Higher-Order Elements and the Lorenz System

Anders Logg

logg@math.chalmers.se

Chalmers Finite Element Center

Outline

- Multi-adaptive Galerkin
- Higher-order elements
- Solving the Lorenz system on $[0, T]$

Ordinary Galerkin

Ordinary Galerkin $\mathrm{cG}(q)$ for $\dot{u} = f$:

$$\int_0^T (\dot{U}, v)\, dt = \int_0^T (f(U, \cdot), v)\, dt \;\forall v \in W,$$

with $U \in V$, $U(0) = u_0$ and the trial and test spaces defined as

$$\begin{aligned} V &= \{v \in C^N([0,T]) : v_i|_{I_j} \in \mathcal{P}^q(I_j)\}, \\ W &= \{v : v_i|_{I_j} \in \mathcal{P}^{q-1}(I_j)\}. \end{aligned}$$

Multi-Adaptive Galerkin

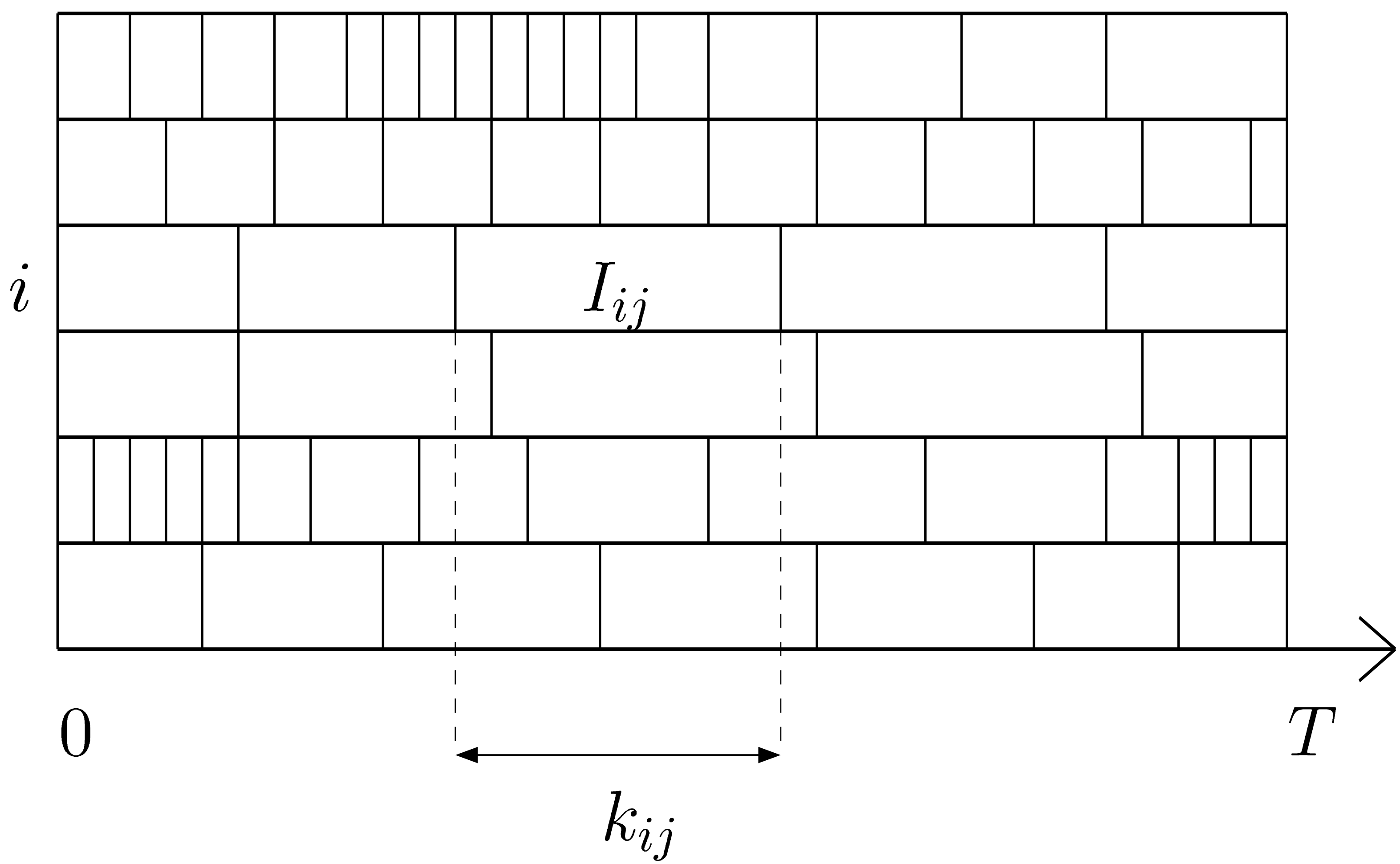
Multi-adaptive Galerkin $\mathrm{mcG}(q)$:

$$\int_0^T (\dot{U}, v)\, dt = \int_0^T (f(U,\cdot), v)\, dt \ \forall v \in W,$$

with $U \in V$, $U(0) = u_0$ and the trial and test spaces now defined as

$$\begin{aligned} V &= \{v \in C^N([0,T]) : v_i|_{I_{ij}} \in \mathcal{P}^{q_{ij}}(I_{ij})\}, \\ W &= \{v : v_i|_{I_{ij}} \in \mathcal{P}^{q_{ij}-1}(I_{ij})\}. \end{aligned}$$

Individual Timesteps



The Discrete Equations

Making an ansatz for U_i on I_{ij},

(1) $$U_i(t) = \sum_{n=0}^{q_{ij}} \xi_{ijn} \lambda_n^{[q_{ij}]}(\tau_{ij}(t)),$$

we end up with

(2) $$\xi_{ijn} = \xi_0 + \int_{I_{ij}} w_n^{[q_{ij}]}(\tau_{ij}(t)) \; f_i(U,t) \; dt,$$

for certain weight functions $\{w_n^{[q]}\} \subset \mathcal{P}^{q-1}(0,1)$.

The Discontinuous Method

Similarly for the multi-adaptive version of the discontinuous method, $\mathrm{mdG}(q)$, we obtain

(3) $$\xi_{ijm} = \xi_{ij0}^{-} + \int_{I_{ij}} w_m^{[q_{ij}]}(\tau_{ij}(t)) \; f_i(U,t) \; dt,$$

for certain weight functions $\{w_n^{[q]}\} \subset \mathcal{P}^q(0,1)$.

General Order

- The $\mathrm{mcG}(q)$ method is of order $2q$, i.e. locally of order $2q_{ij}$.
- The $\mathrm{mdG}(q)$ method is of order $2q+1$, i.e. locally of order $2q_{ij}+1$.

Simple Galerkin thus gives methods of arbitrary order with individual timesteps, such as e.g. the 40:th order method $\mathrm{mcG}(20)$.

Error Estimates

- A priori error estimates,

(4) $$\|e\| \leq \int_0^T (k^{2q}, w)\, dt,$$

(5) $$\|e\| \leq \int_0^T (k^{2q+1}, w)\, dt.$$

- A posteriori error estimates,

(6) $$\|e\| \leq E_G + E_C + E_Q.$$

Galerkin Errors

- Many different variants
- More or less local/global
- Two extremes (for the continuous method):

$$E_G = \left| \int_0^T (R, \varphi - \pi_k \varphi) \, dt \right| \leq \sum_{i=1}^{N} S_i \max\{C_{q_i} k_i^{q_i} |R_i|\}$$

(7)

Computational Errors

(8) $$E_C \approx \sum_{i=1}^{N} \bar{S}_i^{[0]} \max_{[0,T]} |R_i^C|,$$

where

$$R_i^C = \frac{1}{k_{ij}} \left[\int_{I_{ij}} f_i(U, \cdot)\, dt - (U(t_{ij}) - U(t_{i,j-1})) \right]$$

is the *discrete residual*. (For the discontinuous method, $t_{ij}, t_{i,j-1}$ are replaced by $t_{ij}^-, t_{i,j-1}^-$.)

$\rightarrow$

Quadrature Errors

$$E_Q \approx \sum_{i=1}^{N} \bar{S}_i^{[0]} \max_{[0,T]} |R_i^Q|, \tag{9}$$

where

$$R_i^Q = \frac{1}{k_{ij}} \left[\int_{I_{ij}} f_i(U,\cdot)\, dt - \tilde{\int}_{I_{ij}} f_i(U,\cdot)\, dt \right]$$

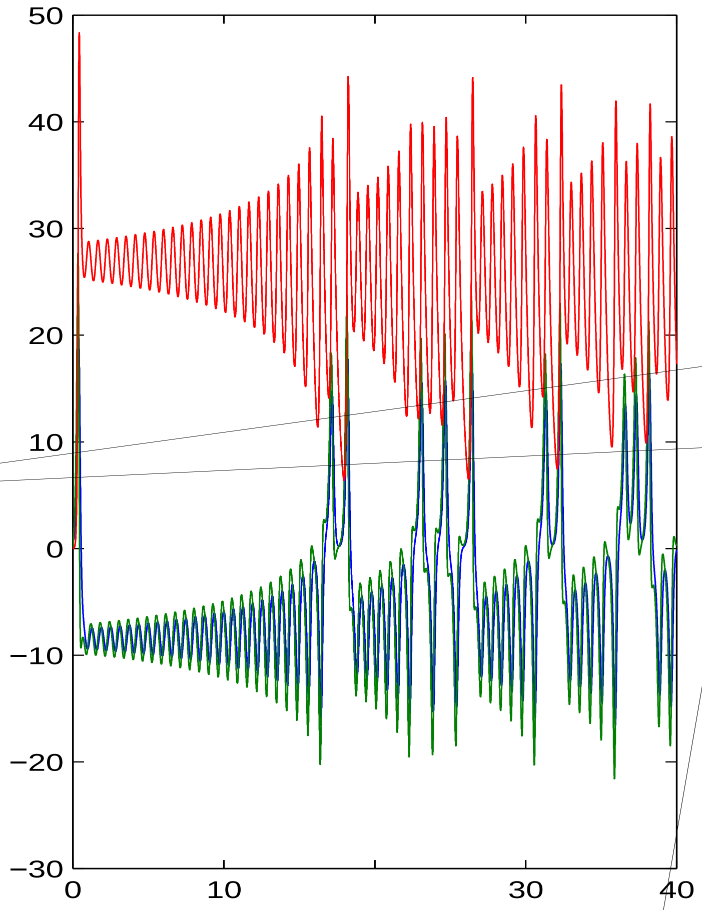
is the *quadrature residual*.

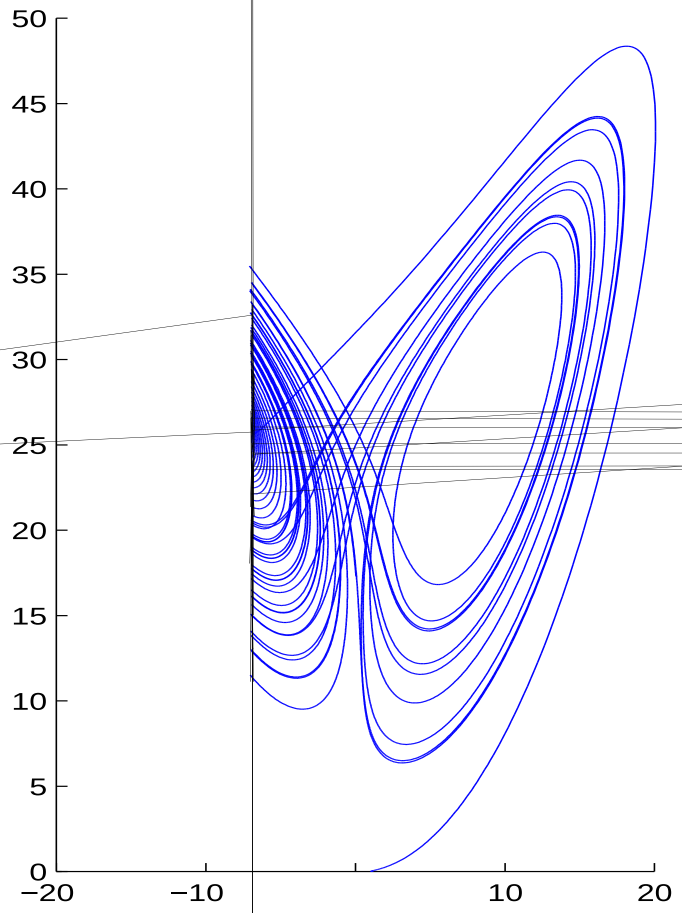
The Lorenz System

$$\left\{\begin{array}{rcl}\dot{x} & = & \sigma(y-x),\\ \dot{y} & = & rx-y-xz,\\ \dot{z} & = & xy-bz,\end{array}\right. \tag{10}$$

where we, as ususal, take $(x_0, y_0, z_0) = (1, 0, 0)$, $\sigma = 10$, $b = 8/3$ and $r = 28$.

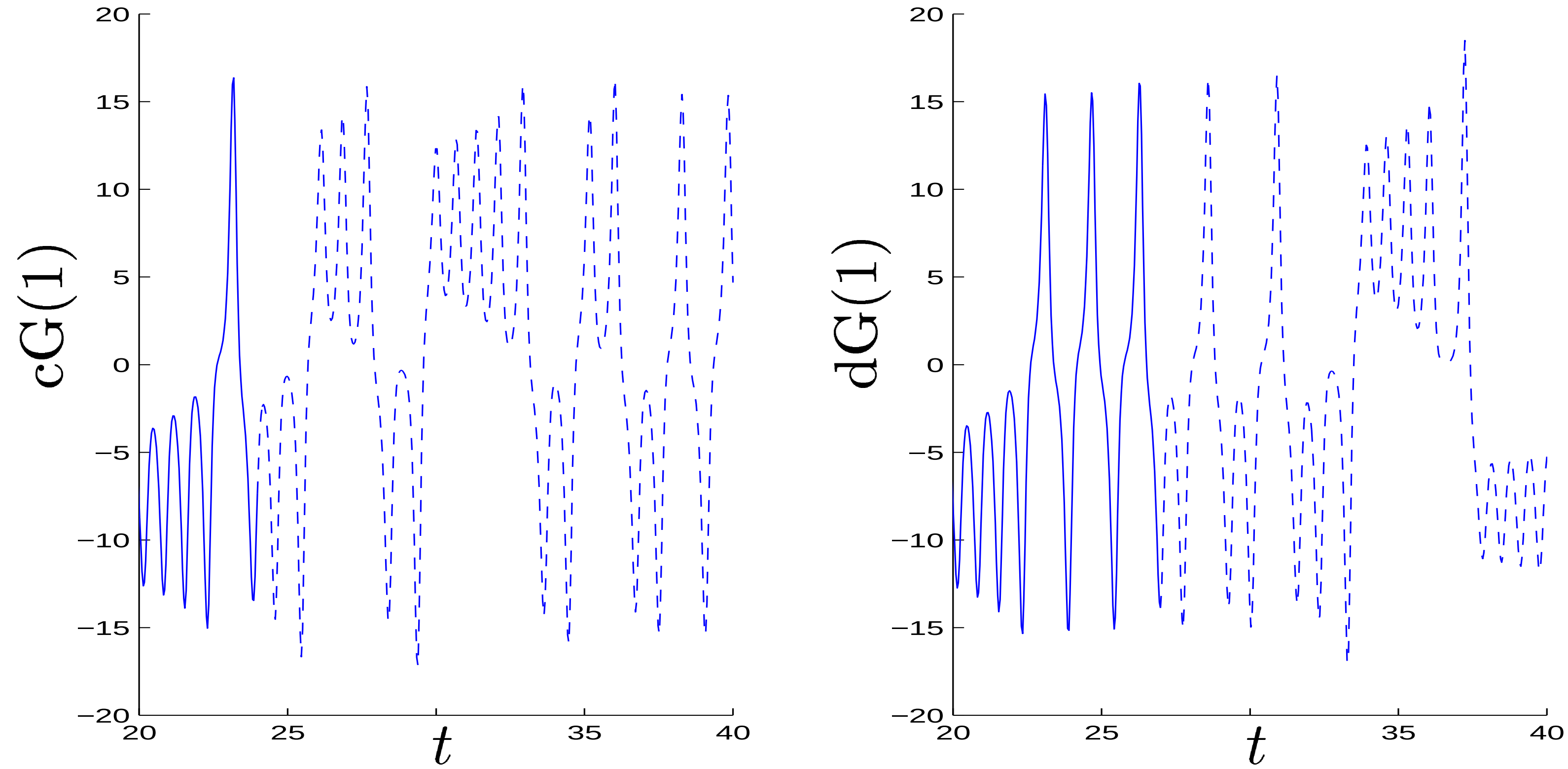
The Solution?





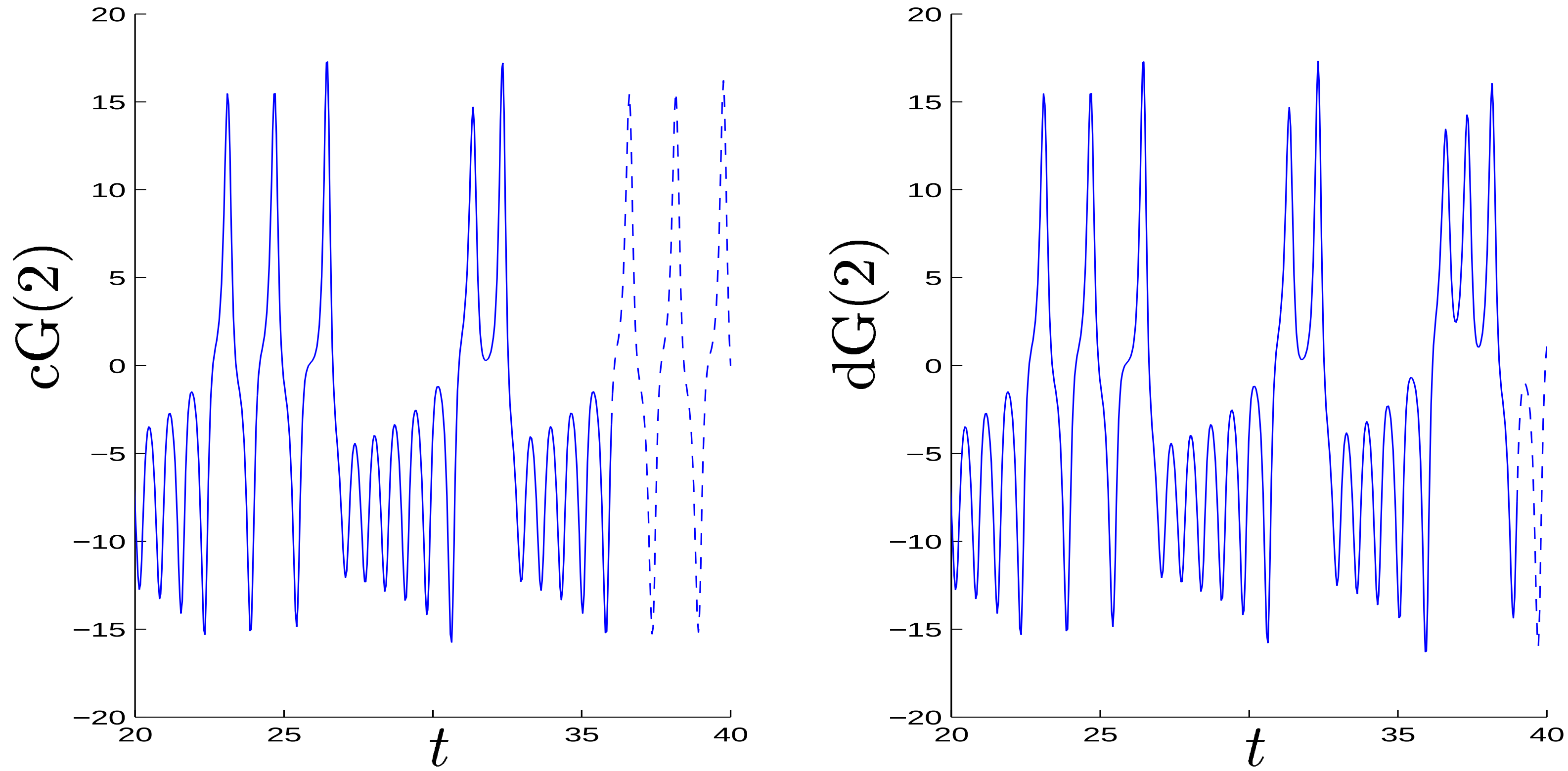
A Simple Experiment

- $T = 40$
- $k = 0.001$



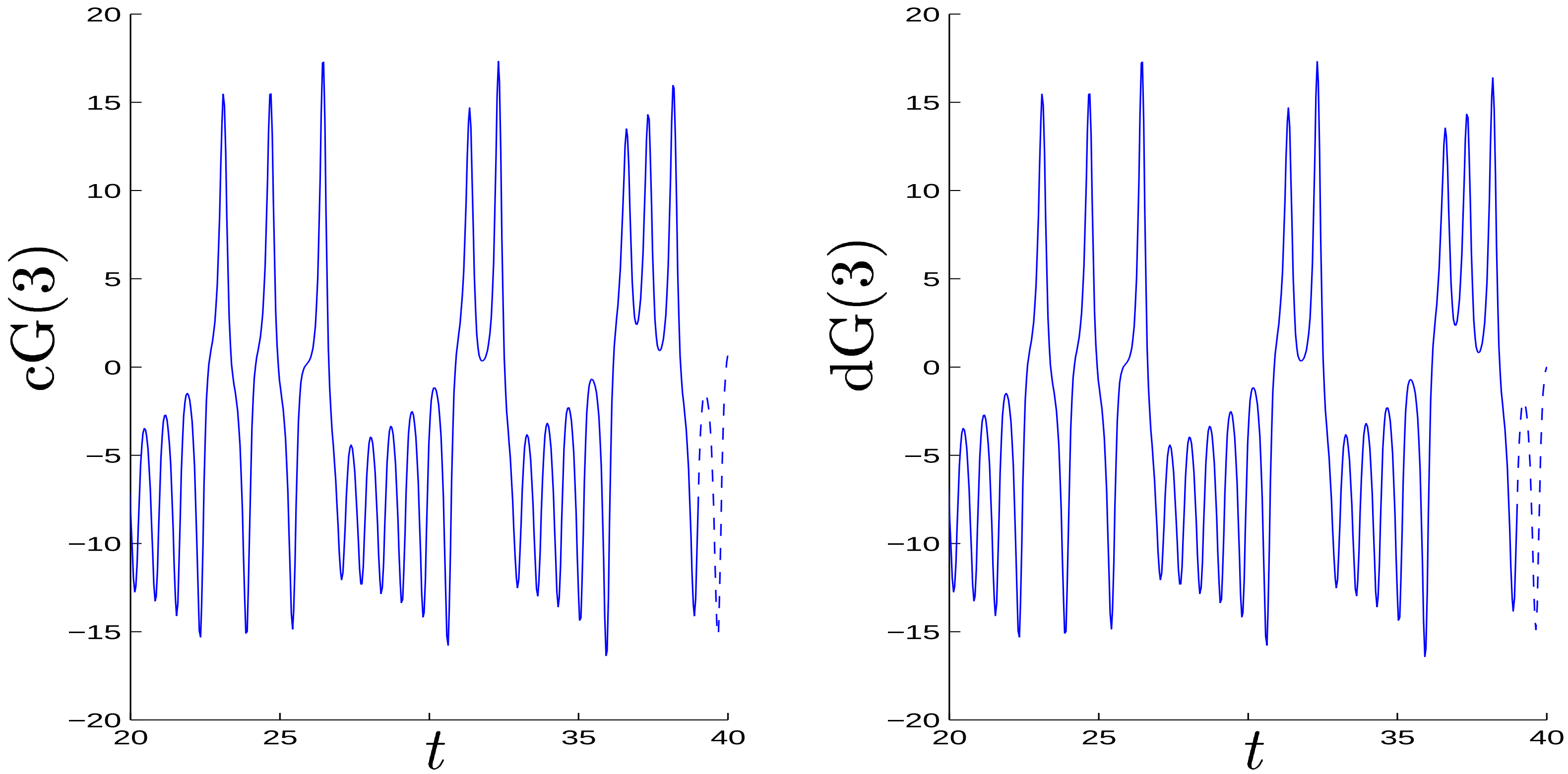
A Simple Experiment

- $T = 40$
- $k = 0.001$



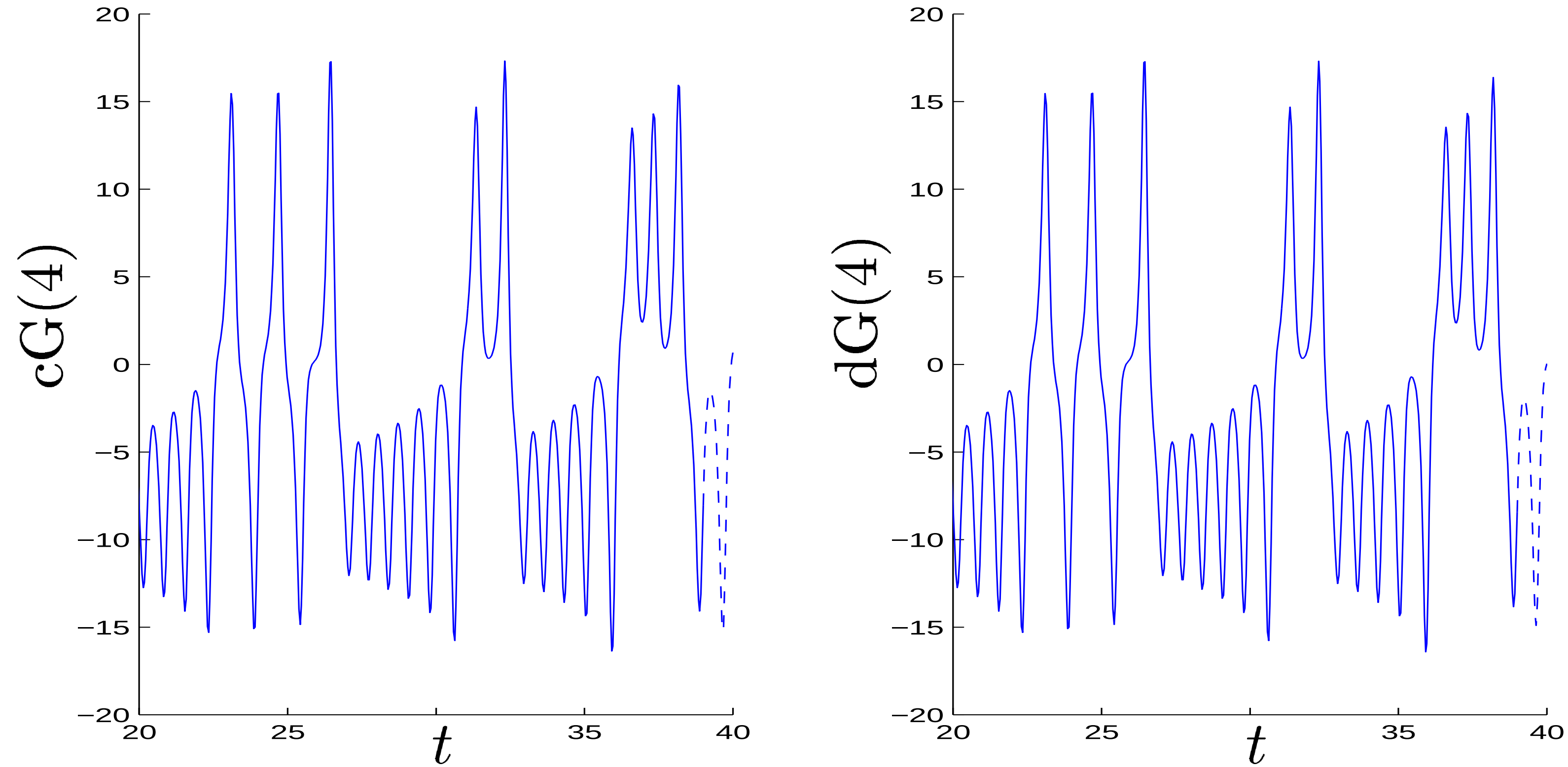
A Simple Experiment

- $T = 40$
- $k = 0.001$



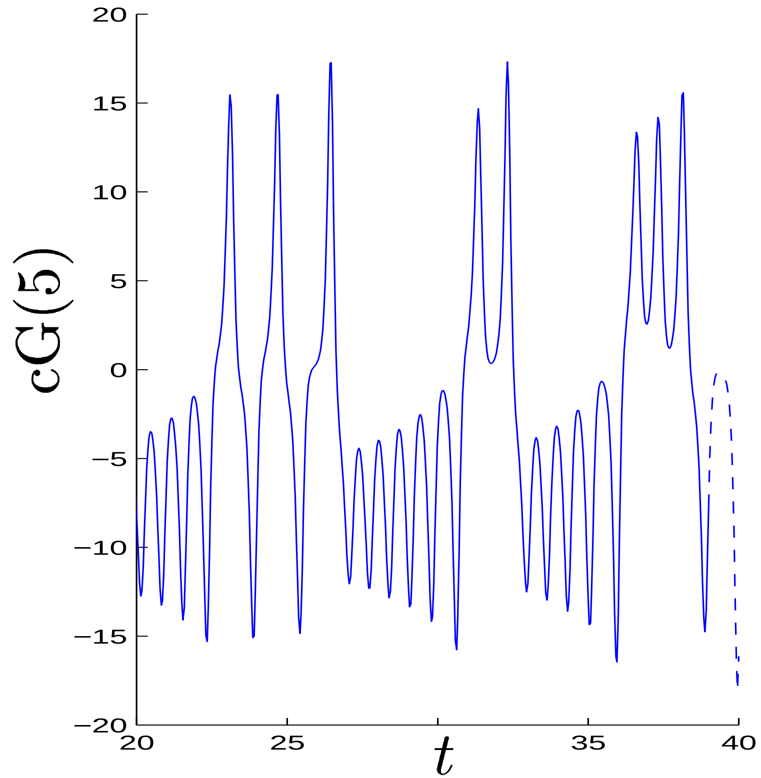
A Simple Experiment

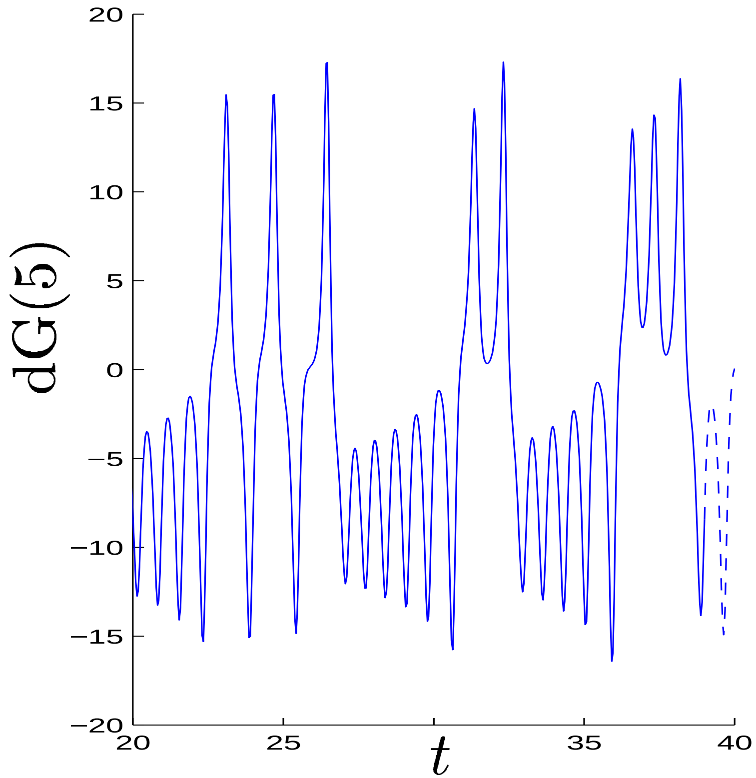
- $T = 40$
- $k = 0.001$



A Simple Experiment

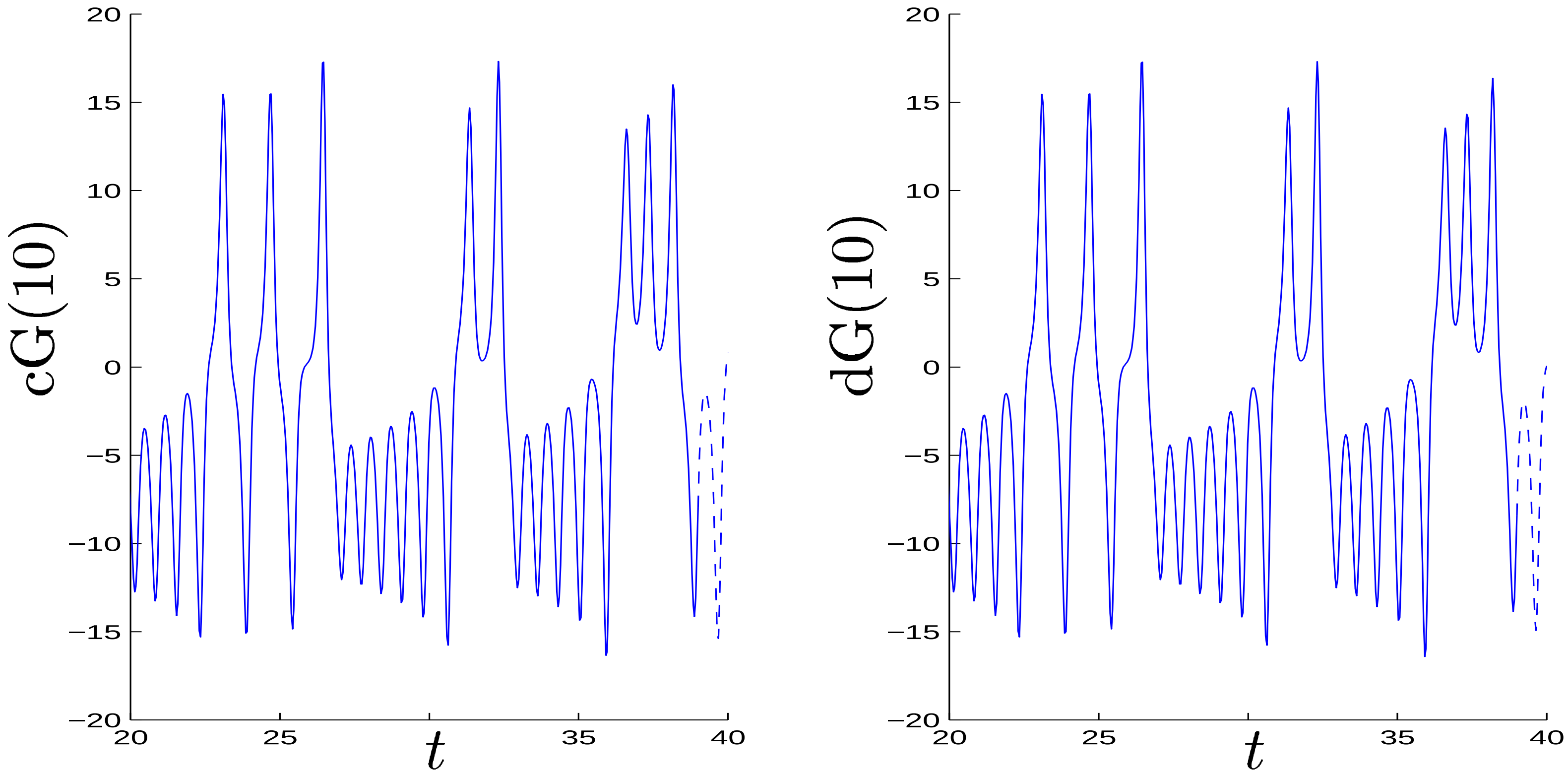
- $T = 40$
- $k = 0.001$



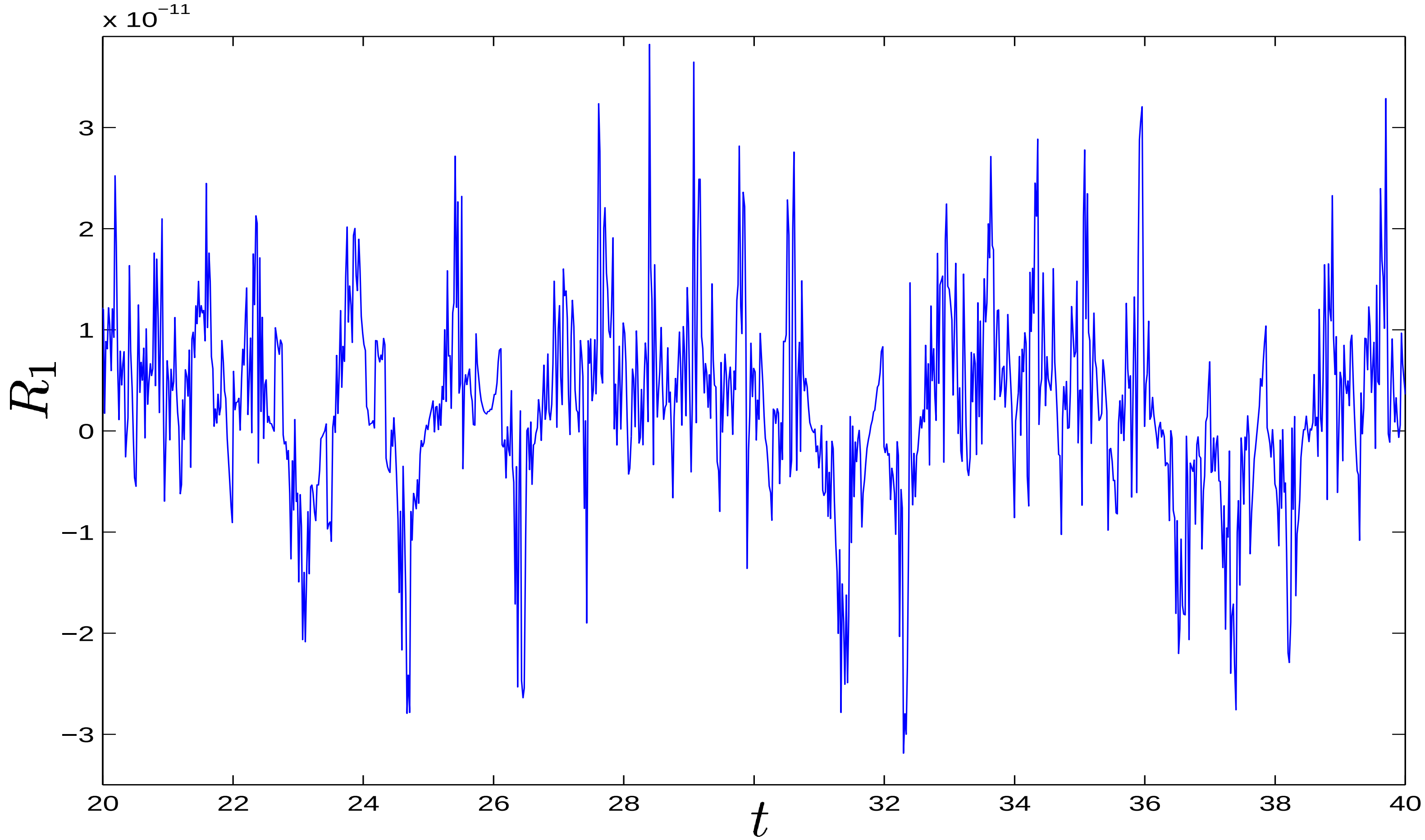


A Simple Experiment

- $T = 40$
- $k = 0.001$



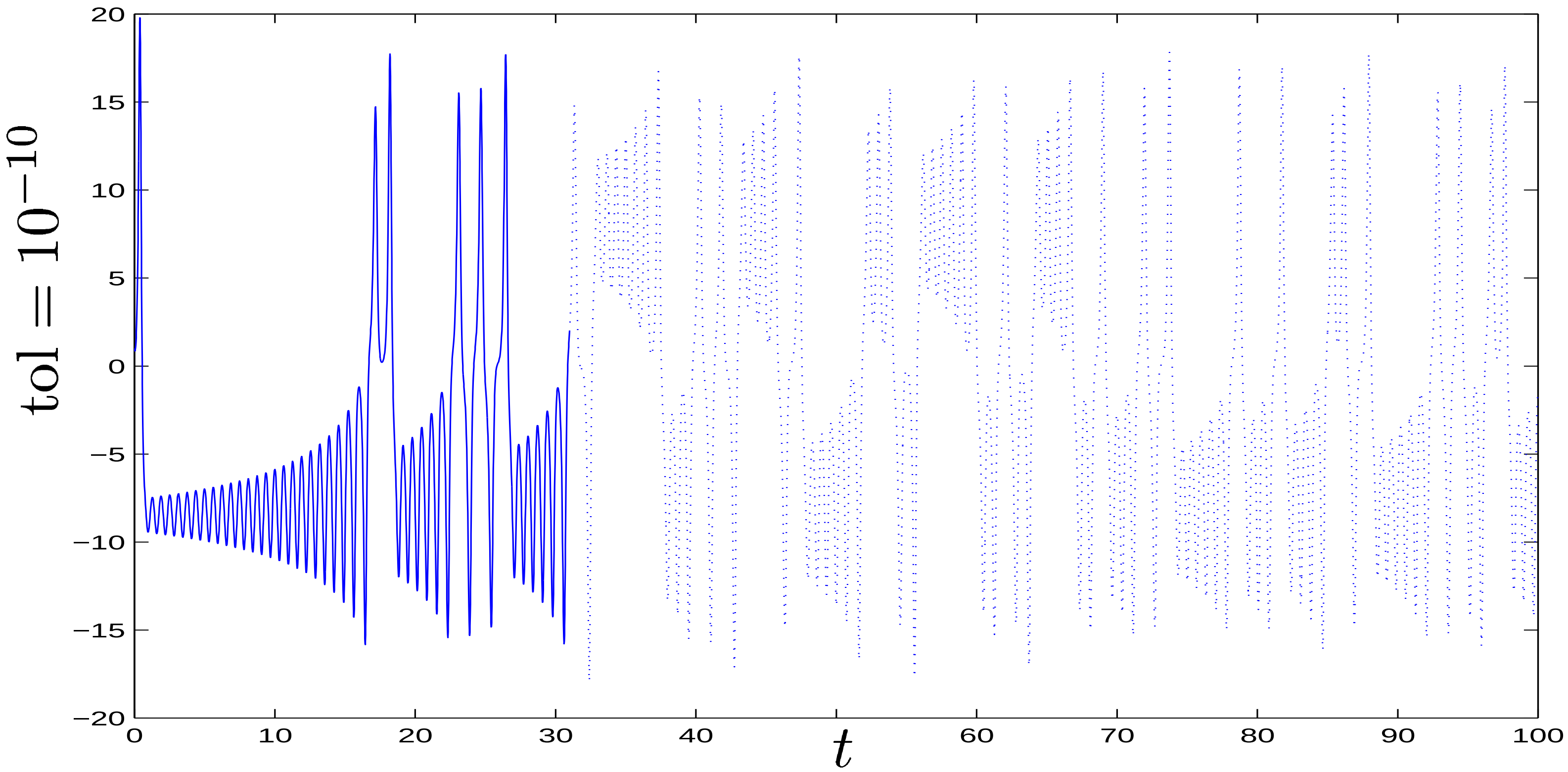
The Residual



[cG(5) residual with $k = 0.001$]

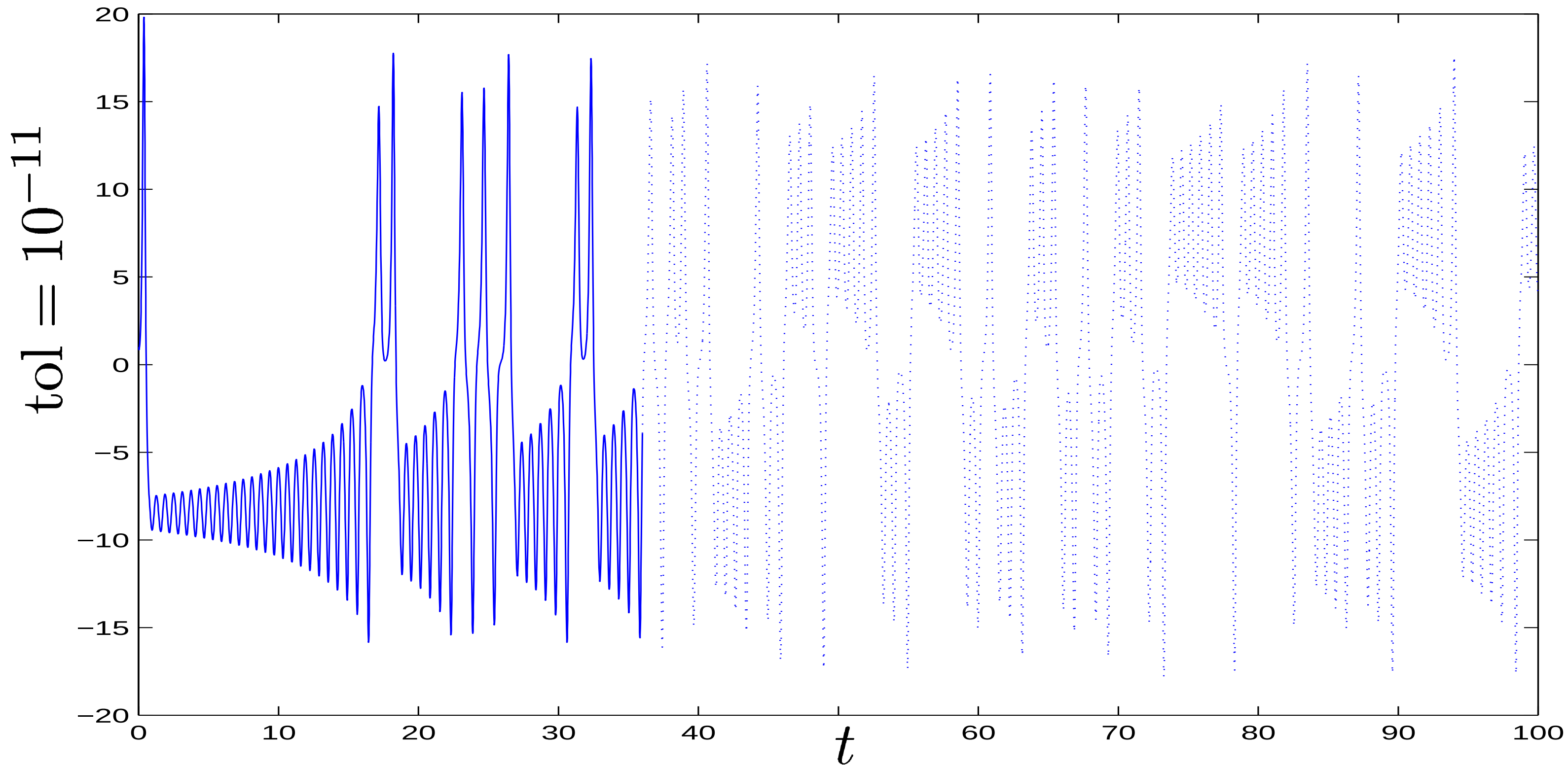
ode45

Trying the same thing with Matlabs ode45, we get the following results:



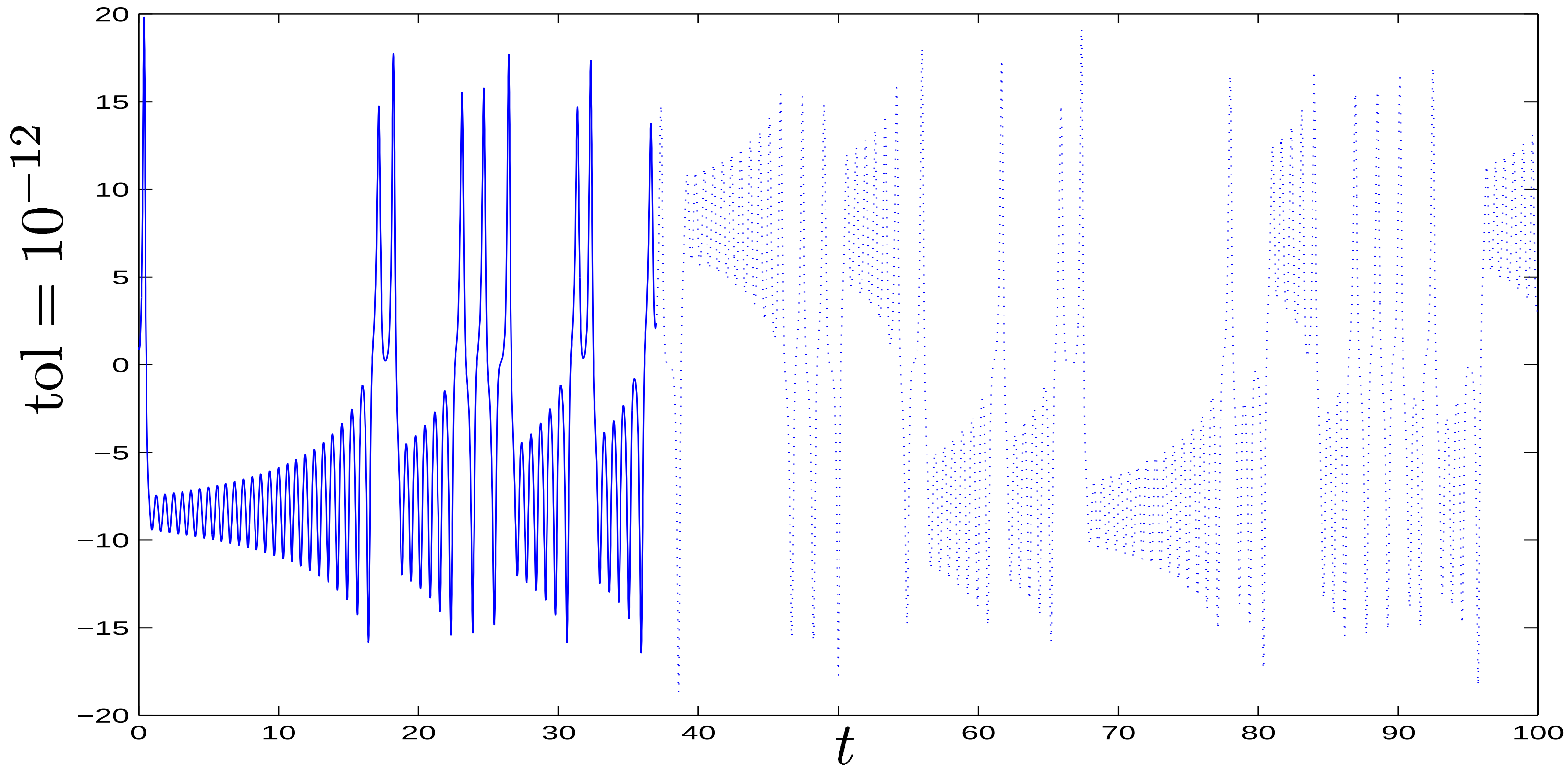
ode45

Trying the same thing with Matlabs `ode45`, we get the following results:



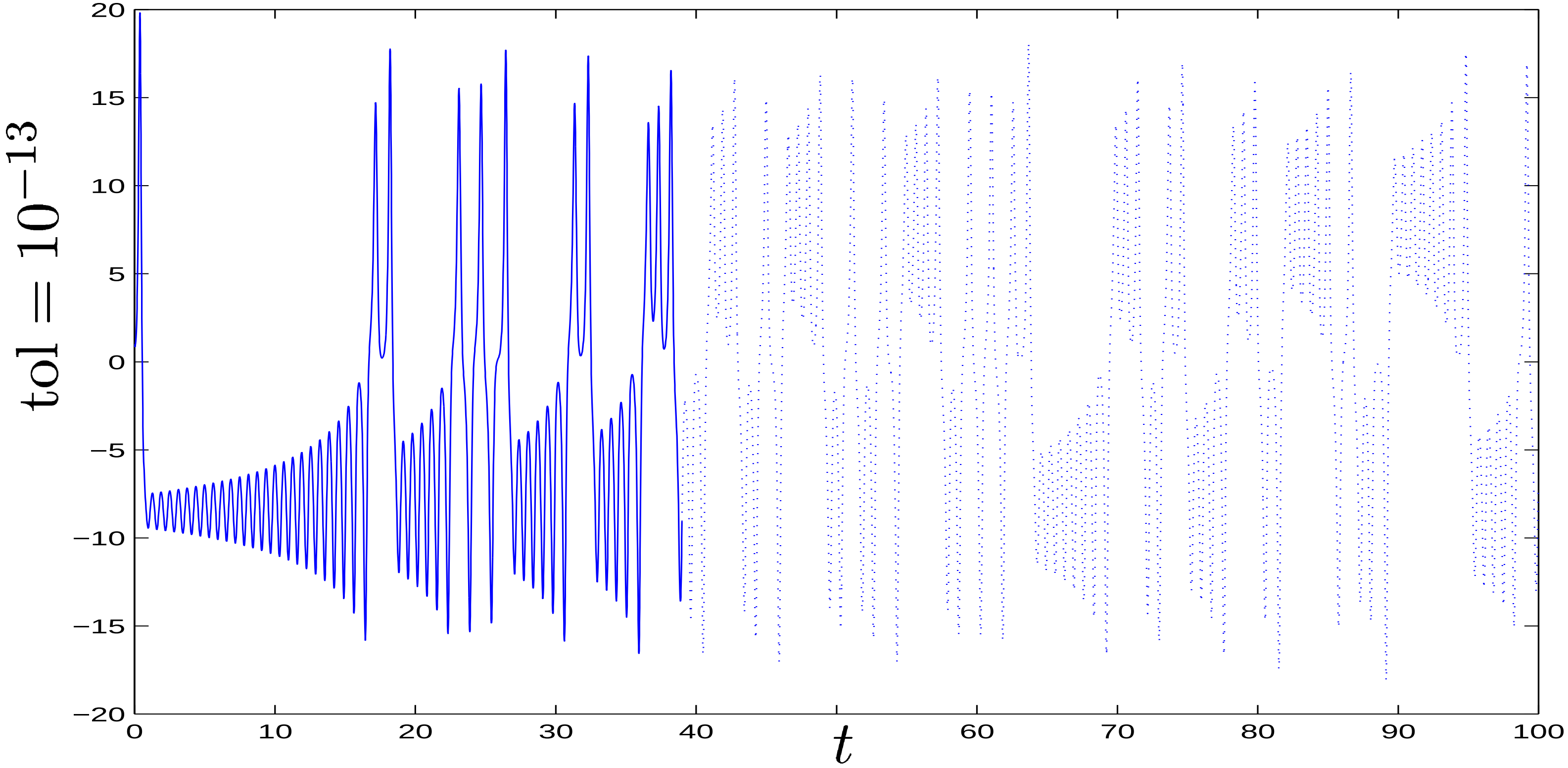
ode45

Trying the same thing with Matlabs `ode45`, we get the following results:



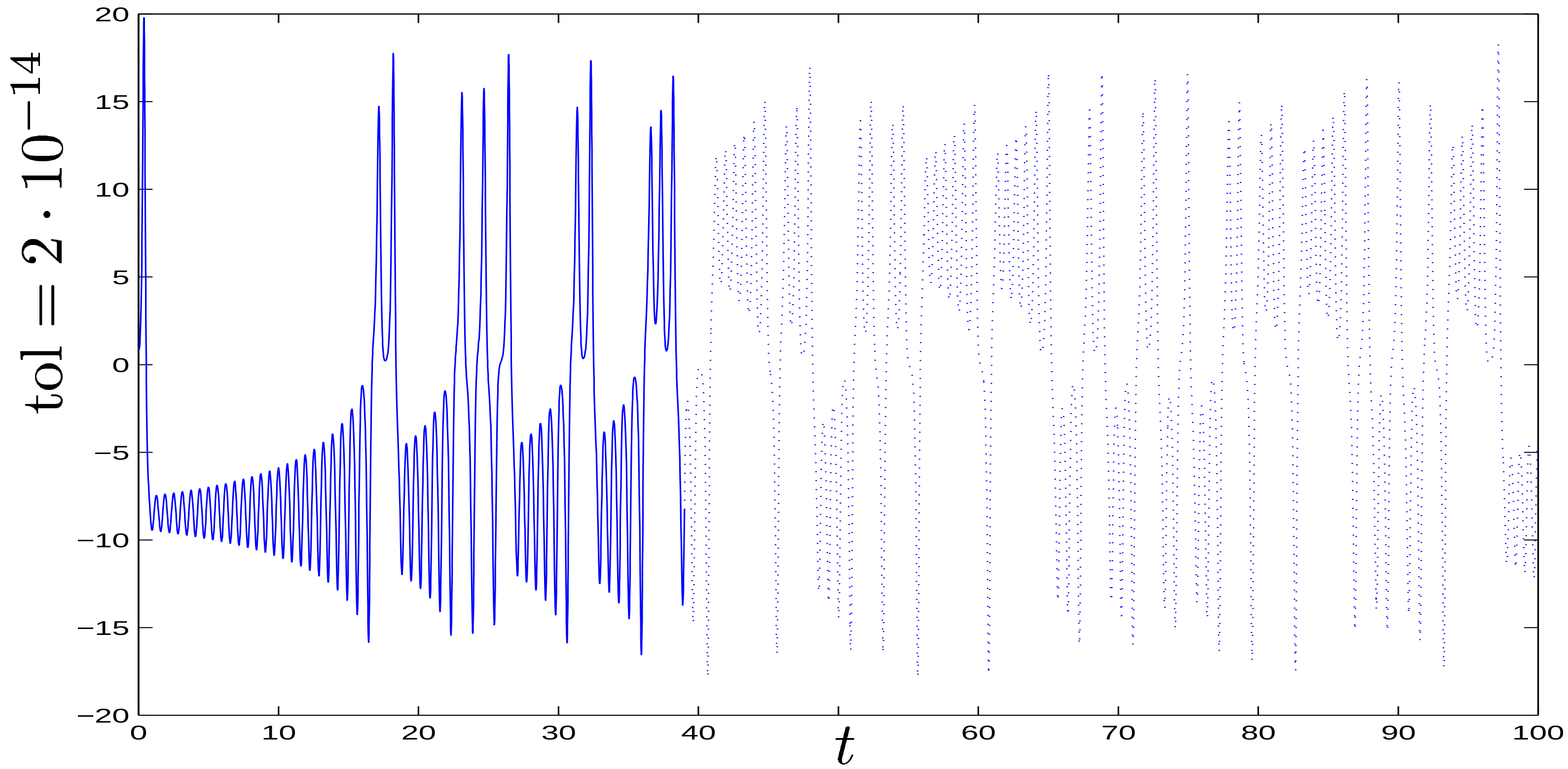
ode45

Trying the same thing with Matlabs `ode45`, we get the following results:



ode45

Trying the same thing with Matlabs `ode45`, we get the following results:



Getting Further

- Small Galerkin error
- Large computational error

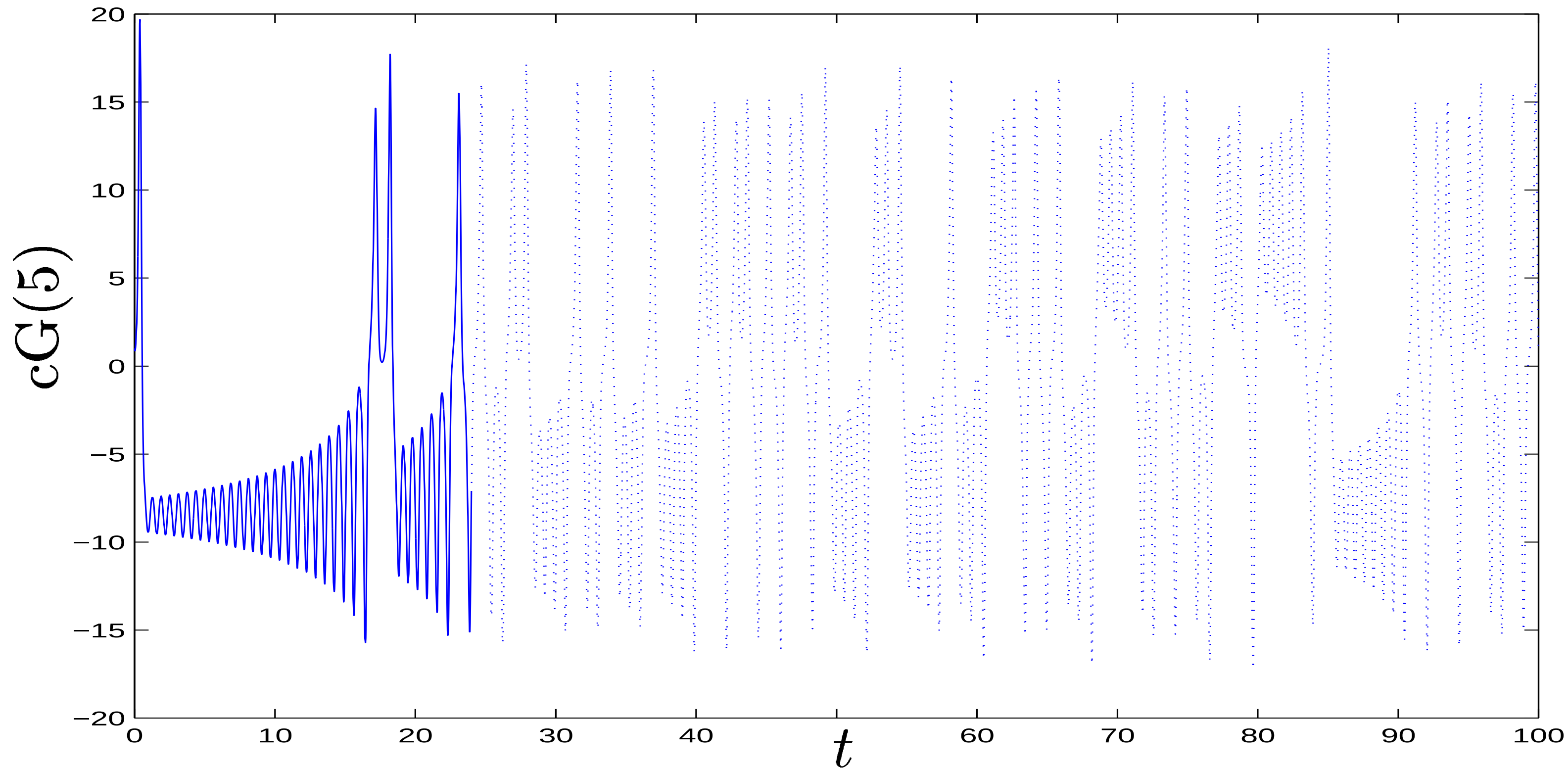
We must thus decrease the computational error, which means decreasing the discrete residual.

$$R^C \approx \frac{10^{-16}}{k} \tag{10}$$

This in turn means that we have to *increase* the timestep!

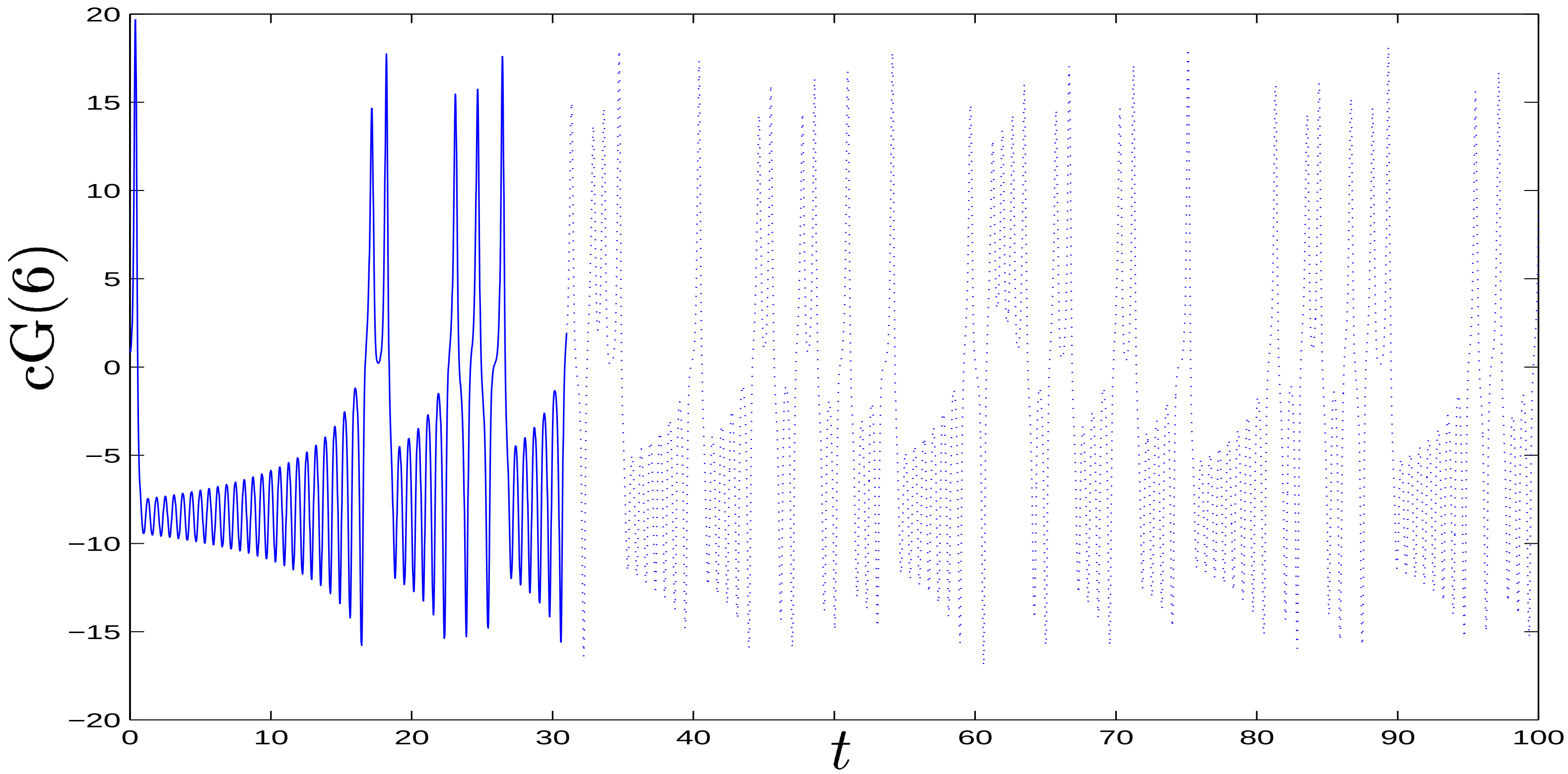
A Simple Experiment — *continued*

Increasing the timestep to $k = 0.1$, we get the following results when increasing the order:



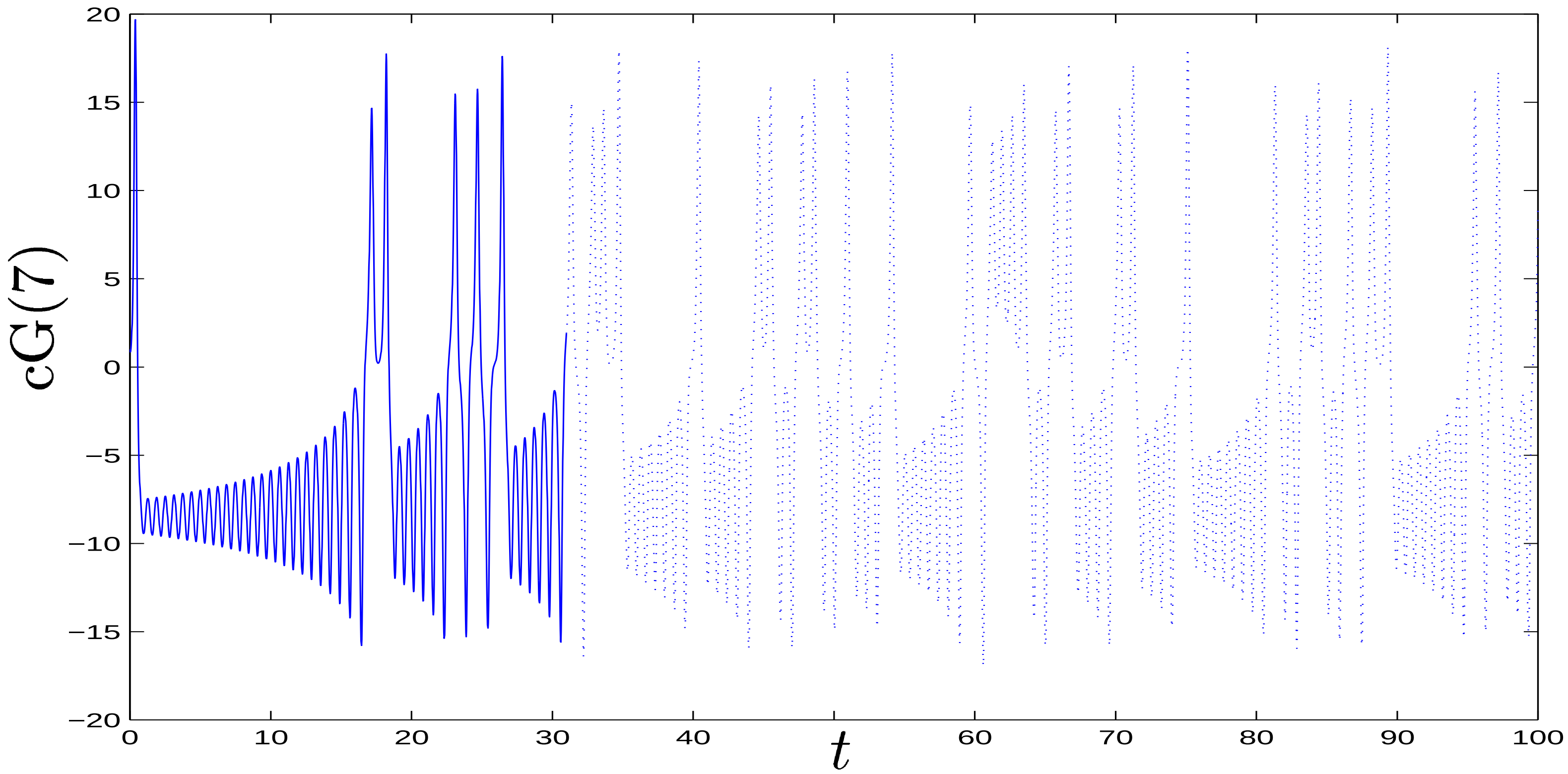
A Simple Experiment — *continued*

Increasing the timestep to $k = 0.1$, we get the following results when increasing the order:



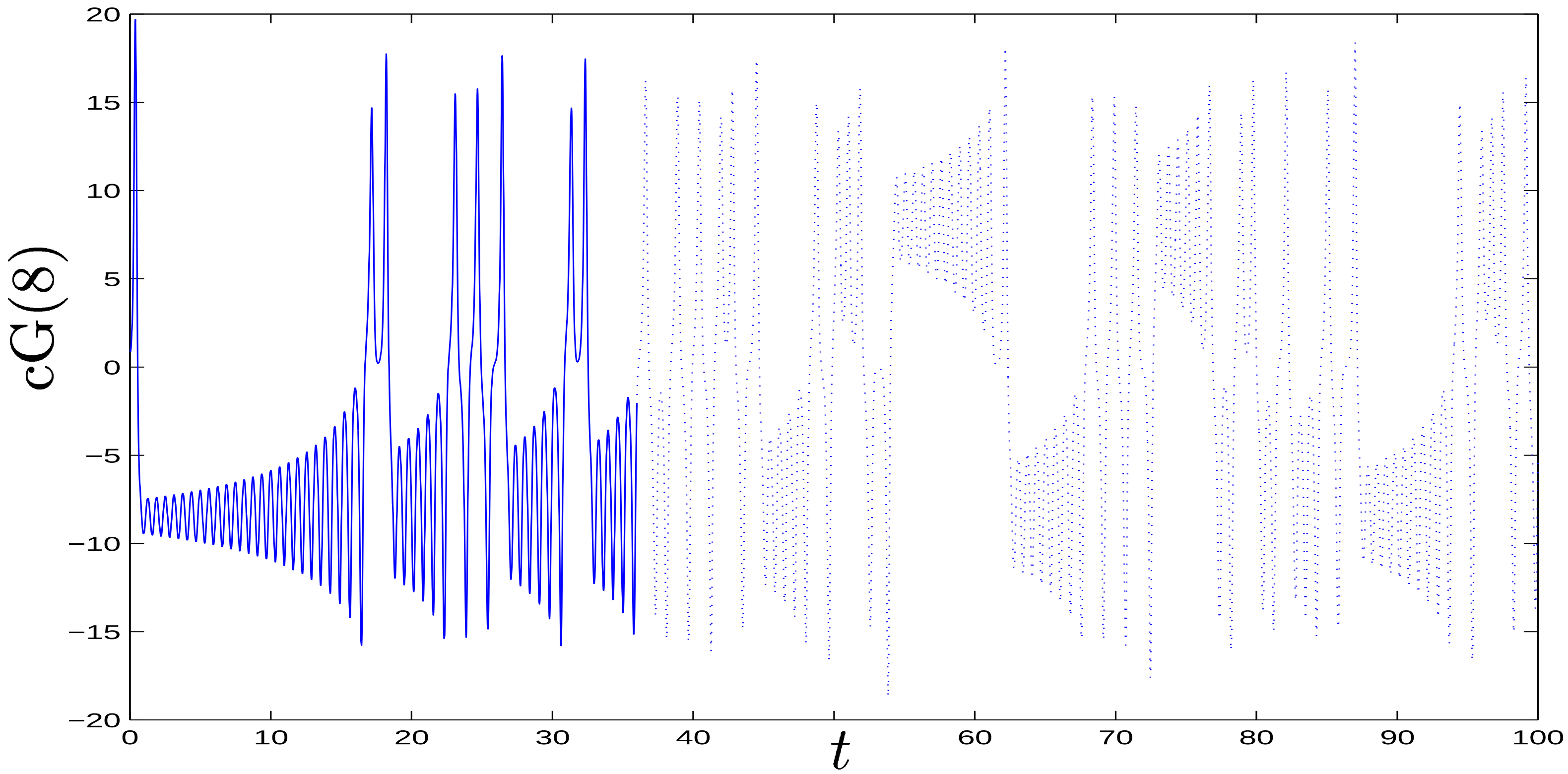
A Simple Experiment — *continued*

Increasing the timestep to $k = 0.1$, we get the following results when increasing the order:



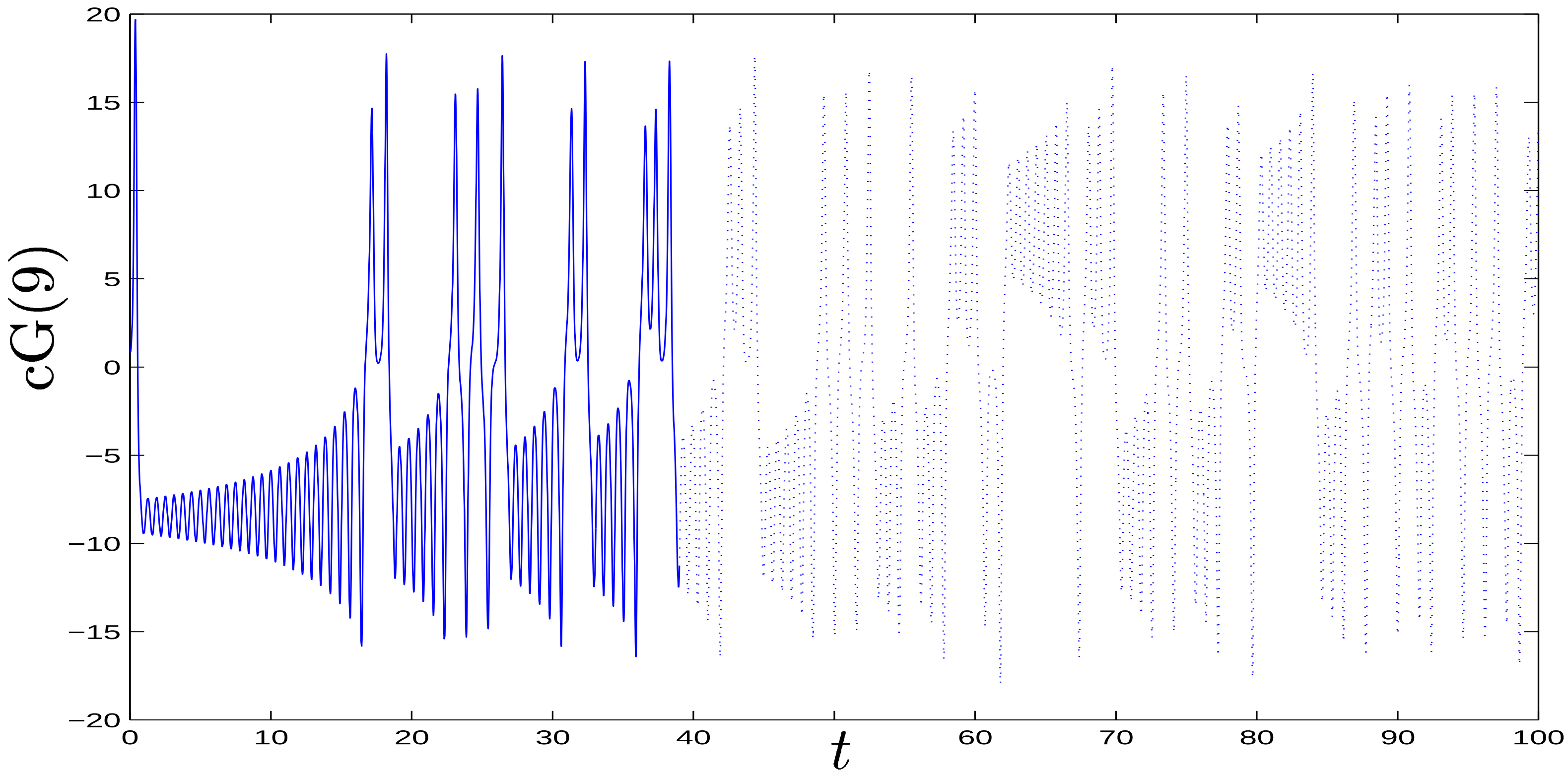
A Simple Experiment — *continued*

Increasing the timestep to $k = 0.1$, we get the following results when increasing the order:



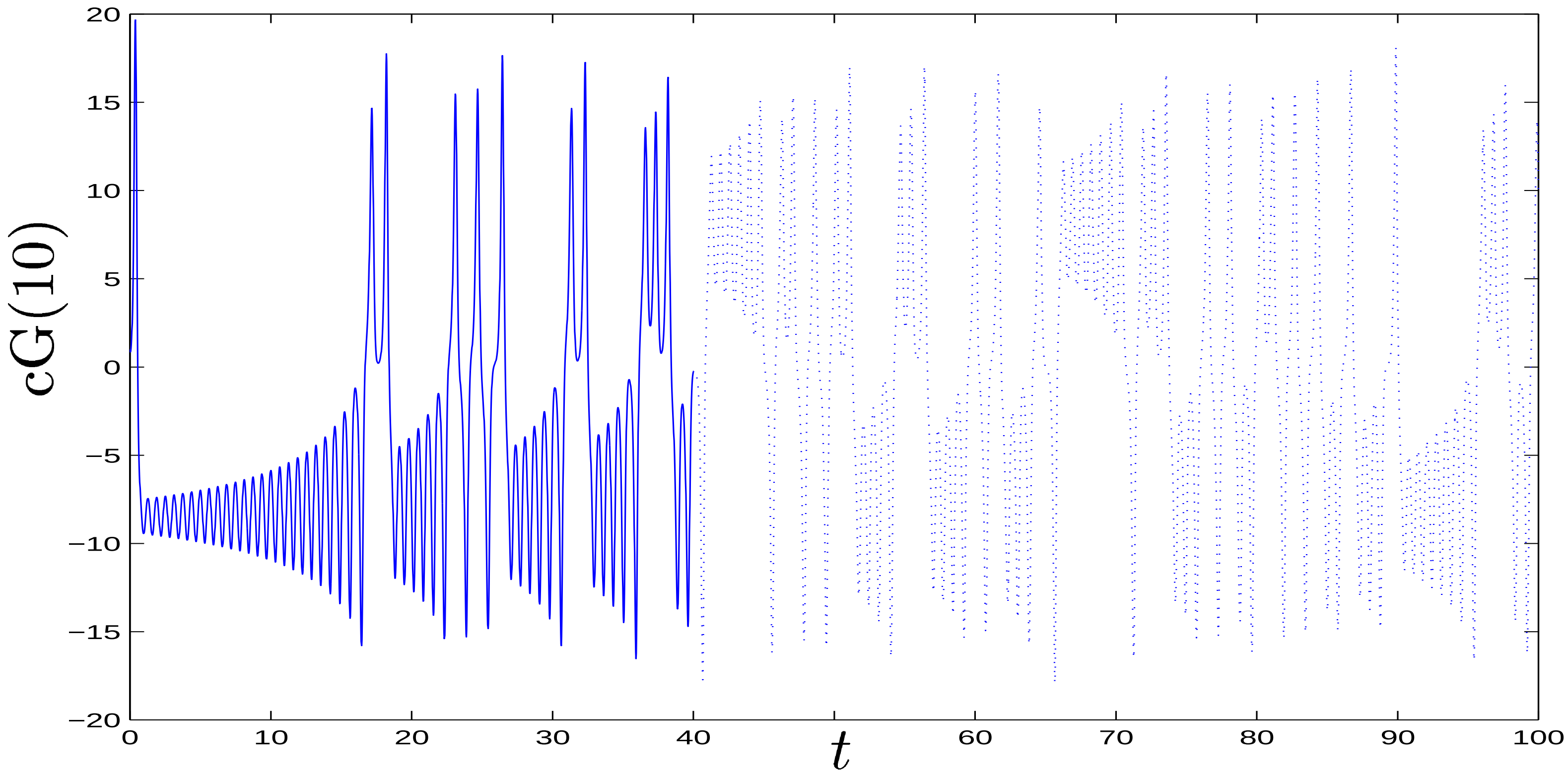
A Simple Experiment — *continued*

Increasing the timestep to $k = 0.1$, we get the following results when increasing the order:



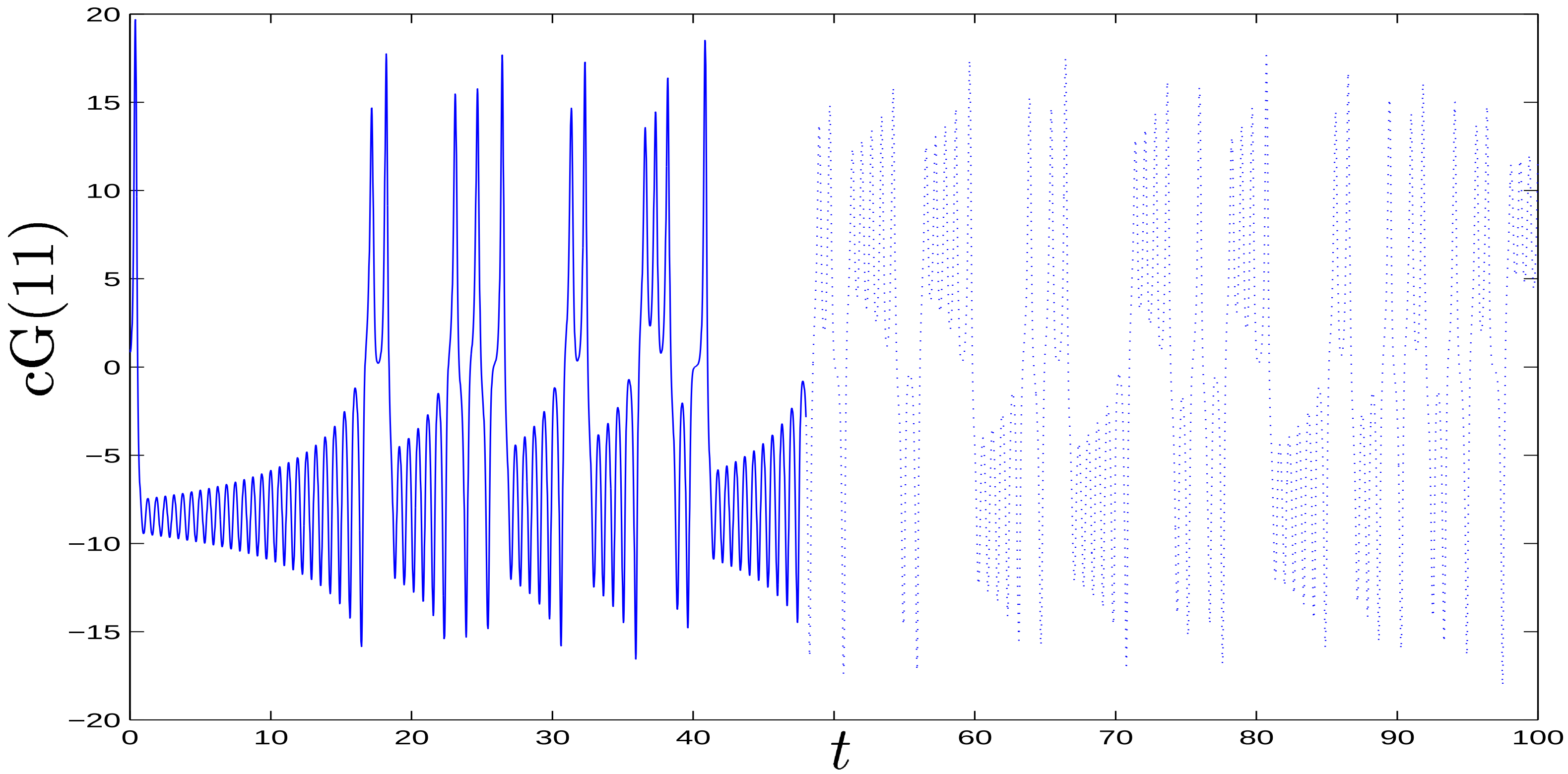
A Simple Experiment — *continued*

Increasing the timestep to $k = 0.1$, we get the following results when increasing the order:



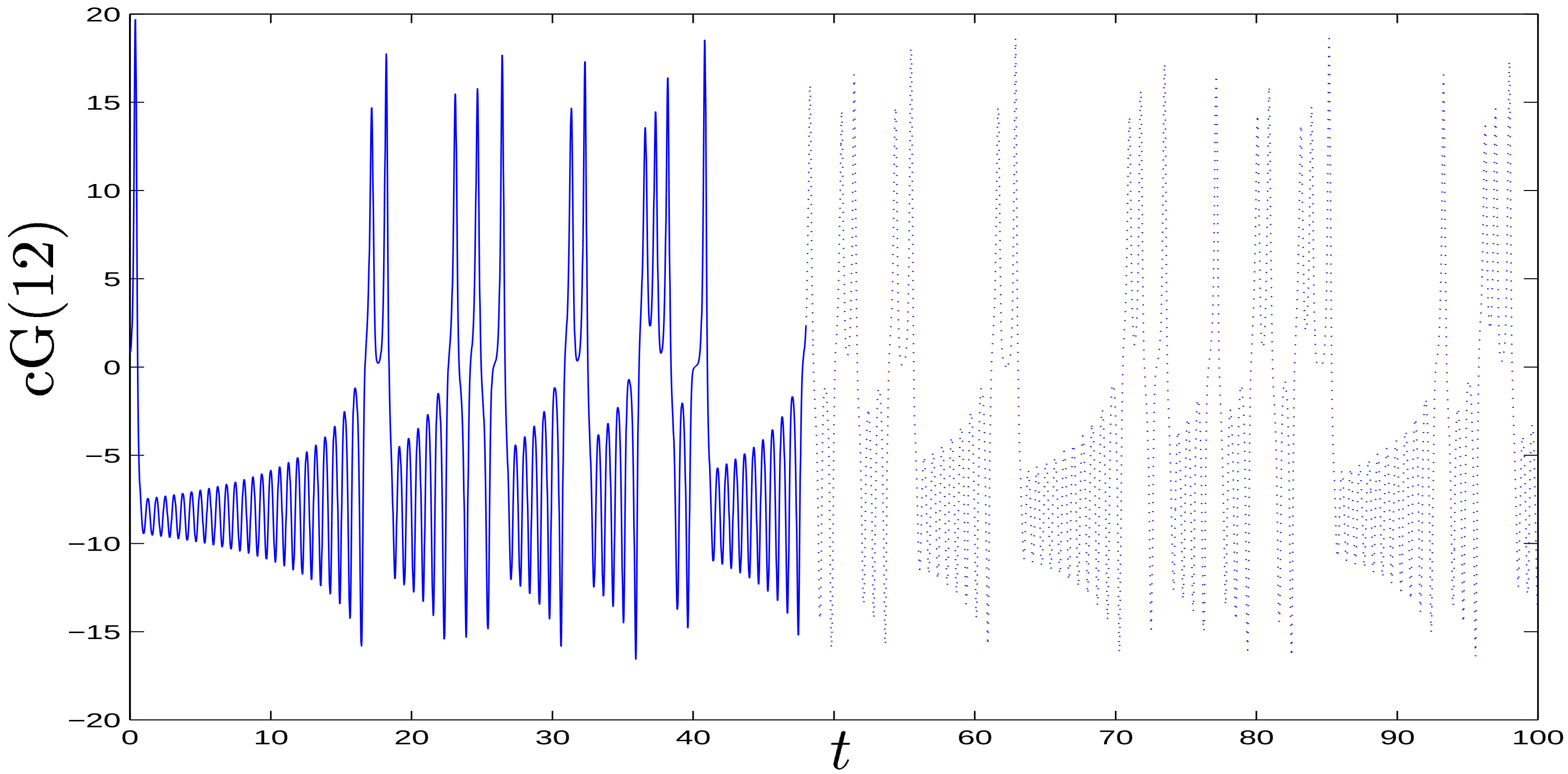
A Simple Experiment — *continued*

Increasing the timestep to $k = 0.1$, we get the following results when increasing the order:



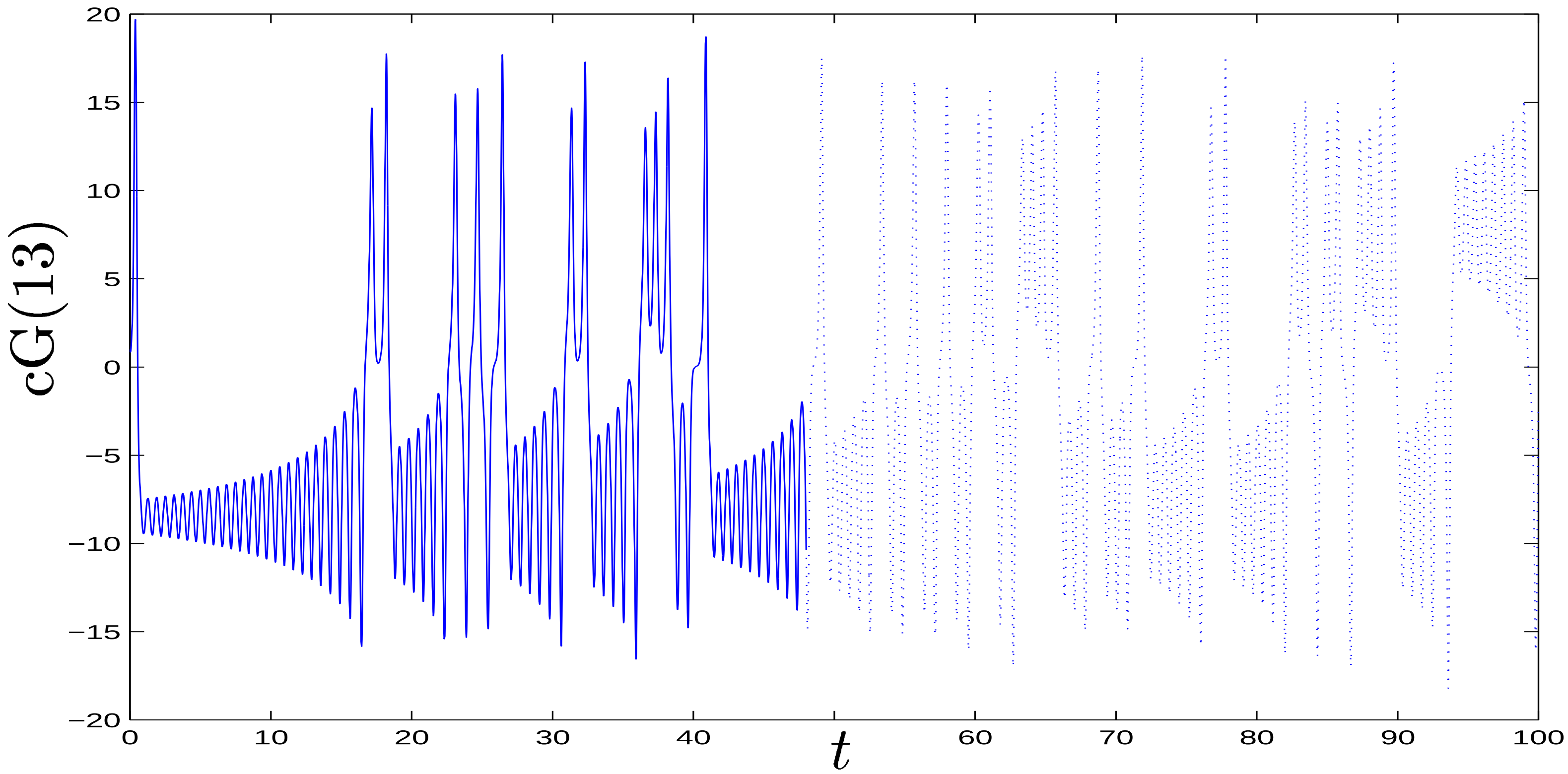
A Simple Experiment — *continued*

Increasing the timestep to $k = 0.1$, we get the following results when increasing the order:



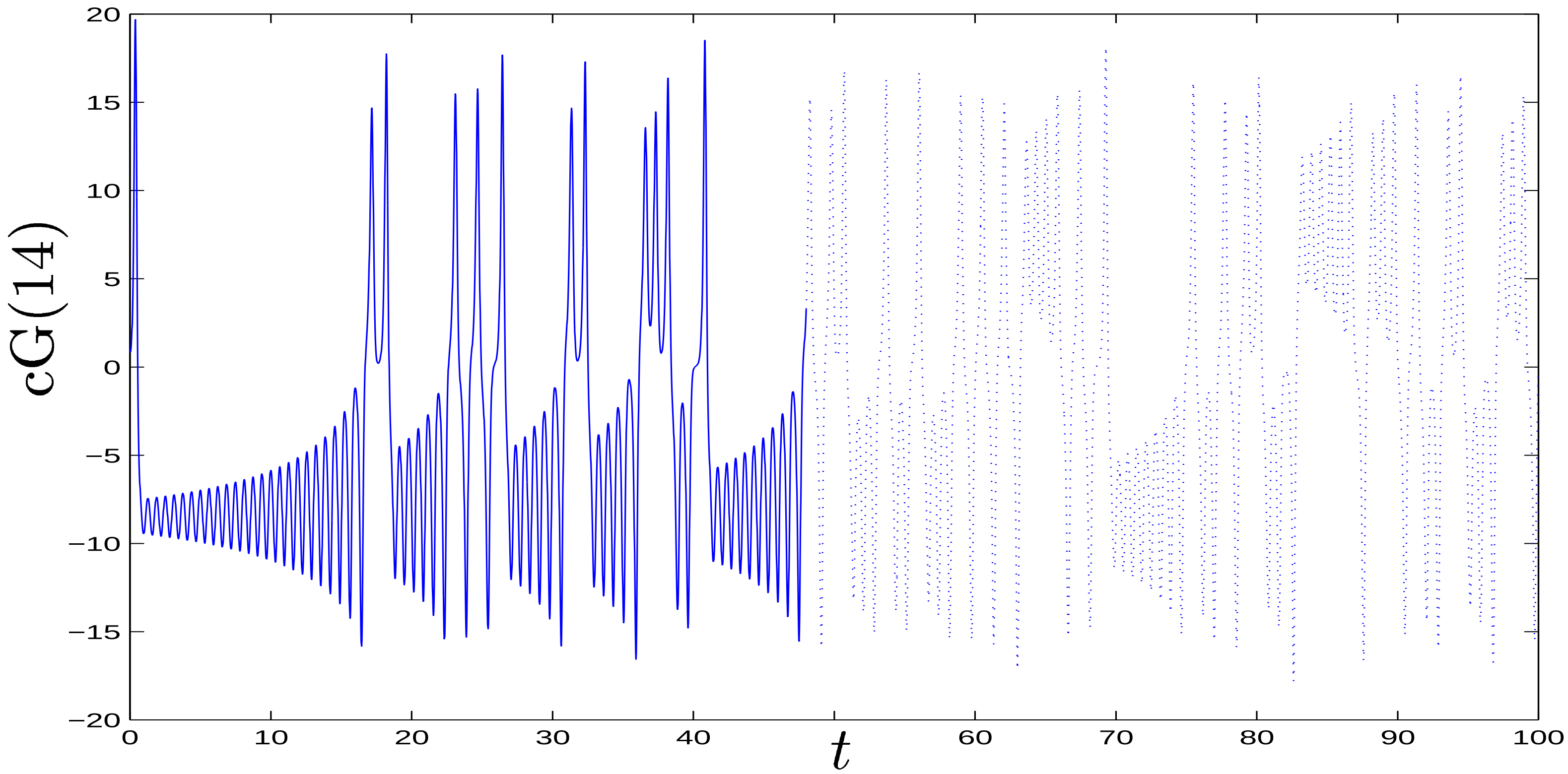
A Simple Experiment — *continued*

Increasing the timestep to $k = 0.1$, we get the following results when increasing the order:



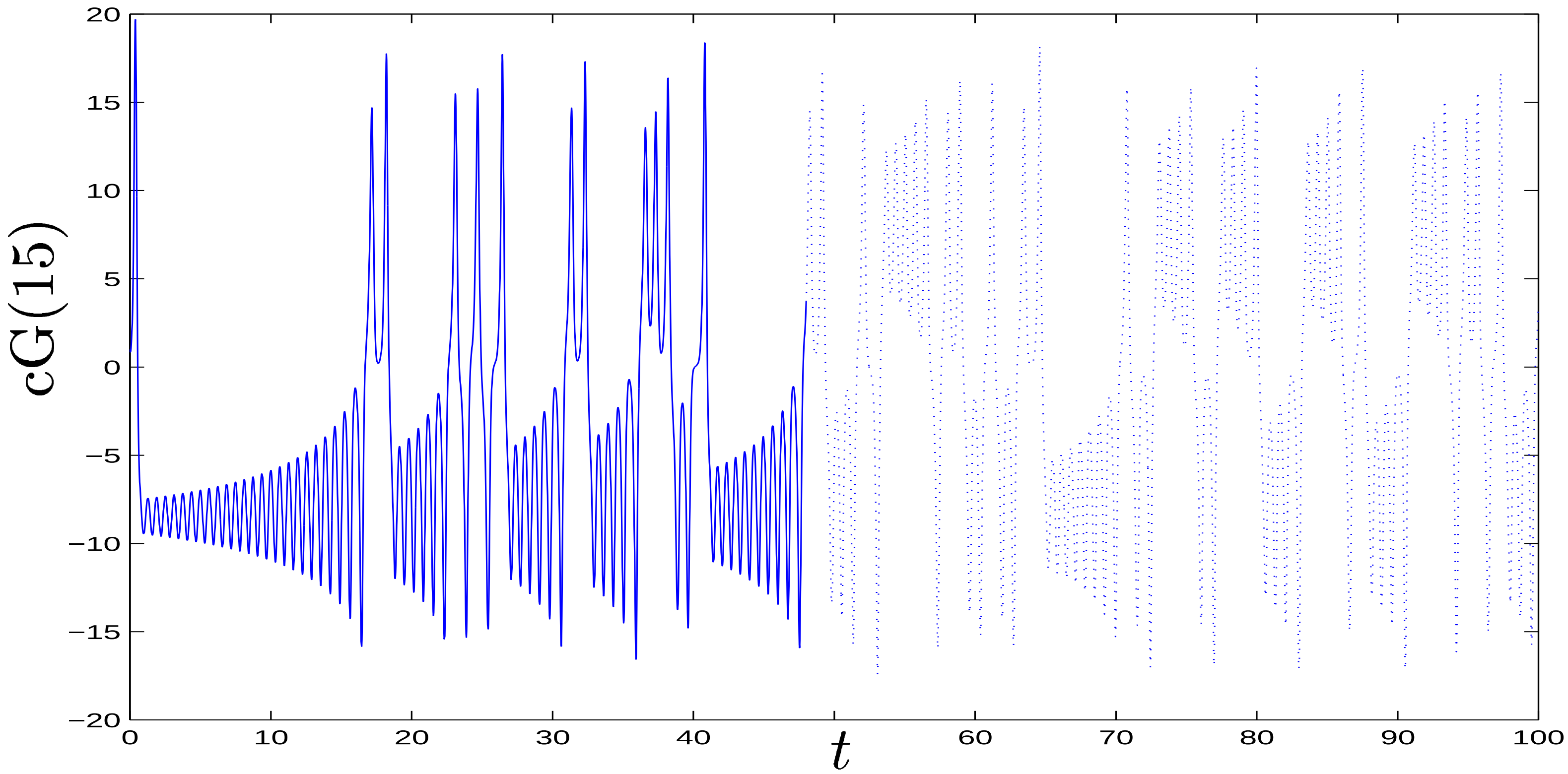
A Simple Experiment — *continued*

Increasing the timestep to $k = 0.1$, we get the following results when increasing the order:

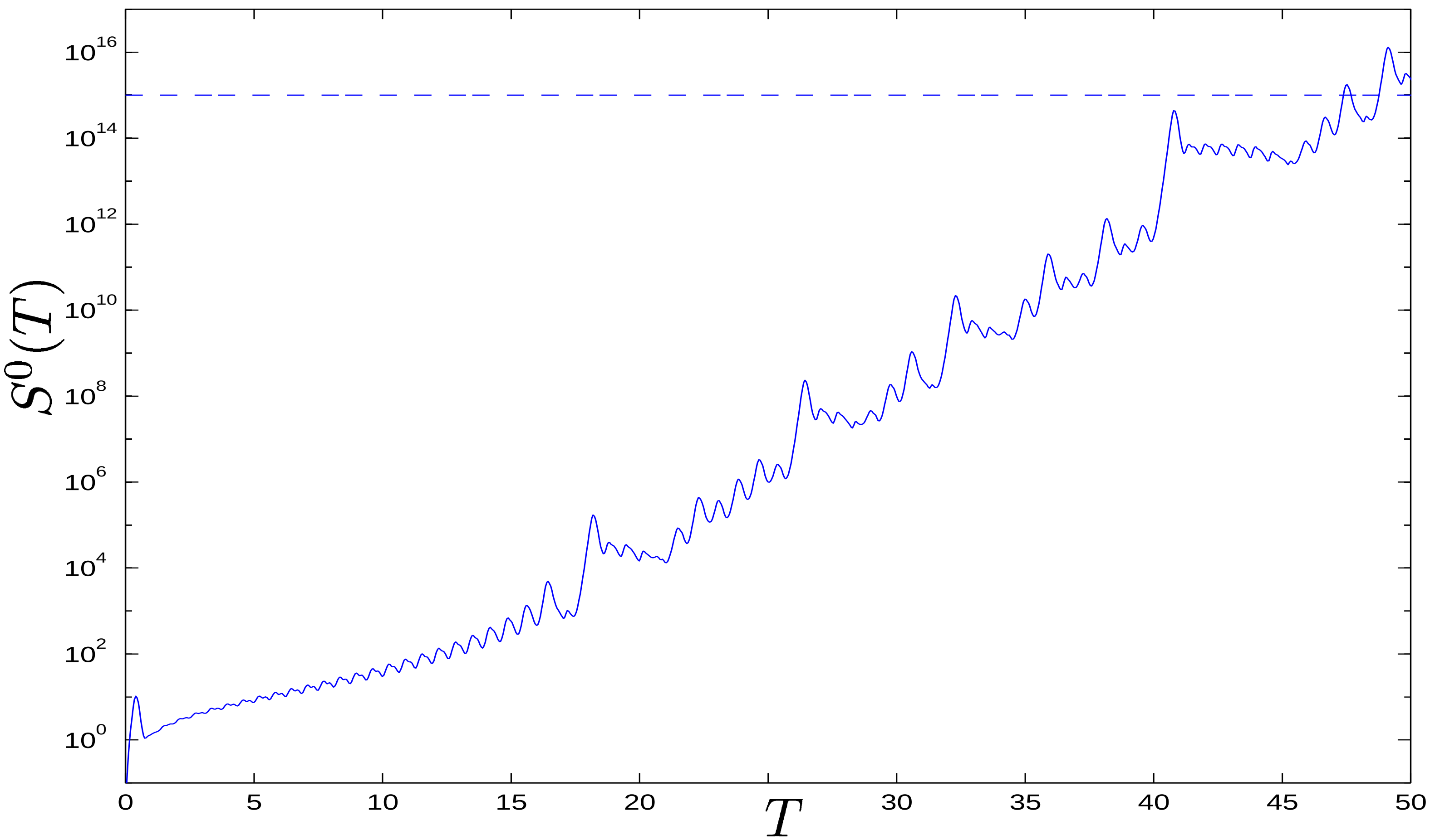


A Simple Experiment — *continued*

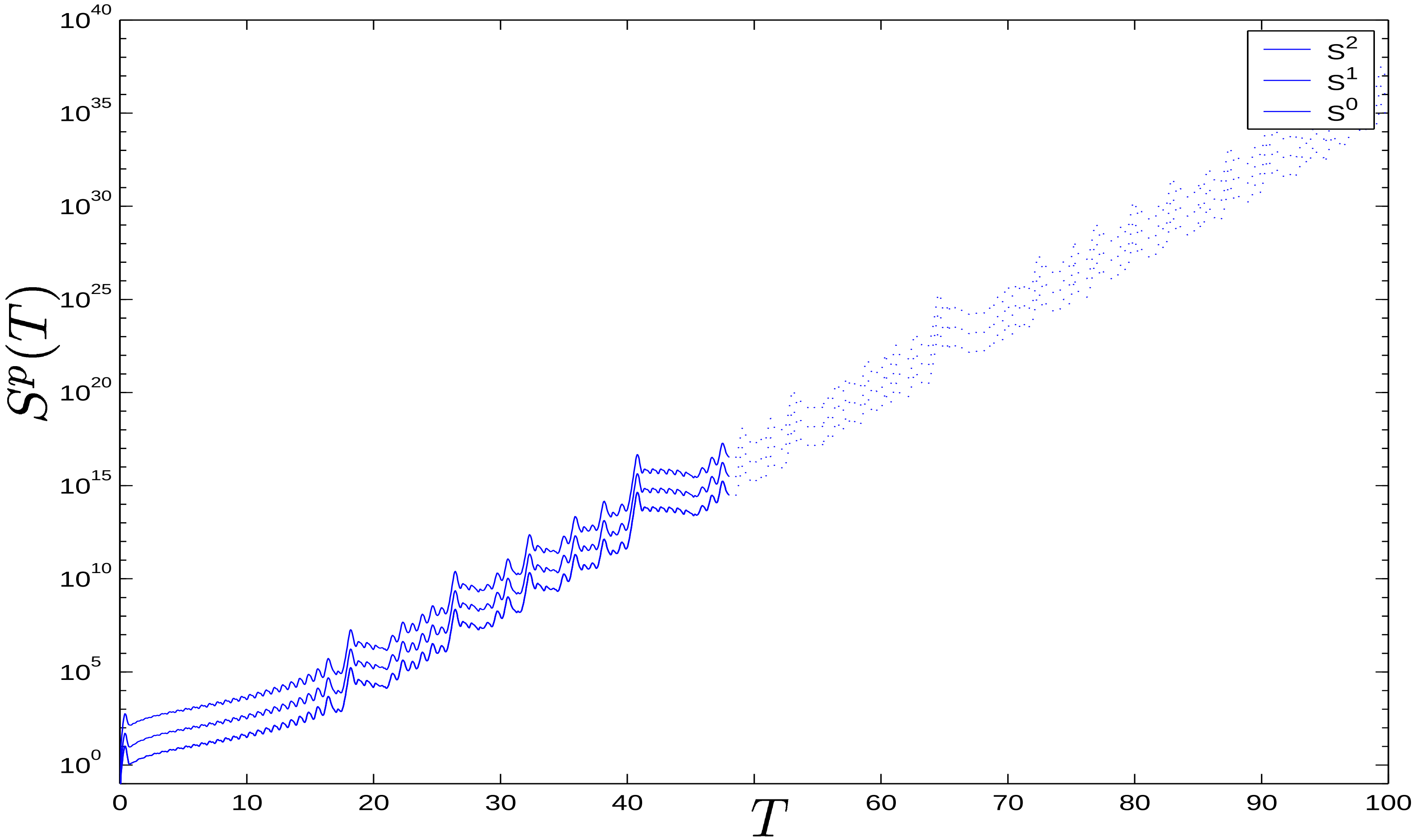
Increasing the timestep to $k = 0.1$, we get the following results when increasing the order:



The Stability Factor



Galerkin Stability Factors



A Simple Estimate

A simple estimate for the stability factors is

$$S^{[q]}(T) \approx 10^{q+T/3}, \tag{10}$$

so that

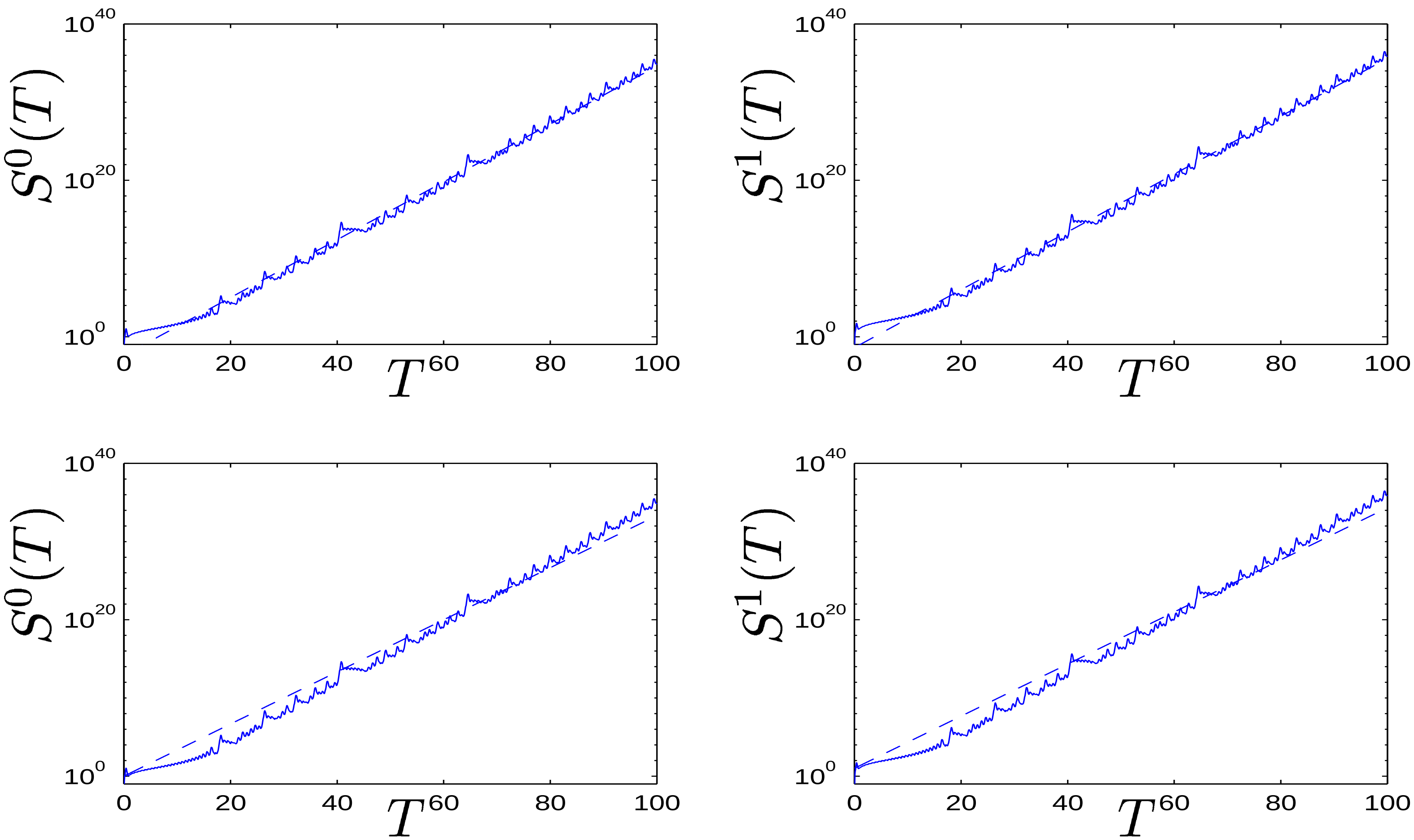
$$E_C(T) \approx \frac{10^{-16}}{k} 10^{T/3} = 10^{T/3-16}/k.$$

How far we get

Keeping $E_C < 1$ we find the maximum value of T:

- $T = 3 \cdot 13 = 39$ with $k = 0.001$,
- $T = 3 \cdot 15 = 45$ with $k = 0.1$,
- not much further with larger timestep, since soon we will have $k > T$.

The Approximation



Conclusions

- There are several sources for the total error.
- Each of these are equally important, in that for a certain application any one of these may dominate.
- From computing the stability factors, we may see clearly which of the sources is dominant for a specific problem and tune the implementation accordingly.

Chalmers Finite Element Center

Visit our homepage!

`www.phi.chalmers.se`

- `www.phi.chalmers.se/preprints`
- `www.phi.chalmers.se/presentations`