# Rough notes about the exponential of an SVF
# Results

Some notes on the numerical computation of the exponential of a stationary velocity field.

UCL, TIG, for internal use, November 25, 2015, s.ferraris@ucl.ac.uk

## Contents

ref: (3)

# 1 $se(2)$-generated SVF results

A first comparison between the methods for the SE(2)-generated vector field with the same number of steps for each method (10) results in the figure 1.

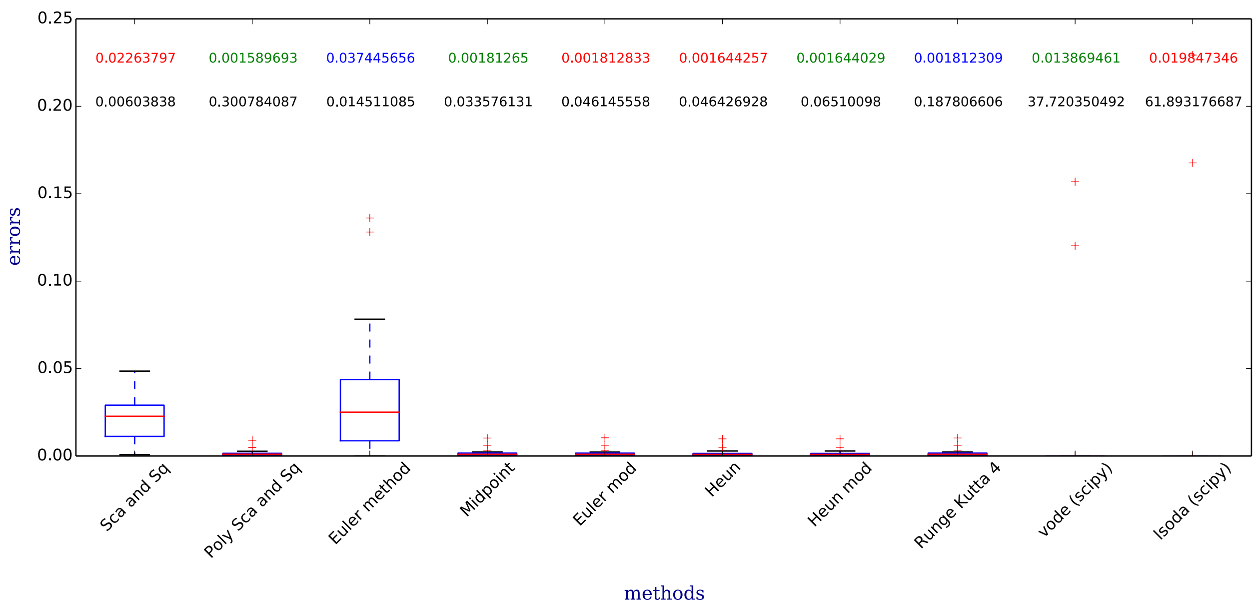


Figure 1: A data-set of 20 $se(2)$-generated SVF are sampled wth the rotation angle $\theta$ between $-\pi/4$ and $\pi/4$ excluding an interval centerd on the zero of radius 0.01. First row of coloured values in the foreground of the boxplot are teh mean of the resulting errors, values in black in the secon row are the mean of the computational time in seconds.

## 1.1 Number of iterations V.S. errors, some empirical considerations

If we consider the flow-integrator methods scaling and squaring, polyaffine scaling and squaring, Euler, midpoint, Euler modified, Heun, Heun modified and Runge Kutta 4 for a given SE(2)-generated SVF, where the ground truth of the exponential is available, we have the relation between the number of steps of each method and the error given in figure 2. As expected, the polyaffine scaling and squaring reach the solution in 1 step (the ODE is linear). Scaling and squaring is faster than Euler, but any other methods gets better than them (remarkably the RK4 behave really well for the linear case; Heun modified is comparable to RK4 after 2 steps).

The vertical dashed line represent the number of iteration automatically computed by the method presented in section **??**. This shows that the automatic computation can be improved with a +2, or settings the threshold length equals to the size of 1/4 of a voxel.

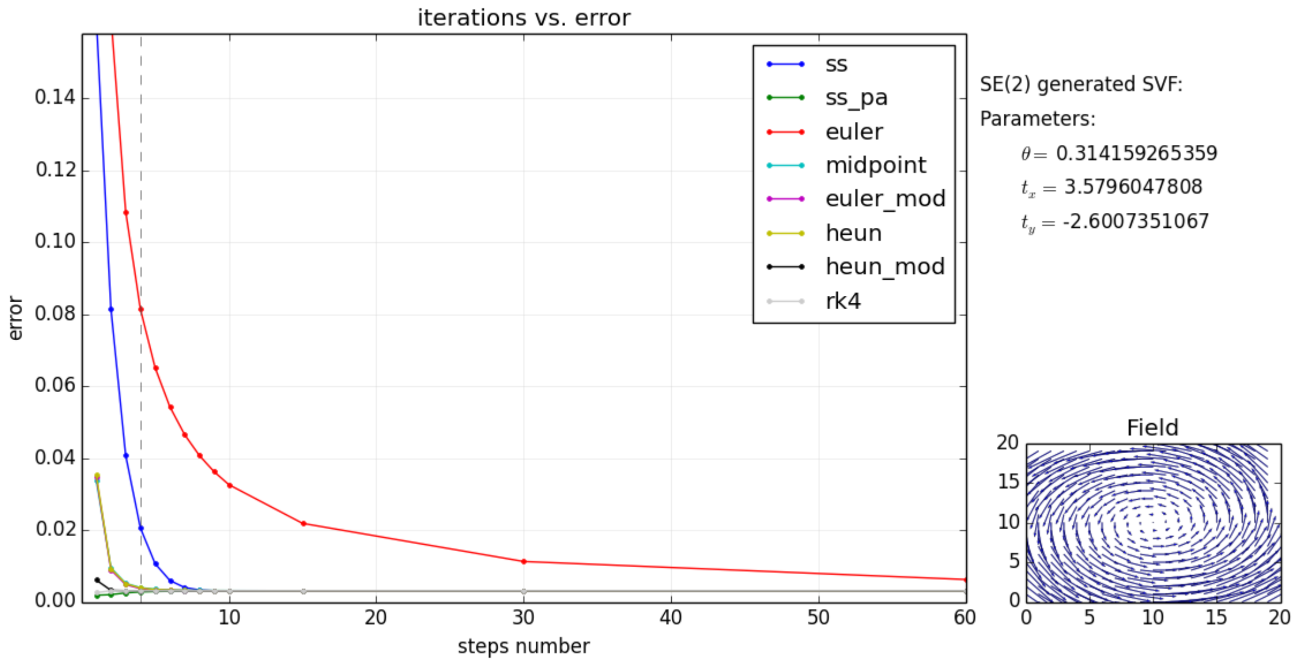


Figure 2: .

For the non-linear case, (Gauss-generated SVF) we do not have a ground truth. According to the results obtained in the linear case we selected RK4 as a *fake ground truth*. Result are shown in figure 3. Repeatedly sampling shows consistency in the results for the chosen parameters.

Heun modified is based on a similar concept than the RK4, so no surprise if it is the one with smallest error. The polyaffine scaling and squaring behave well with this fake ground truth.

These 2 experiments where useful to have some information to chose an ideal number of steps for each method:

1. Number of steps manually tuned for the linear case for each method: $[7, 1, 40, 10, 10, 10, 10, 1]$ (same order as in the legend).
2. Number of steps manually tuned for the non linear case for each method: $[8, 8, 40, 10, 10, 10, 10, 8]$.

With this choice of parameters we can repeat the experiments for a bigger data samples for both the linear and the non linear cases. Results are shown in figures 4 and 5.

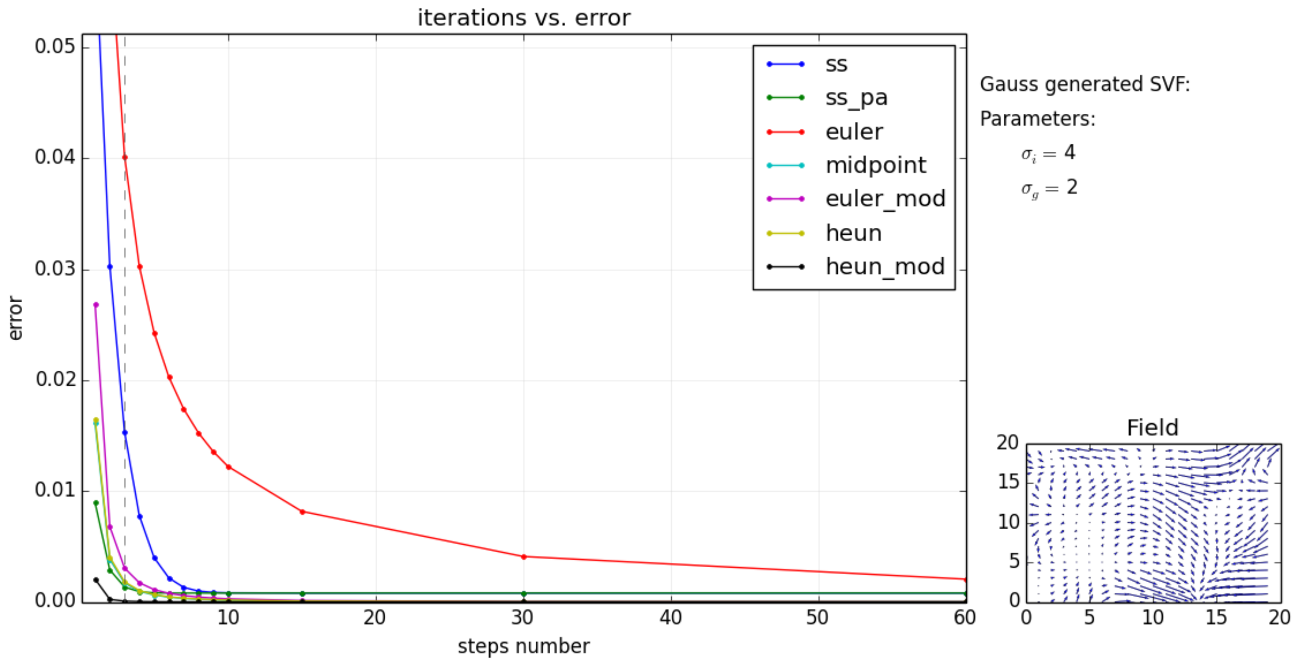


Figure 3: .

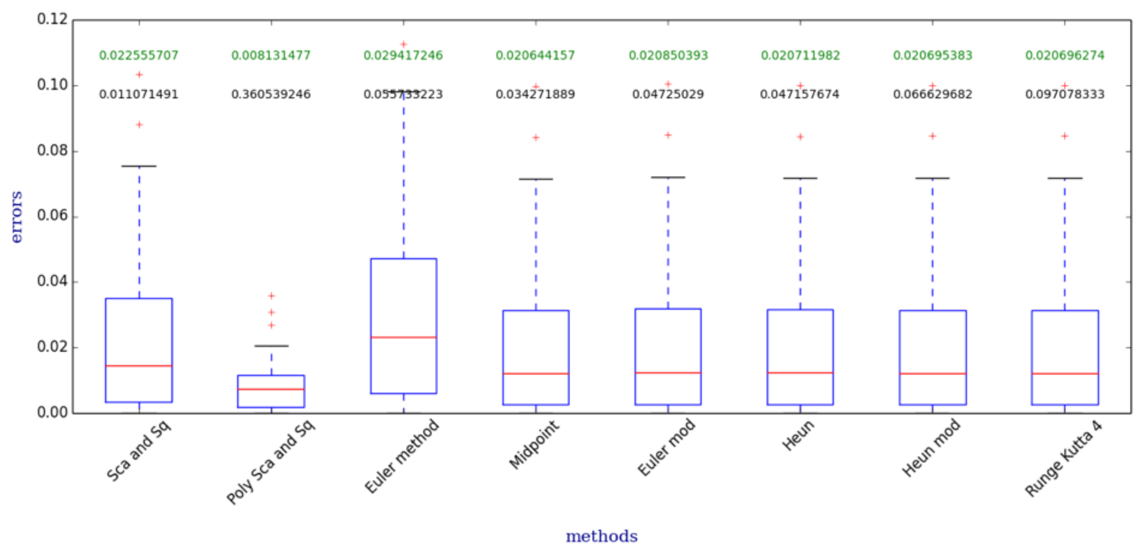


Figure 4: .

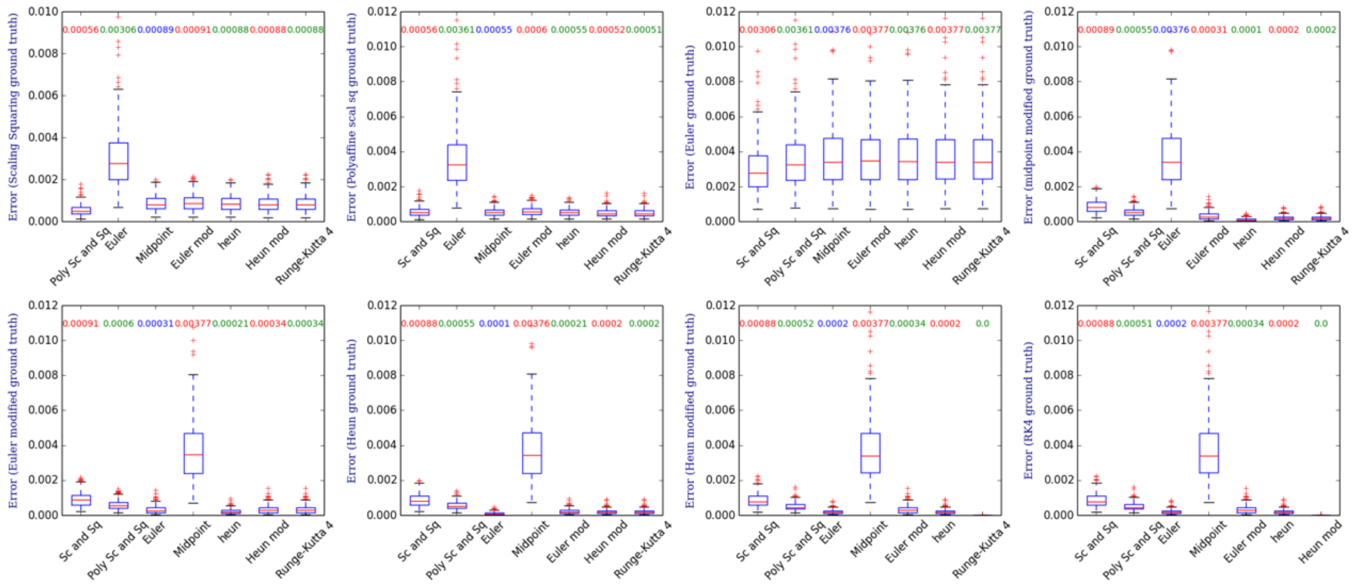


Figure 5: .

## 1.2 Stability through the step-error, some empirical considerations

When $\exp(\mathbf{v})$ is computed with a numerical integrator $\psi_1$ with step size $h$ and number of steps $N = 1/h$, its *step-error* at each iteration $k$ is defined as

$$\text{err}(\mathbf{v}, k) = |\psi_{k+1} - \psi_k|$$

We note that it depends only on the integrator method and on the previous steps. It does not need any ground truth (and we like it!).

In addition the sequence $\{\text{err}(\mathbf{v}, k)\}_{k=1}^{N}$ provides an measure of the stability of the method.

For a given fixed SE(2)-generated SVF the step-error is given in the figure 6. No too curiously for more than approximatively 60 steps, the scaling and squaring method and its polyaffine version goes overflow (too small number). For this reason the blue and the green lines stops at 60 steps, after a scary fall to the ground.

Euler is very smooth, as up to some point Euler modified, midpoint and Heun methods. Heun modified and RK4 are more unstable but the general trend shows a reduction in the error.

For more than one SE(2)-generated SVF results are shown in figure 7.

Here are random SE2-generated SVF (please note that the different colour of the line does not means different method, but different sampling from random generated SVF):

Results for Gauss-generated SVF can be seen in figure 8.

Lastly, for ADNII generated SVF the results are shown in figure 9.

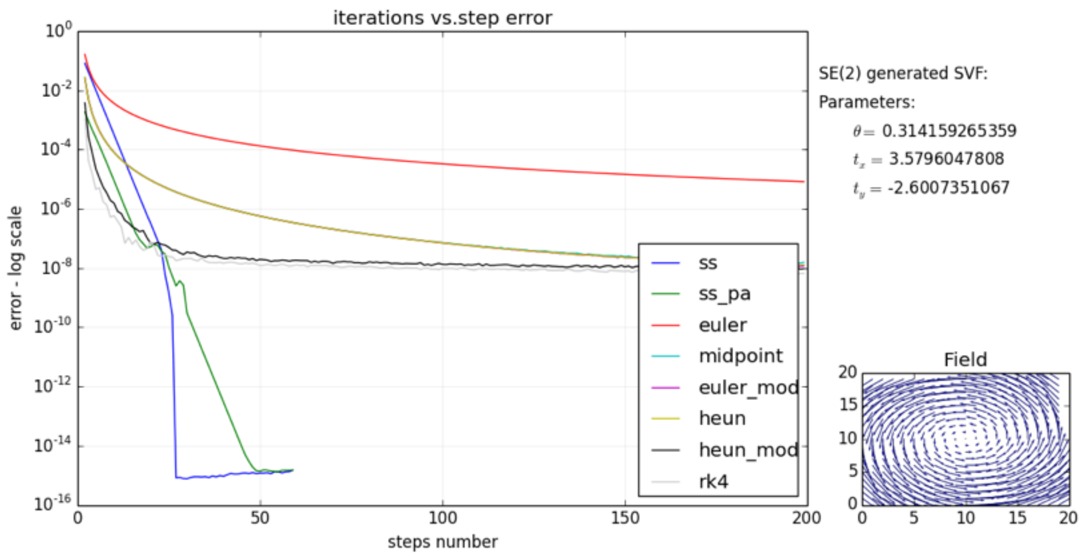


Figure 6: .

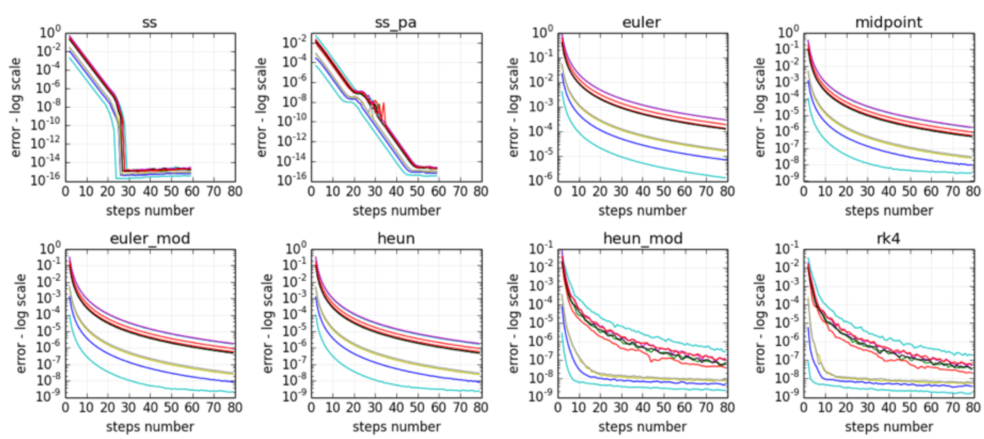


Figure 7: .

### 1.3 On the inverse consistency of the methods

TODO next

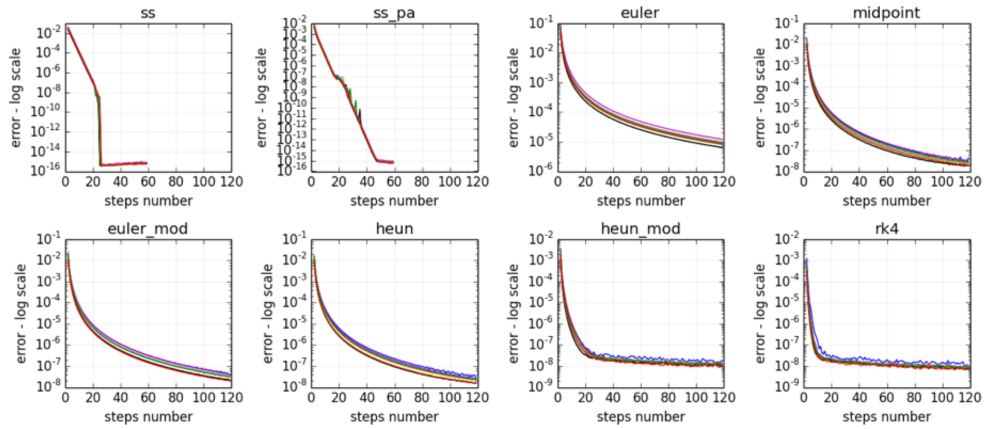


Figure 8: .

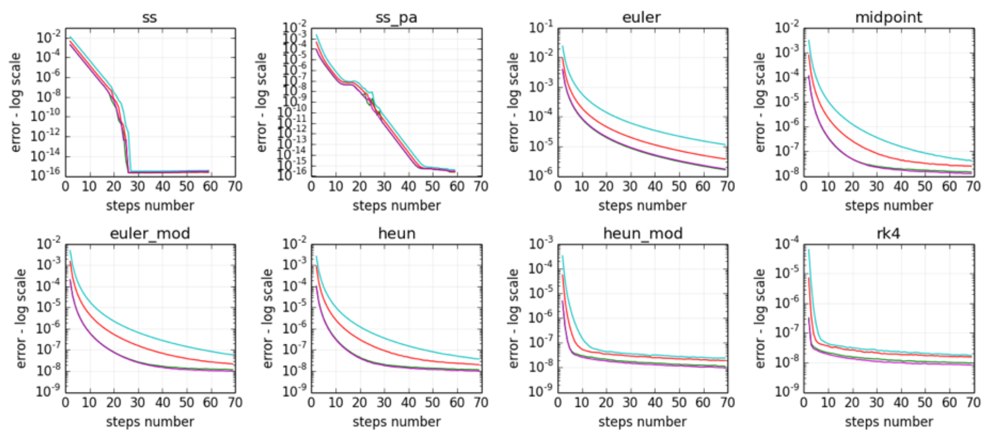


Figure 9: .

# References

[1] Nicholas J Higham and Lin Lijing. Matrix functions: A short course. 2013.

[2] Marlis Hochbruck, Christian Lubich, and Hubert Selhofer. Exponential integrators for large systems of differential equations. *SIAM Journal on Scientific Computing*, 19(5):1552–1574, 1998.