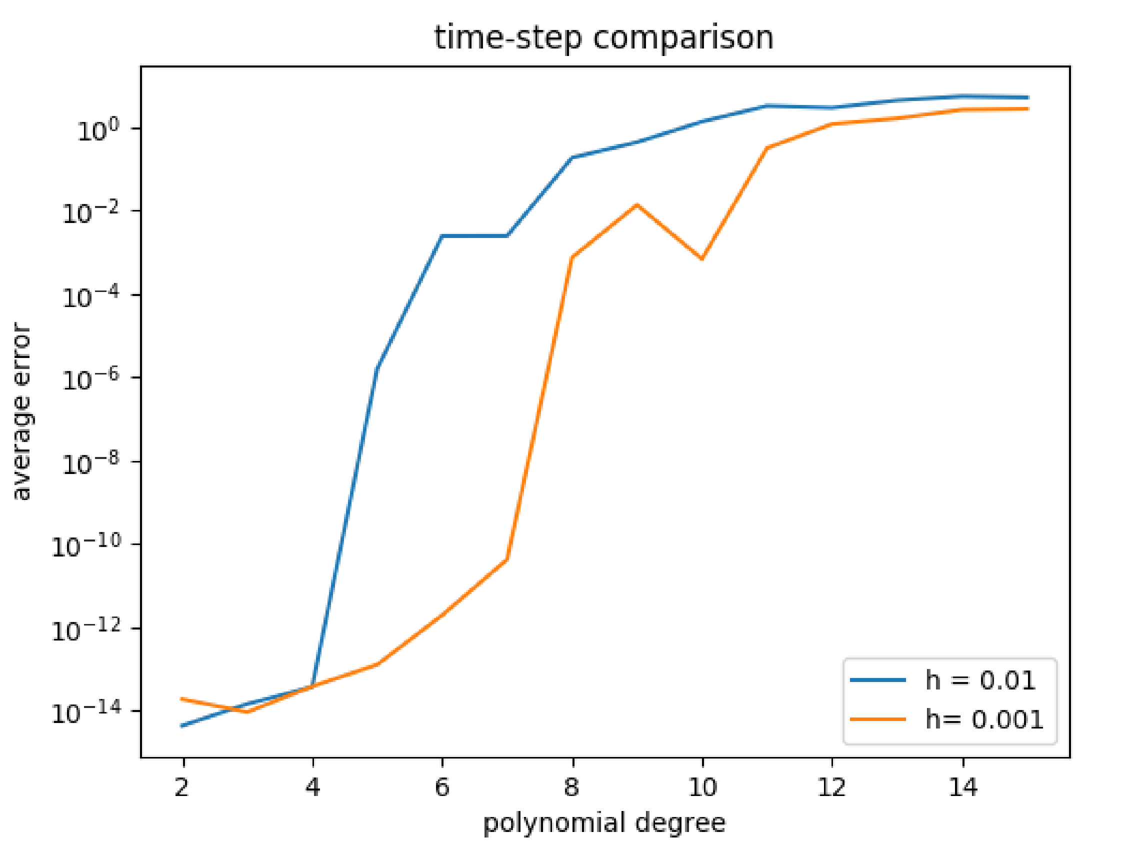
Time step



$$-2 \leq Re(y_i(0)) \leq 2$$
$$-2 \leq Im(y_i(0)) \leq 2$$
$$-10 \leq Re(x_i(0)) \leq 10$$
$$-10 \leq Im(x_i(0)) \leq 10$$

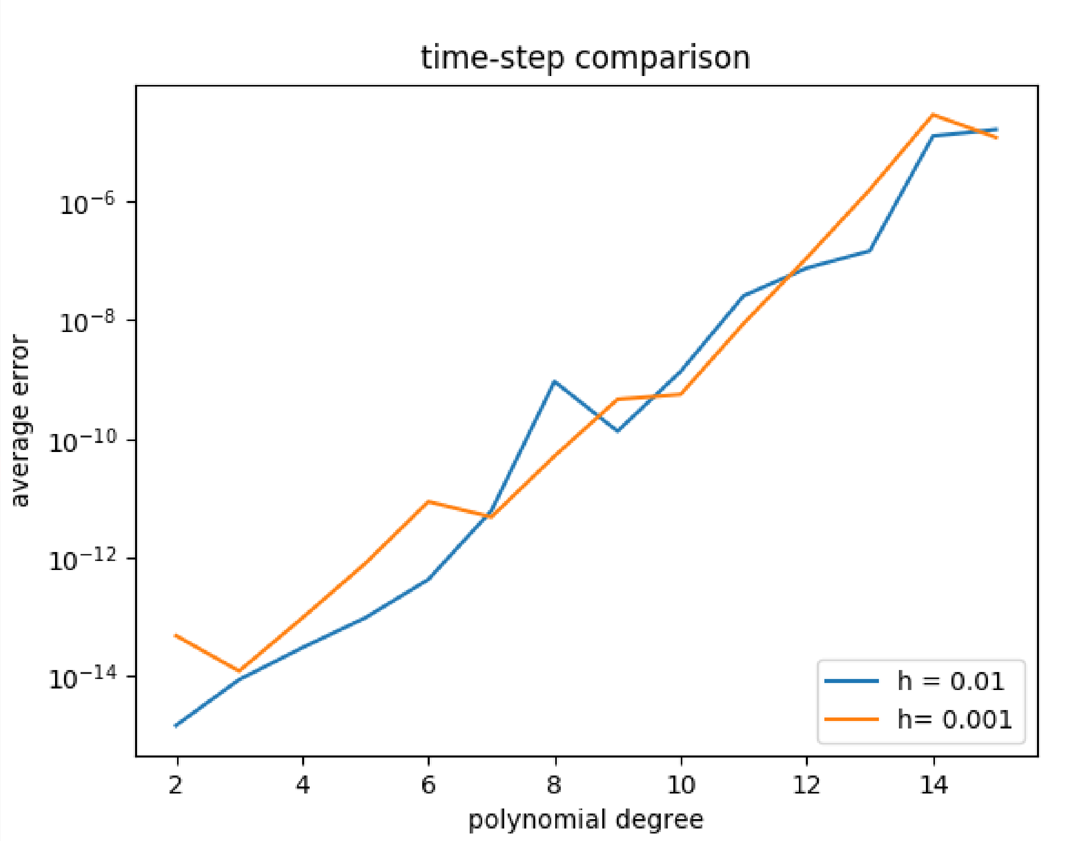
1 iteration

Time step



$$-2 \le Re(y_i(0)) \le 2$$
$$-2 \le Im(y_i(0)) \le 2$$
$$-10 \le Re(x_i(0)) \le 10$$
$$-10 \le Im(x_i(0)) \le 10$$

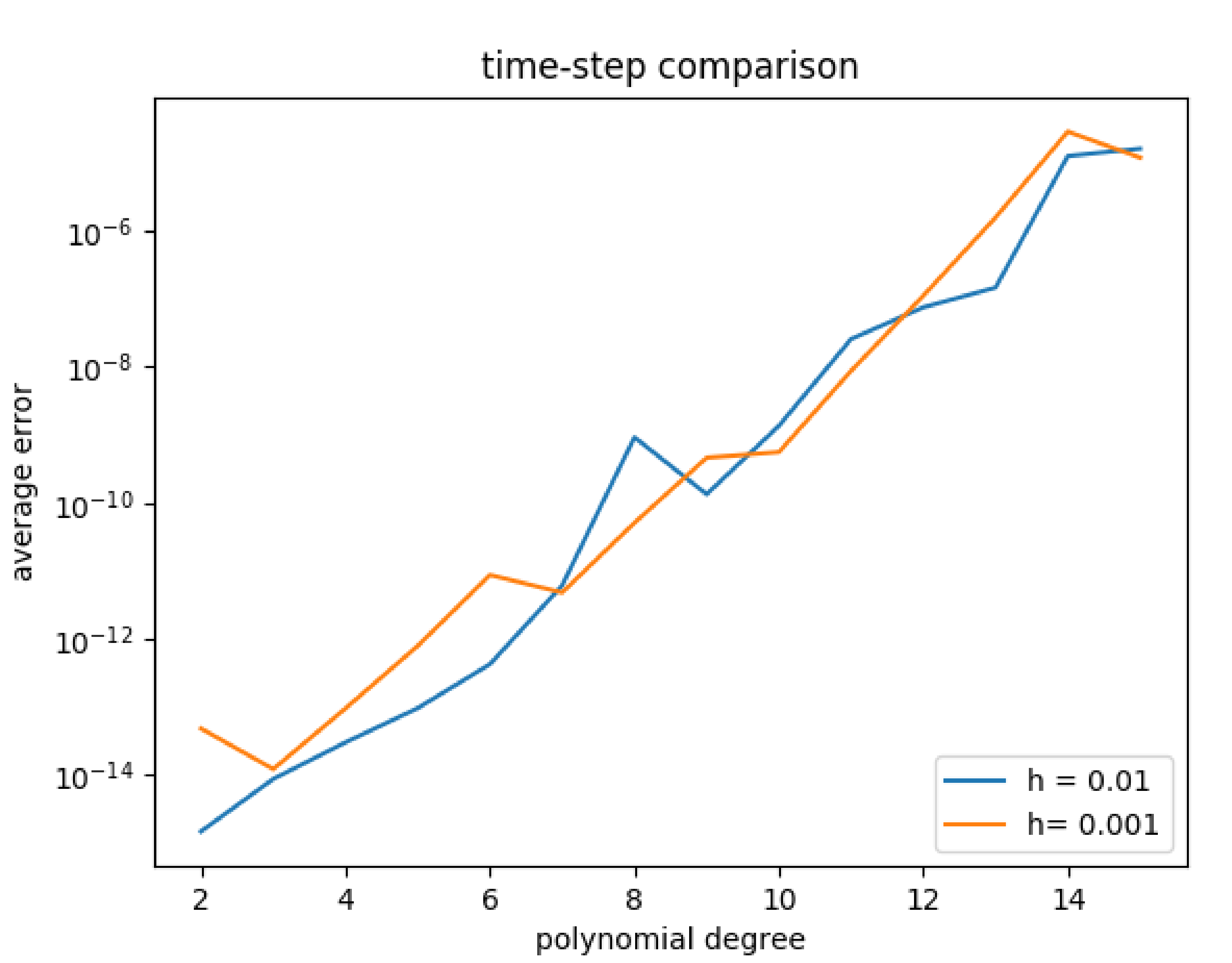
3 iterations



$$-2 \le Re(y_i(0)) \le 2$$
$$-2 \le Im(y_i(0)) \le 2$$
$$-10 \le Re(x_i(0)) \le 10$$
$$-10 \le Im(x_i(0)) \le 10$$

10 iterations

Time step



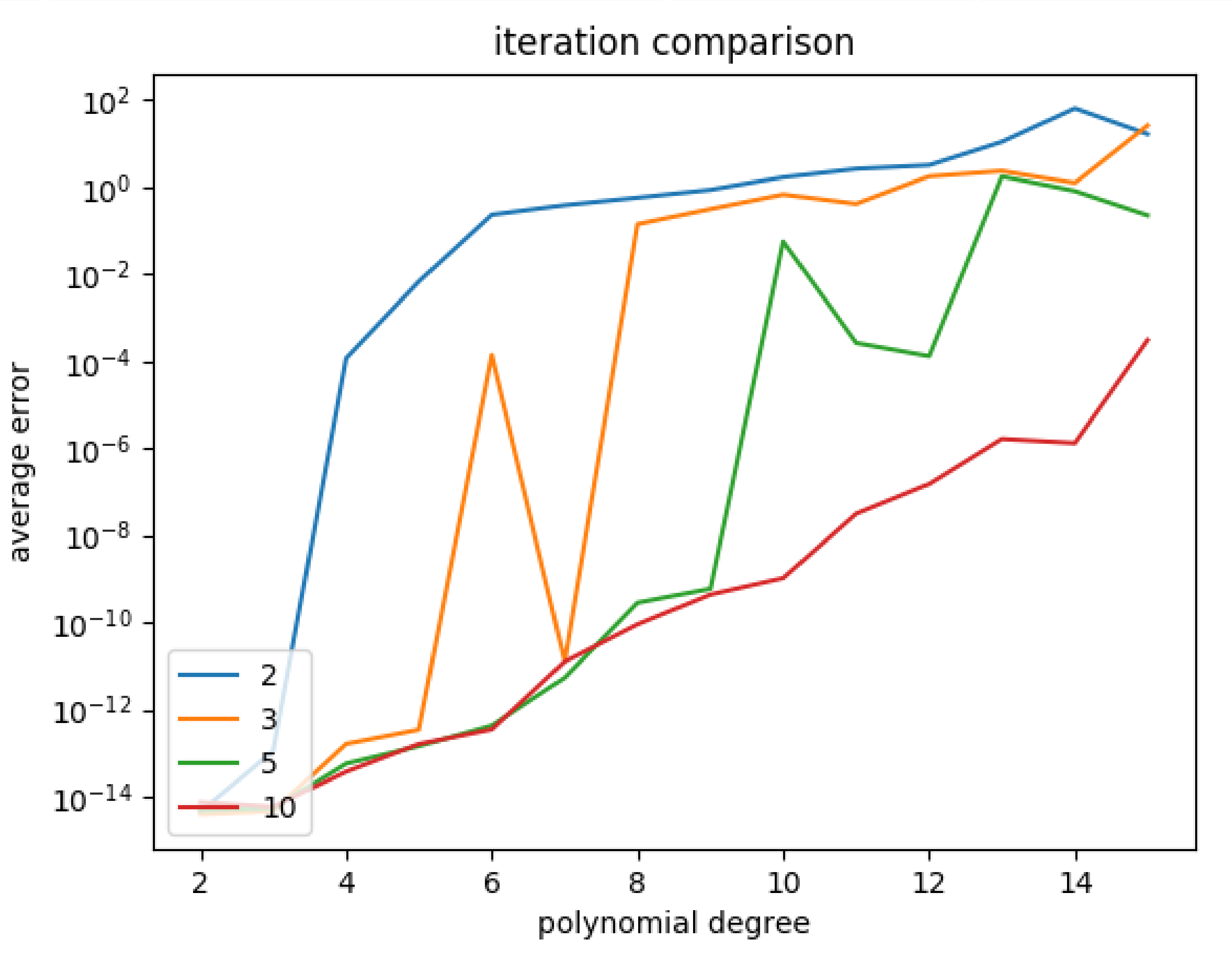
$$-2 \leq Re(y_i(0)) \leq 2$$
$$-2 \leq Im(y_i(0)) \leq 2$$
$$-10 \leq Re(x_i(0)) \leq 10$$
$$-10 \leq Im(x_i(0)) \leq 10$$

10 iterations

Iteration Number

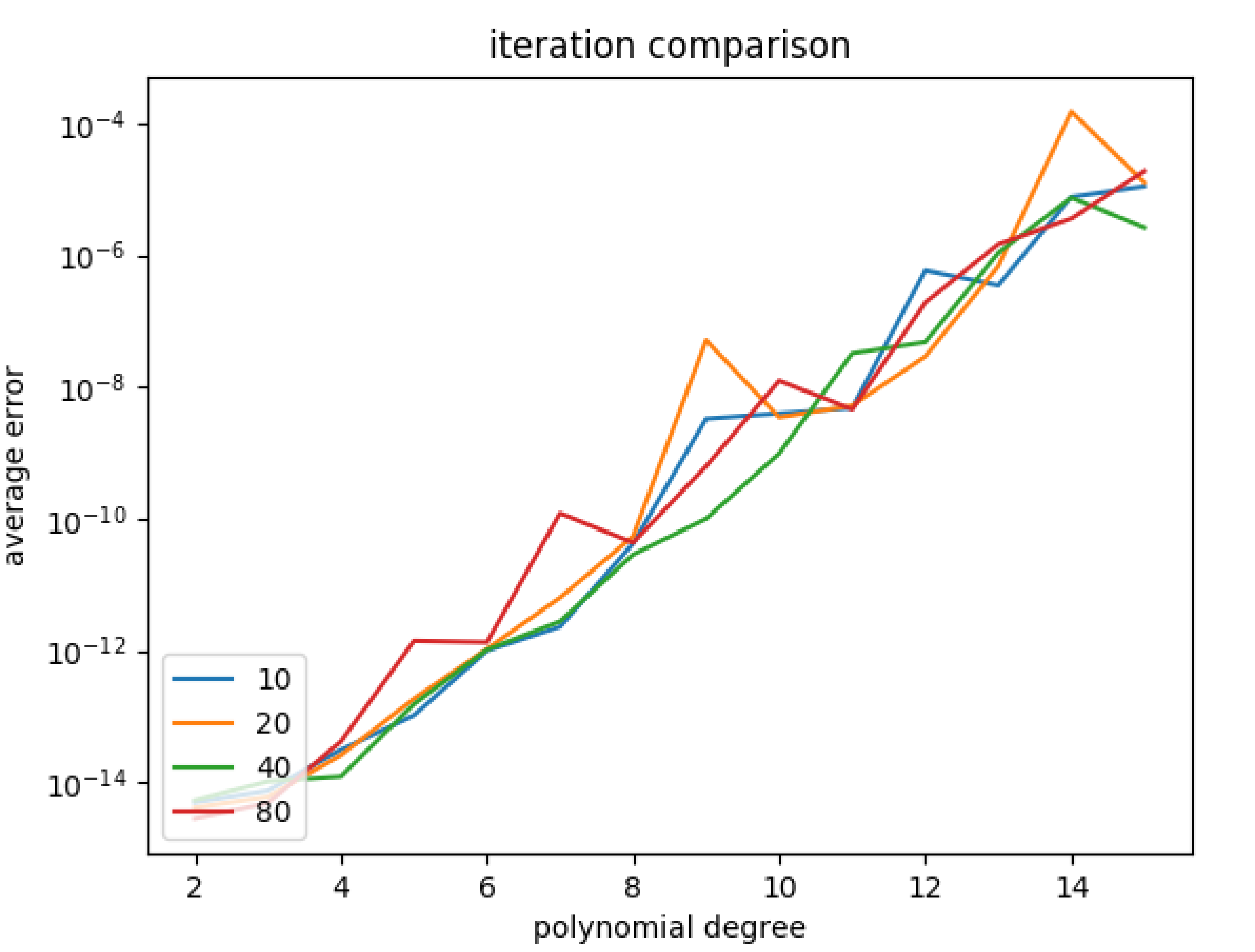
1.

Iteration Number



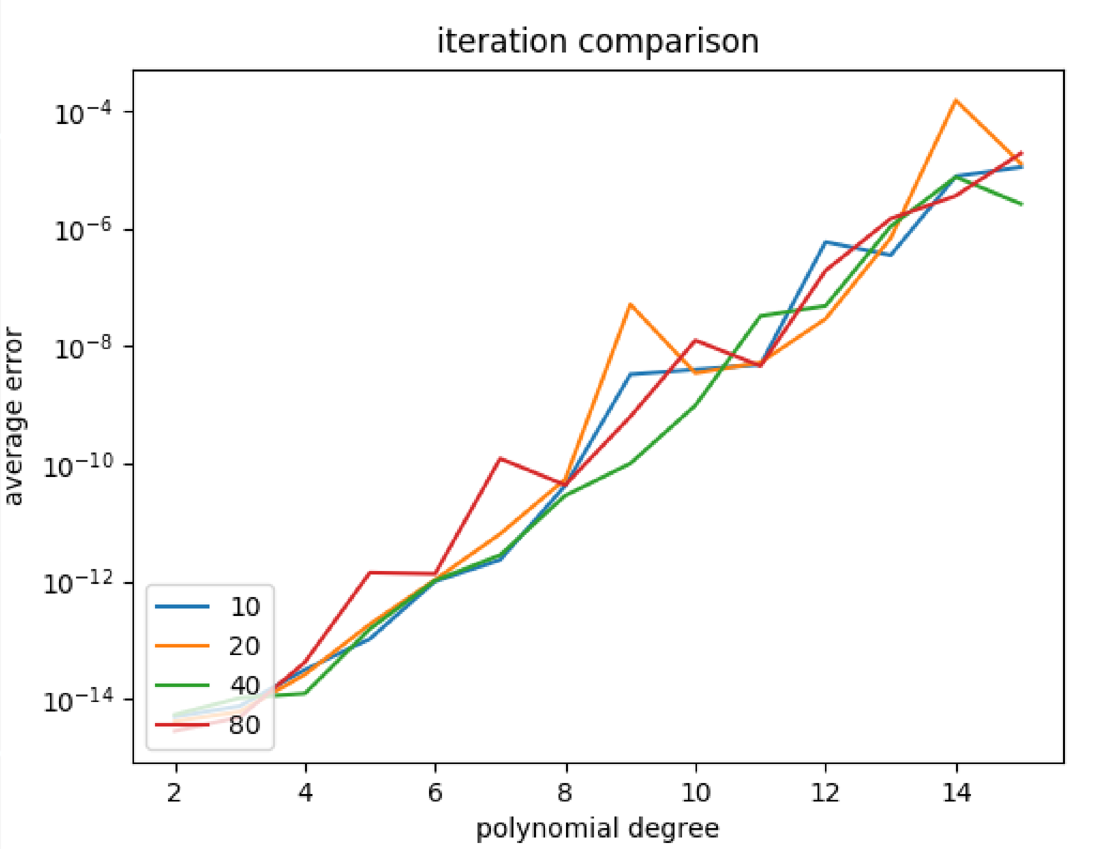
$$-2 \leq Re(y_i(0)) \leq 2$$
$$-2 \leq Im(y_i(0)) \leq 2$$
$$-10 \leq Re(x_i(0)) \leq 10$$
$$-10 \leq Im(x_i(0)) \leq 10$$
$$h = 0.01$$

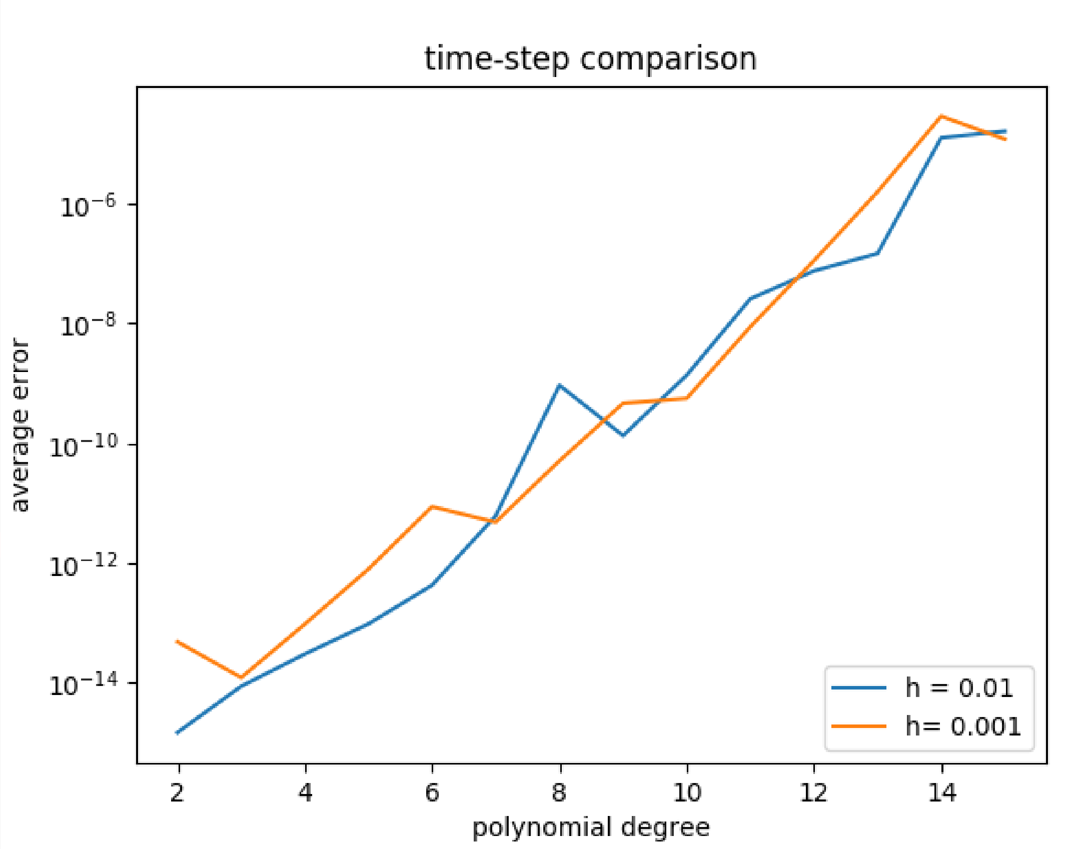
Iteration Number



$$-2 \le Re(y_i(0)) \le 2$$
$$-2 \le Im(y_i(0)) \le 2$$
$$-10 \le Re(x_i(0)) \le 10$$
$$-10 \le Im(x_i(0)) \le 10$$
$$h = 0.01$$

Combined





$$-2 \leq Re(y_i(0)) \leq 2$$
$$-2 \leq Im(y_i(0)) \leq 2$$
$$-10 \leq Re(x_i(0)) \leq 10$$
$$-10 \leq Im(x_i(0)) \leq 10$$
$$h = 0.01$$

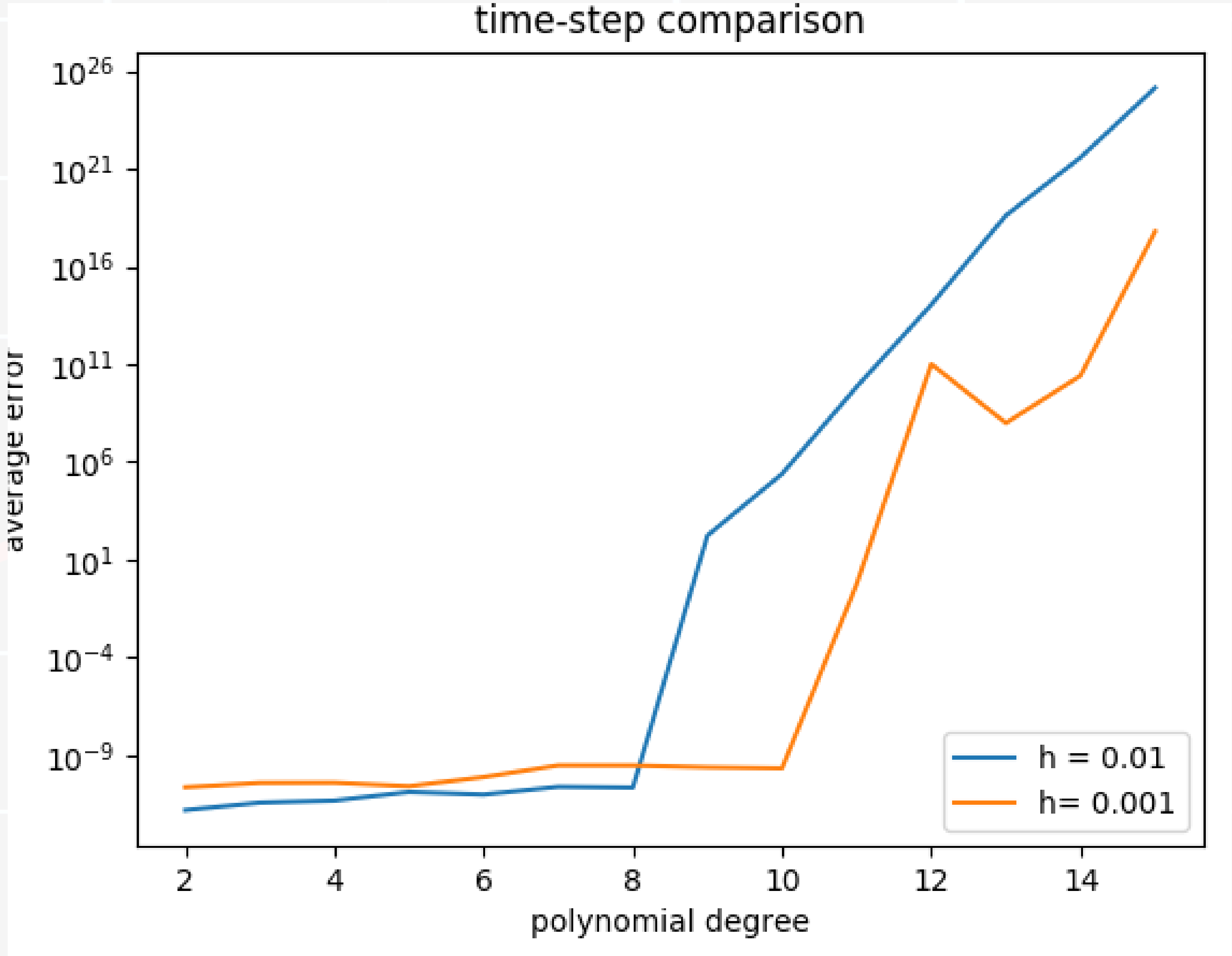
$$-2 \leq Re(y_i(0)) \leq 2$$
$$-2 \leq Im(y_i(0)) \leq 2$$
$$-10 \leq Re(x_i(0)) \leq 10$$
$$-10 \leq Im(x_i(0)) \leq 10$$
$$iterations = 10$$

What about polynomials with bigger coefficients?

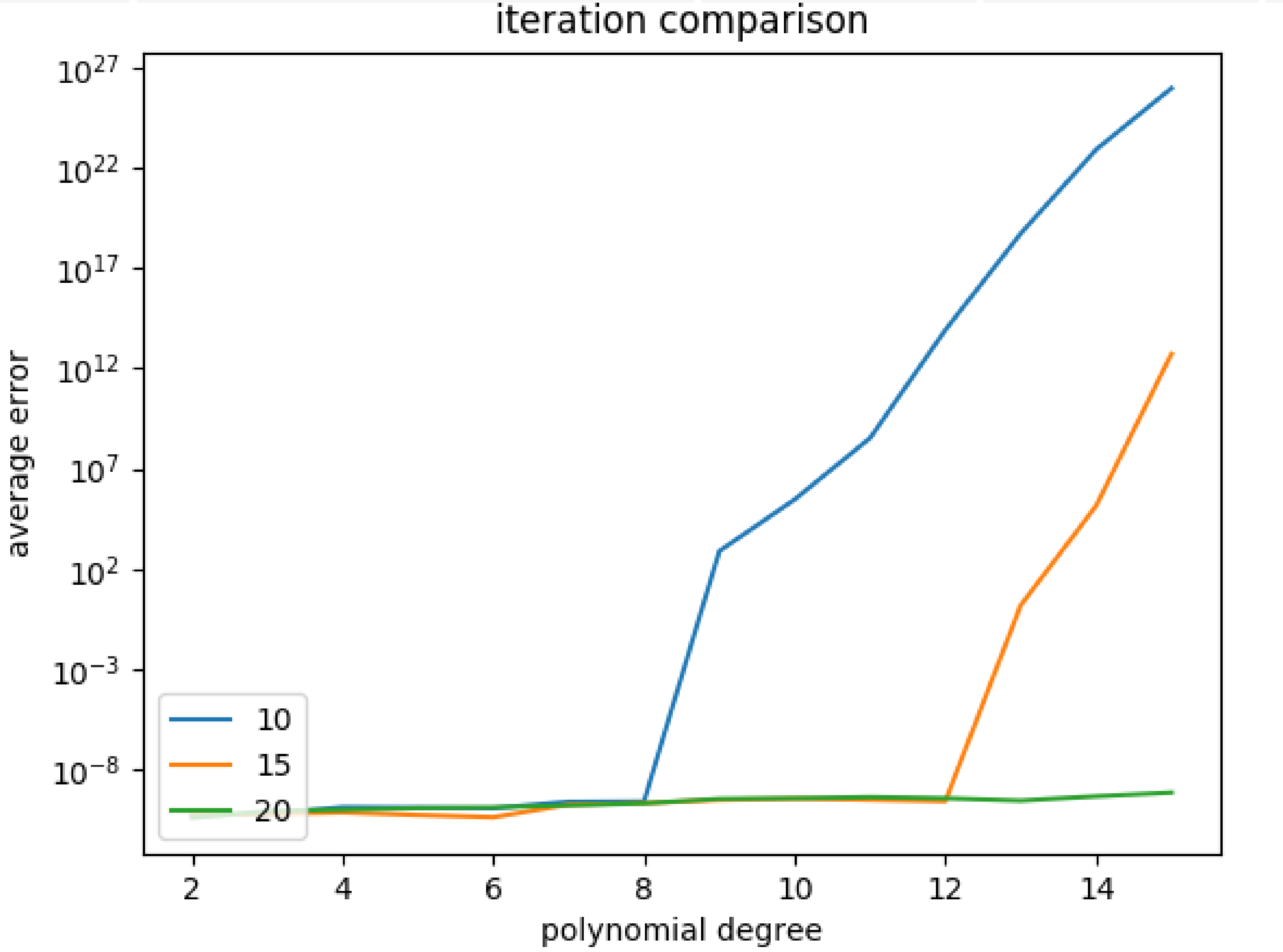
higher degree?

What about polynomials with bigger coefficients?

1. Generate N random coefficients and make a polynomial from them.
2. Find the zeros of this polynomial using the algorithm.
3. Solve for zeros numerically.
4. Expand the zeros to generate approximate coefficients.
5. Compare original coefficients with approximated coefficients.



$$-10000 \leq Re(y_i(0)) \leq 10000$$
$$-10000 \leq Im(y_i(0)) \leq 10000$$
$$-10000 \leq Re(c_m) \leq 10000$$
$$-10000 \leq Im(c_m) \leq 10000$$
$$iterations = 10$$

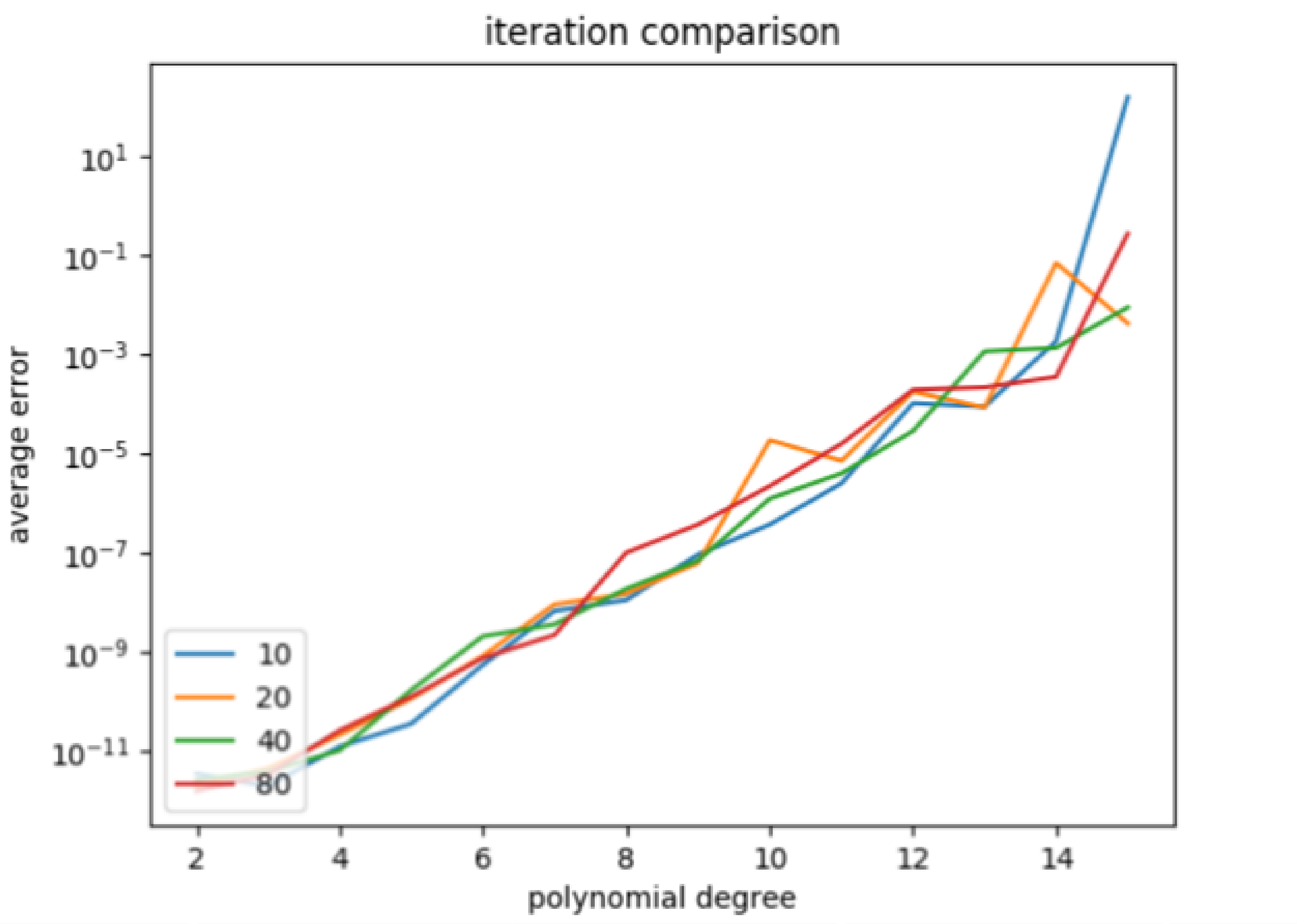


$$-10000 \leq Re(y_i(0)) \leq 10000$$
$$-10000 \leq Im(y_i(0)) \leq 10000$$
$$-10000 \leq Re(c_m) \leq 10000$$
$$-10000 \leq Im(c_m) \leq 10000$$
$$h = 0.01$$

Some Concluding thoughts…

Extra slides in case needed during questions. Not part of presentation.

Iteration Number



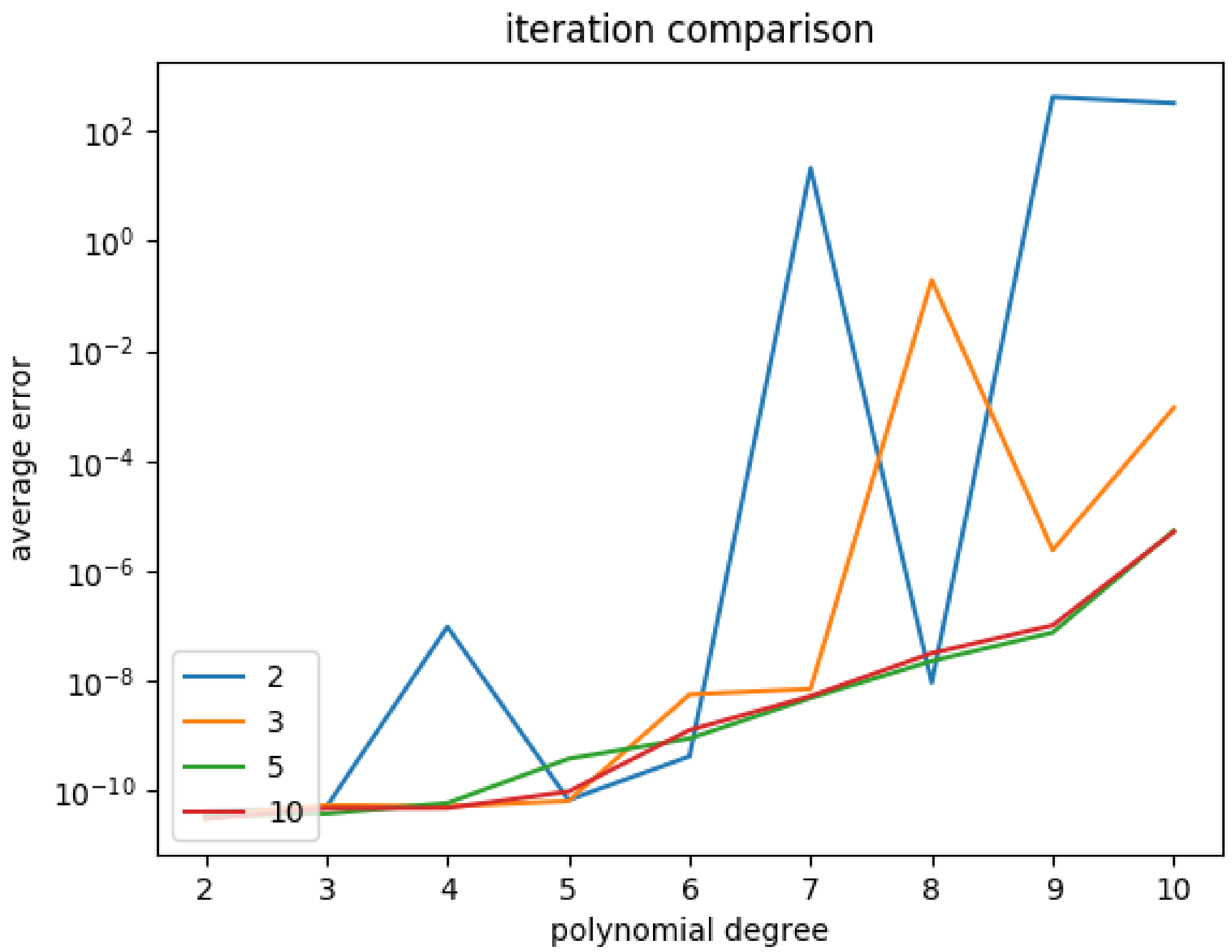
$$-1000 \leq Re(y_i(0)) \leq 1000$$
$$-1000 \leq Im(y_i(0)) \leq 1000$$
$$-1000 \leq Re(x_i(0)) \leq 1000$$
$$-1000 \leq Im(x_i(0)) \leq 1000$$
$$h = 0.01$$

Original Coefficients
26.64217938+52.97154236i
5.47138843+65.89068862i
2.28886421+28.28523669i
69.78152008+93.94533544i
75.10168980+42.09074322i
38.94603962 +6.62592201i
97.17776530+27.83221136i

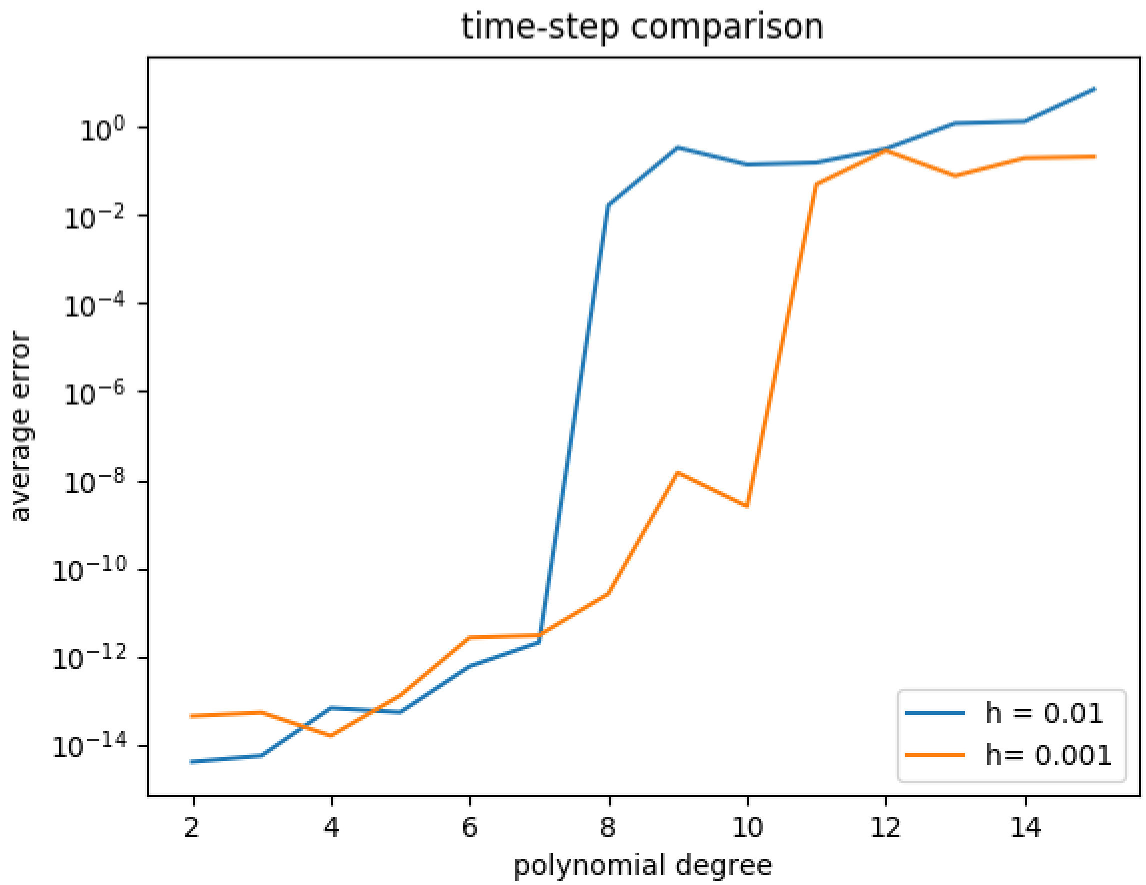
10 iterations

Coefficient error
2.08370334311e-12

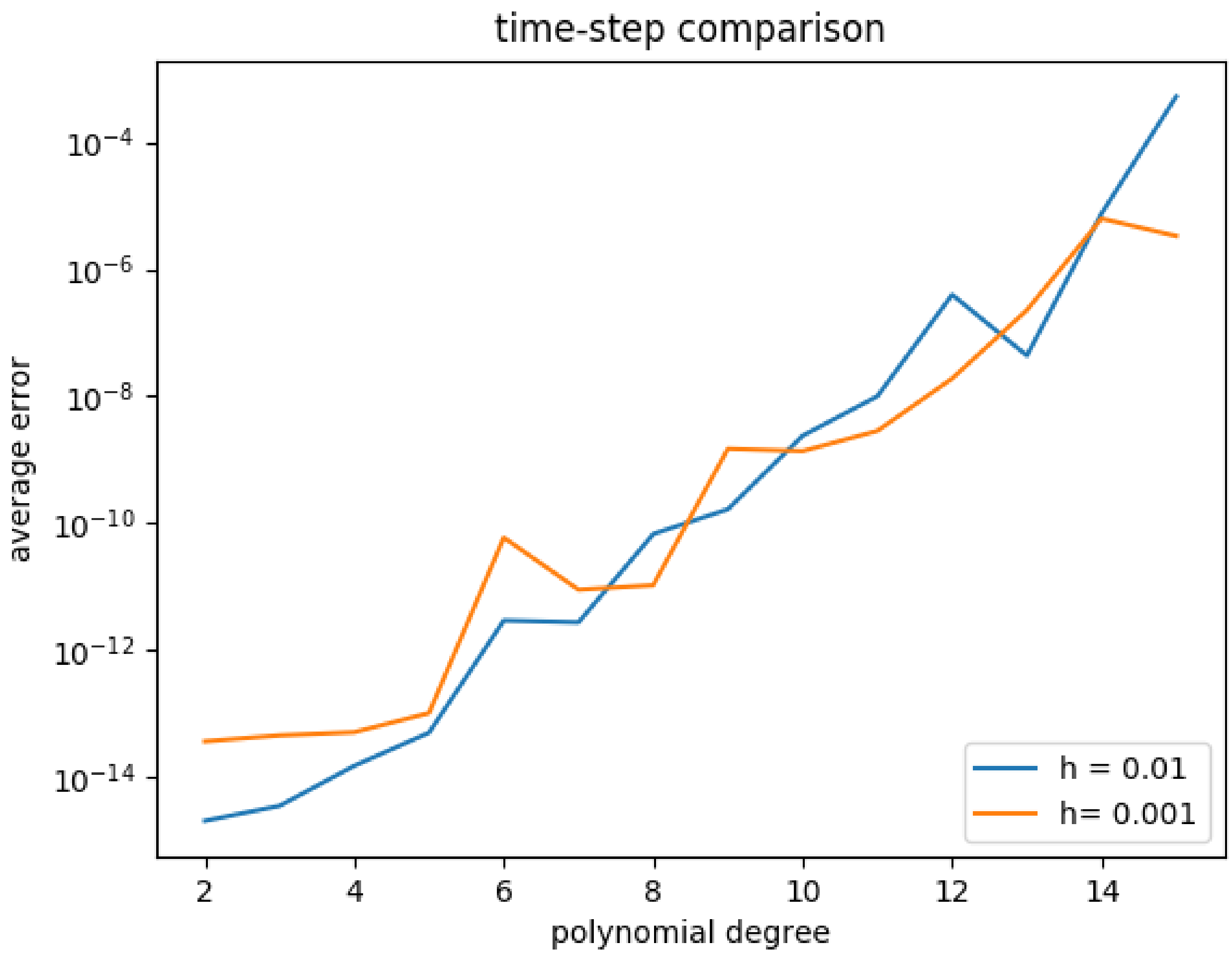
Evaluation error
0.00311678812213



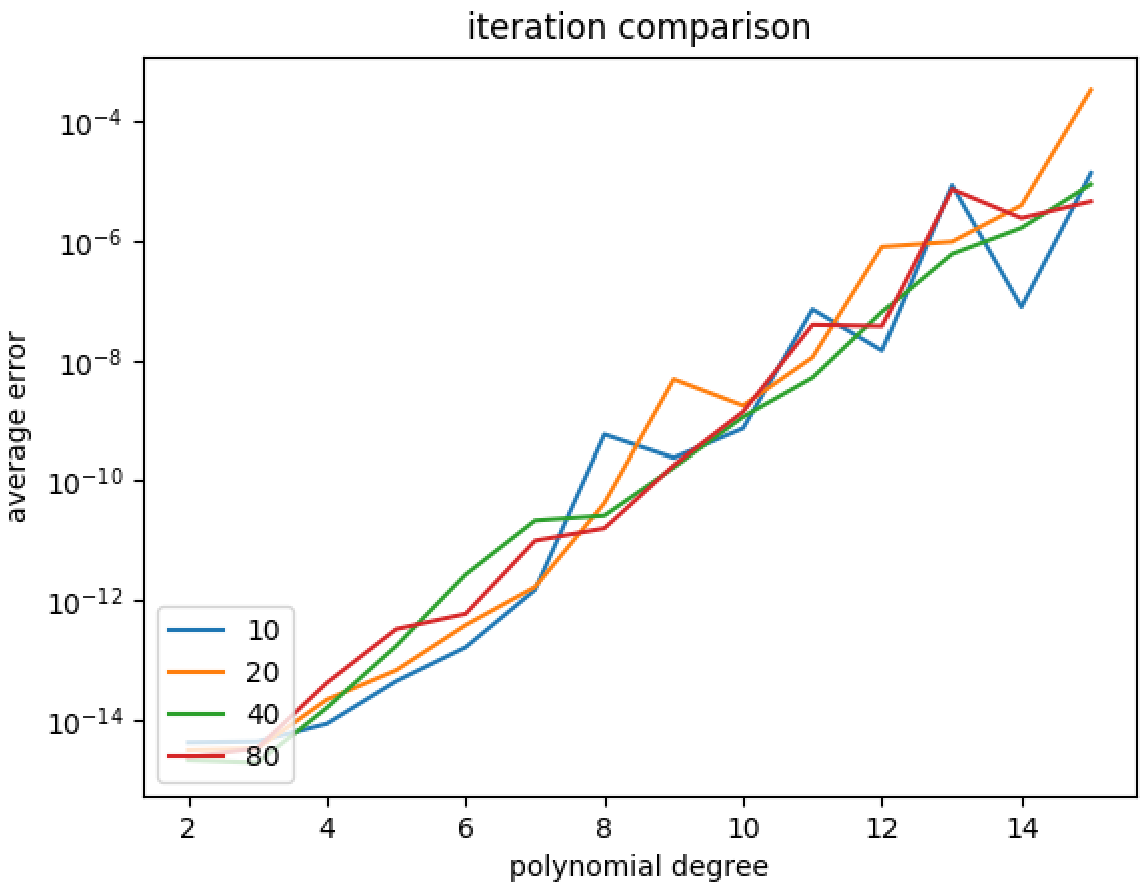
$$-1000 \leq y_i(0) \leq 1000$$
$$-1000 \leq x_i \leq 1000$$
$$h = 0.001$$



$$-10 \leq y_i(0) \leq 10$$
$$-1.000001 \leq x_i \leq 1.000001$$
$$10\ iterations$$

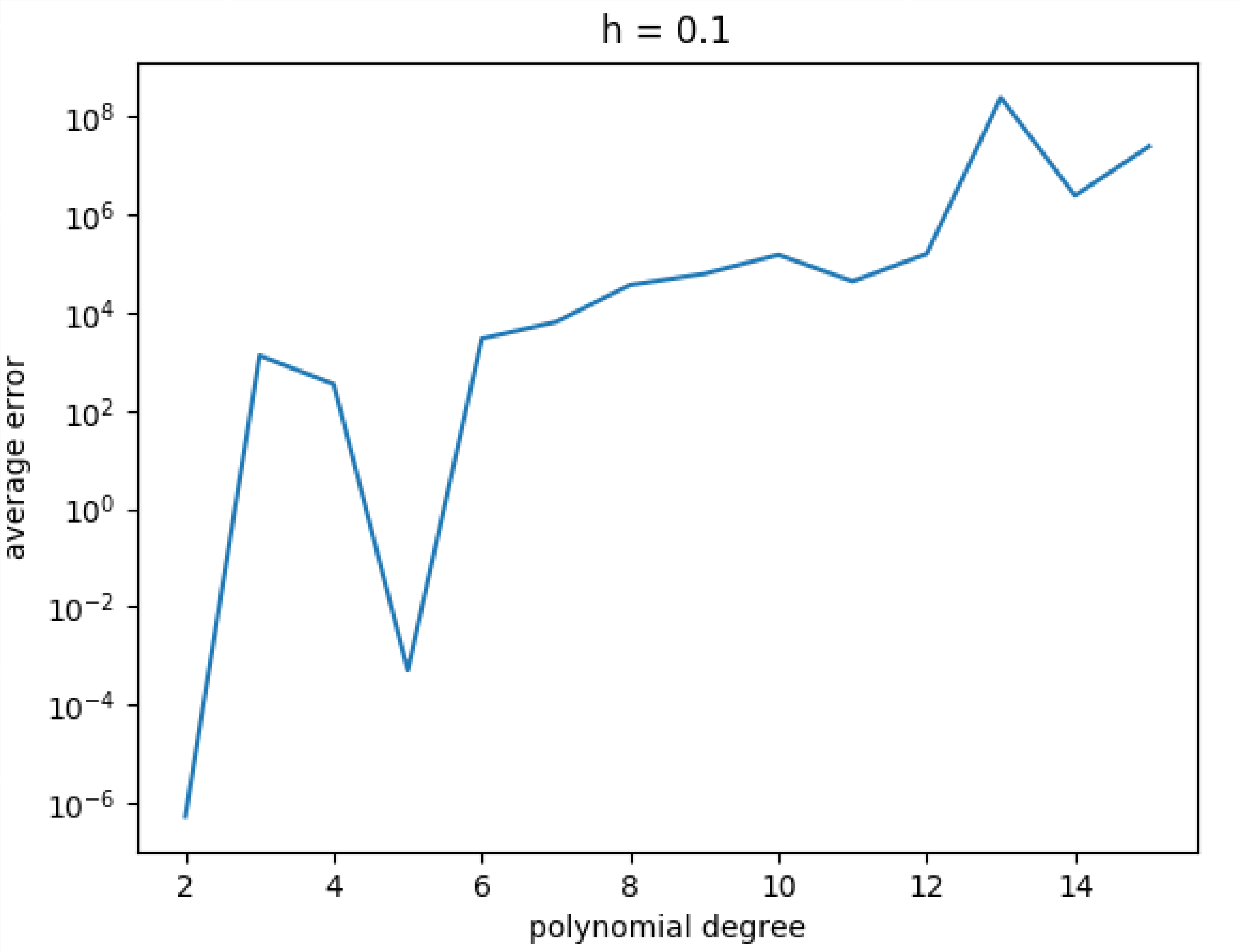


$$-10 \leq y_i(0) \leq 10$$
$$-1.000000001 \leq x_i \leq 1.000000001$$
$$10 \ iterations$$



$$-10 \leq y_i(0) \leq 10$$
$$-1.000001 \leq x_i \leq 1.000001$$
$$h = 0.01$$

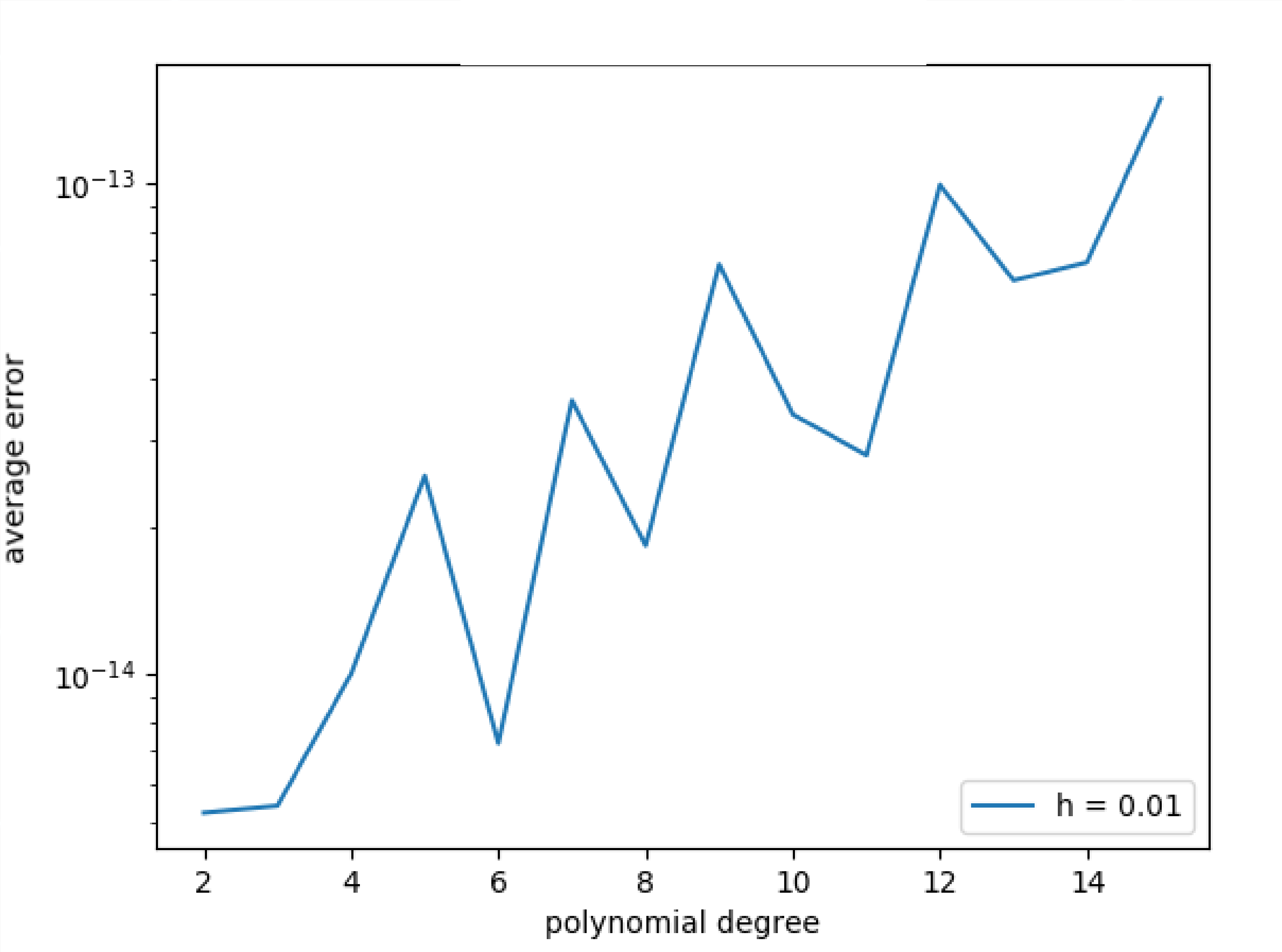
Time step



$-1000 \leq Re(y_i(0)) \leq 1000$
$-1000 \leq Im(y_i(0)) \leq 1000$
$-1000 \leq Re(x_i(0)) \leq 1000$
$-1000 \leq Im(x_i(0)) \leq 1000$

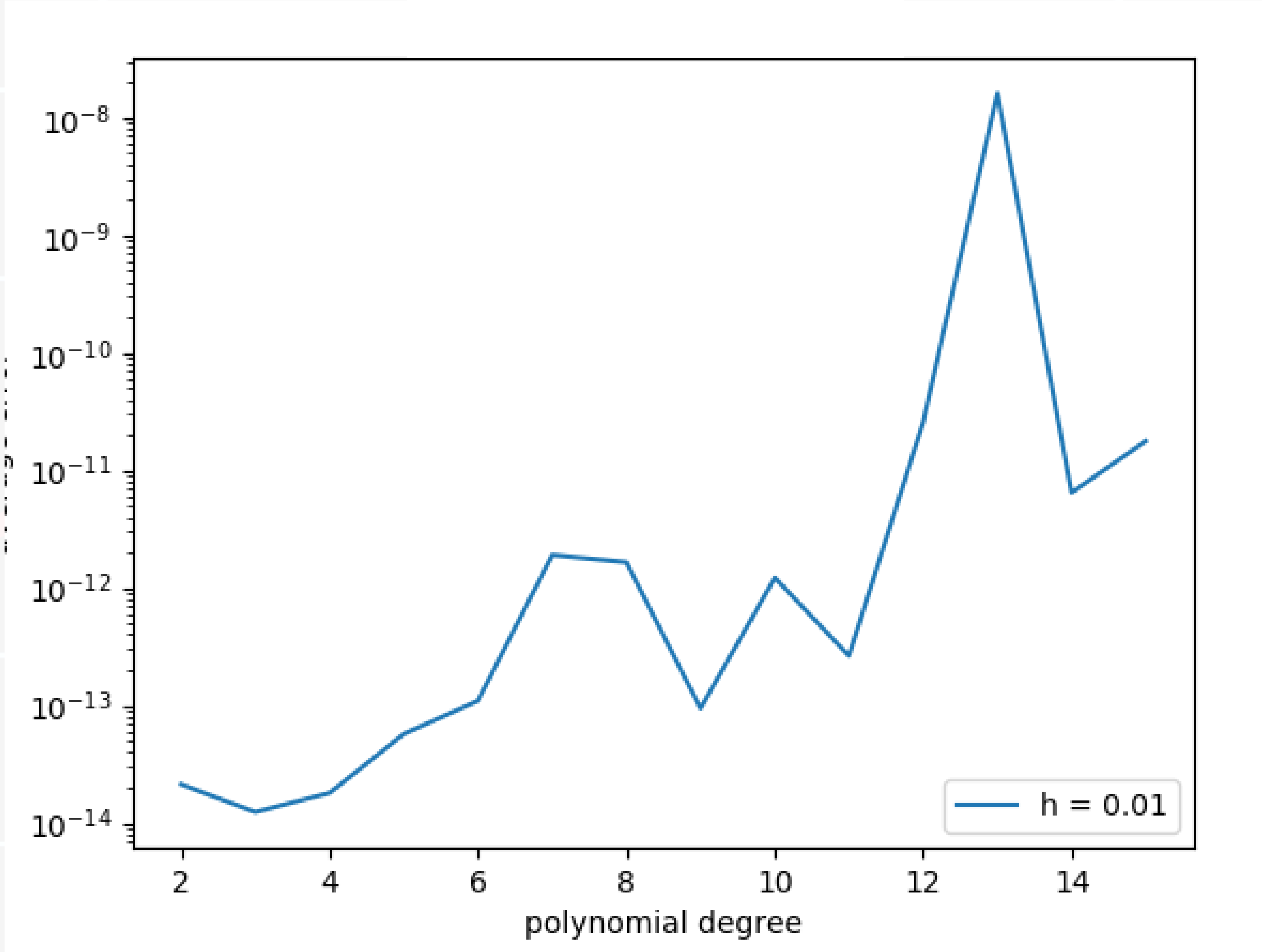
10 iterations

How about evaluating the zero?



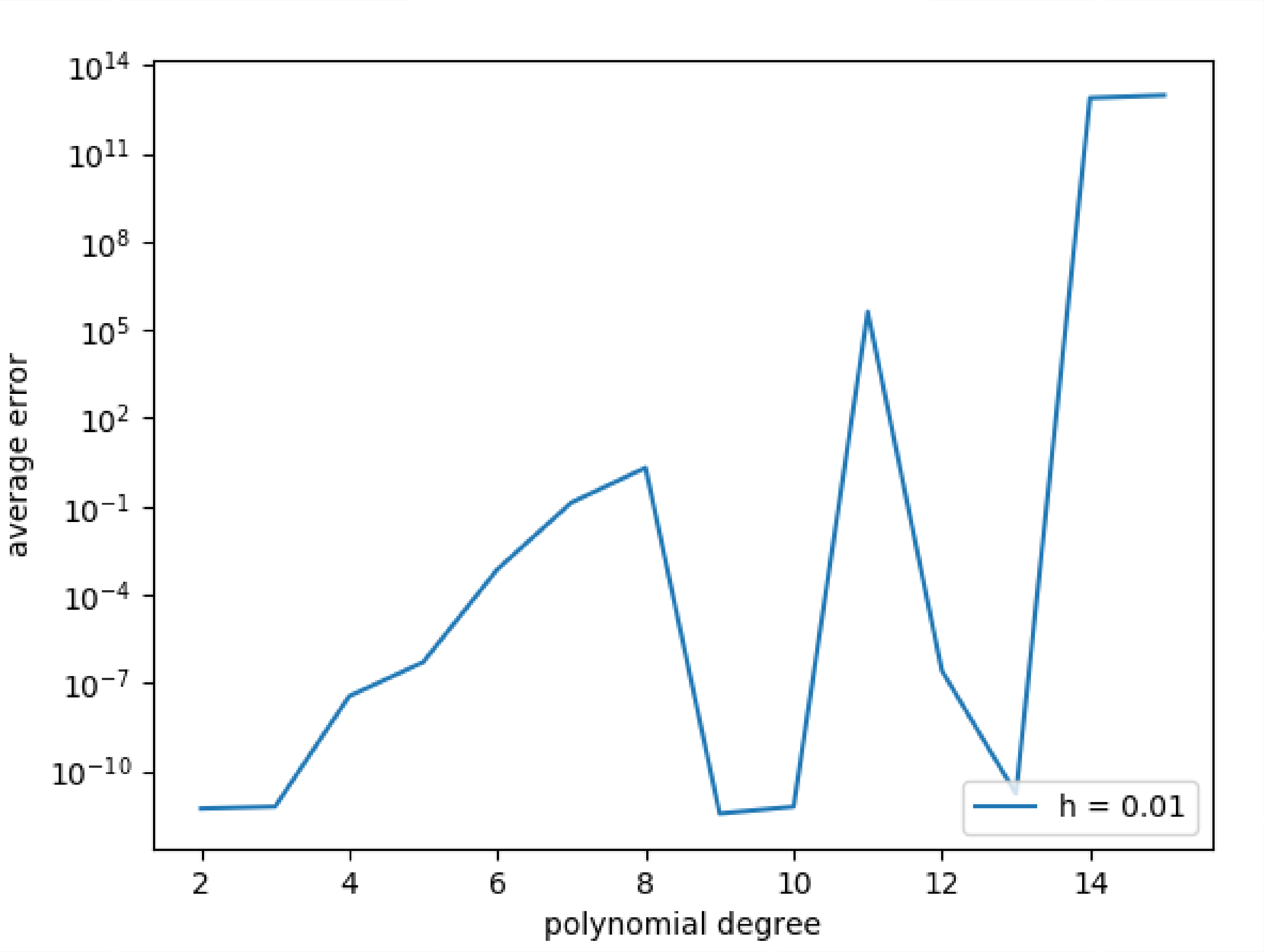
$$-10 \leq Re(y_i(0)) \leq 10$$
$$-10 \leq Im(y_i(0)) \leq 10$$
$$-2 \leq Re(c_n) \leq 2$$
$$-2 \leq Im(c_n) \leq 2$$
$$iterations = 10$$

How about evaluating the zero?



$$-10 \leq Re(y_i(0)) \leq 10$$
$$-10 \leq Im(y_i(0)) \leq 10$$
$$-5 \leq Re(c_n) \leq 5$$
$$-5 \leq Im(c_n) \leq 5$$
$$iterations = 10$$

How about evaluating the zero?



$$-100 \leq Re(y_i(0)) \leq 100$$
$$-100 \leq Im(y_i(0)) \leq 100$$
$$-100 \leq Re(c_n) \leq 100$$
$$-100 \leq Im(c_n) \leq 100$$
$$iterations = 10$$