

CAS 741: Problem Statement

rkf45.ml

Alexander Schaap `schaapal`

September 15, 2017

Table 1: Revision History

Date	Developer(s)	Change
Sept 15	Alex	Initial version

Scientists in many areas use ordinary differential equations (ODEs) in their research. While many solvers exist already, a new programming paradigm may provide performance benefits over existing approaches.

Multi-stage is a metaprogramming paradigm in which type-safe code fragments are created and combined. Its goal is enable development of “generic software that does not pay a runtime penalty for this generality”¹. (Note that this heavily implies a compile-time penalty.) Multi-stage programming allows for potentially every bit of information known at compile-time to be optimized out of the generated code. Doing this can lead to significant performance benefits, which ultimately allow users of programs constructed in this way to focus more on their other activities.

A popular method for numerical approximation of ODEs is the Runge-Kutta-Fehlberg (RKF45) method. (The 4 in the abbreviation comes from the method being of order $O(h^4)$ and the 5 signifies the error estimator of order $O(h^5)$). This method has been implemented in a multi-stage programming language called MetaOCaml by Dr. J. Carette. The goal therefore would be to recreate and improve this implementation.

One might argue that manually optimizing for performance is a thing of the past. While it is true that computer hardware has become much faster, there are still computationally intensive problems as well as a large portion of the world population that cannot afford the latest and greatest hardware. This could be motivated by financial reasons as well as a need for the hardware to be mobile (e.g. low-voltage processors in laptops).

Primary stakeholders would be scientists using this implementation either directly or indirectly, as well as programmers integrating this implementation into larger pieces of software.

The environment of the software is limited to what the OCaml compiler supports, but this includes Windows, macOS and Linux.

¹<https://web.archive.org/web/20170804032218/http://www.cs.rice.edu/~taha/publications/journal/dspg04a.pdf>