Commonality Analysis / Software Requirements Specification for a Family of ODE Solvers (LODES)

Paul Aoanan

October 2, 2017

Contents

1 Reference Material

This section records information for easy reference.

1.1 Table of Units

This section is not relevant to this CA/SRS.

1.2 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units. The choice of symbols was made to be consistent with the numeral analysis and ordinary differential equation literature and with existing documentation for solving ordinary differential equations. The symbols are listed in alphabetical order.

symbol	unit	description
dy/dx	-	rate of change of y depending on x
$f(x, y)$	-	ODE function containing (x, y)
h	-	step-size from $x_{(0)}$ to the next point $x_{(1)}$, where $x_{(1)} = x_{(0)} + h$
n	-	reference recursion step.
x_0	-	Initial value x
x_{k}	-	Final value x
x_{n}	-	Intermediate n^{th} value x
y_0	-	Initial value y
y_{k}	-	Final value y
y_{n}	-	Intermediate n^{th} value y
y'	-	first order ODE $= f(x, y)$
$y^{(\mathrm{n})}$	-	ODE to the n^{th} order

Table 1: **Revision History**

Date	Version	Notes
October 2, 2017	1.0	First draft.

1.3 Abbreviations and Acronyms

symbol	description
A	Assumption
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
ODE	Ordinary Differential Equation
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
LODES	Family of ODE Solvers
T	Theoretical Model

[Add any other abbreviations or acronyms that you add —SS]

2 Introduction

In physical sciences, mathematical models are derived from scientific models to represent a real world phenomenon through formal mathematical constructs.

Scientific models in the study of radioactivity, carbon decay, and Newton's Law of Cooling involve the use of ordinary differential equations (ODEs).

Known elementary techniques of solving ODEs in the discrete domain use the linear approximation method wherein the solution is based upon assuming or "approximating" the slope of the tangent line from one reference point to the next until the target point has been reached.

The following section provides an overview of the Commonality Analysis (CA) for a program family of ODE solvers. The developed program will be called Library of ODE Solvers (LODES). This section explains the purpose of this document,the scope of the system, and the characteristics of the intended readers and the organization of the document.

2.1 Purpose of Document

The main purpose of this document is to formally describe program families of the known well-known methods of solving ODEs. The goals and mathematical models used in the LODES code are provided with an emphasis on explicitly identifying assumptions, constraints, and unambiguous definitions.

This document contains the description of the functionalities of the LODES software library as well as the non-functional requirements that the software may have to meet. This document leads is the starting point for the subsequent software development activities, including writing the requirements specification, design specification, code, and the software verification and validation plan and execution.

2.2 Scope of Requirements

The scope of requirements is limited to the analysis of the library of ODE solvers. Given the appropriate inputs, each program in LODES is intended to find the solution to an ODE problem.

2.3 Characteristics of Intended Reader

Reviewers of this document should have an elementary understanding of ordinary differential equations and numerical methods, as typically covered in first and second year Calculus courses. The users of LODES can have a lower level expertise, as explained in Section 3.2.

2.4 Organization of Document

The organization of this document follows the template for a CA for scientific computing software proposed by Smith (2006). The presentation follows the standard pattern of presenting goals, theories, definitions, and assumptions. For readers that would like a more bottom up approach, they can start reading the instance models in Section 5.1 and trace back to find any additional information they require. The instance models provide the methods to solve Ordinary Differential Equations (ODEs).

3 General System Description

This section identifies the interfaces between the system and its environment, describes the user characteristics and lists the system constraints.

3.1 Potential System Contexts

Figure **??** shows the system context. A circle represents an external entity outside the software, the user in this case. A rectangle represents the software system itself (LODES). Arrows are used to show the data flow between the system and its environment.

Programs in LODES are used inside a wrapper program. The external interaction is through program calls. The solution to the ODE is the output of the function. The responsibilities of the user and the system are as follows:

- User Responsibilities:
  - Provide the correct program call, while adhering to conventions of the program's prototype
  - Provide the input details of the ODE to be solved, ensuring no errors in data entry
  - Declaration of the ODE method to be used in solving the ODE
- LODES Responsibilities:
  - Detect an improper input, such as invalid characters in the ODE statement and incomplete input arguments
  - Detect a data type mismatch where applicable, such as a string of characters in a floating point argument
  - Calculate the solution to the ODE problem

3.2 Potential User Characteristics

The end user of LODES should have an understanding of undergraduate Level 1 Calculus.

3.3 Potential System Constraints

There are no system constraints.

4 Commonalities

This section first presents the background and motivation of the program family, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the theories, definitions, assumptions, and finally the instance models as variabilities.

4.1 Background Overview

LODES is a software library developed to be provide a means to solve ODE problems using numerical methods. It can be used to solve different variations of ODEs given their initial values. It can be implemented to find the most accurate method (the method which produces the least error in their scientific computing implementation).

4.2 Terminology and Definitions

This subsection provides a list of terms that are used in the subsequent sections and their meaning, with the purpose of reducing ambiguity and making it easier to correctly understand the requirements:

- Initial values: A "starting point" (x_0, y_0) of known values that exists in the domain of the solution
- Final values: An "ending point" $(x_\mathrm{k}, y_\mathrm{k})$ of unknown value y_k that exists in the domain of the solution
- Step size: The measure of arbitrary positive value (h) from the starting point (x_0, y_0) of the domain to the next (x_1, y_1), where $x_1 = x_0 + h$
- Derivative: The amount by which a a function changes at any given point as an instantaneous rate of change
- Numerical Analysis: A branch of mathematics and computer science wherein the solutions are numerical approximations taking into account the errors involved in the process.
- Recursion:

4.3 Goal Statements

GS1: Given an ordinary differential equation (ODE) represented by $y' = f(x, y)$, the set of initial values x_0 and y_0 that satisfy $y(x_0) = y_0$, and x_k, return y_k such that $y(x_\mathrm{k}) = y_\mathrm{k}$ (the final values), where $y(x)$ is a function, $f(x, y)$ is a function, and x is an independent variable.

GS2: Provide the user the means of providing the required inputs, calling the ODE solver program, and displaying the results.

4.4 Theoretical Models

This section focuses on the general equations and laws that LODES is based on.

Number	T1
Label	**Ordinary Differential Equation**
Equation	$y' = f(x, y)$
Description	The above model gives the definition of an ordinary differential equation. A differential equation is an equation, where the unknown is a function and both the function and its derivatives (rate of change) appear in the equation. An ordinary differential equation is a differential equation involving only ordinary derivatives with respect to a single independent variable. For an arbitrary ODE, the true solution will, in general, be unknown. Numerical methods are used to find numerical approximations of the solution to the ODE.
Source	http://users.math.msu.edu/users/gnagy/teaching/ode.pdf
Ref. By	T2, IM1, IM2

Number	T2
Label	**Existence and Uniqueness of the Solution**
Equations	$y' = f(x, y)$ [T1] $y(x_0) = y_0$ Assuming f and y' are continuous in $R = \{(x, y) : a < x < b, c < y < d\}$, where R is a rectangle and a, b, c, d are its vertices The initial value problem has a unique solution in some interval $x_0 - h < x < x_0 + h$
Description	The above theoretical model shows that when an equation satisfies the initial values, it is assured that a solution to the initial value problem exists. It is desirable to know whether or not the equation has an existing solution before effort is made to solve it. As well, the theoretical model states that if a solution is found, then it is the only solution to the initial value problem.
Source	Nagle, et al, "Solutions and Initial Value Problems," in *Fundamentals of Differential Equations and Boundary Value Problems*, 3rd ed. USA: Addison Wesley Longman, 2000, ch. 1, p. 12.
Ref. By	IM1, IM2

4.5 General Definitions

This section collects the laws and equations that will be used in deriving the data definitions, which in turn are used to build the instance models. This section does not apply to this program family.

4.6 Data Definitions

This section collects and defines all the data needed to build the instance models. The dimension of each quantity is also given. This section does not apply to this program family.

5 Variabilities

This section presents the variabilities in LODES. It details the varying instance models, gives the assumptions for the input, the variabilities in the calculations, and finally the target output.

5.1 Instance Models

This section transforms the problem defined in Section 4.1 into one which is expressed in mathematical terms. It uses concrete symbols defined in Section 4.4.

Number	IM1
Label	**Euler's Method of Finding the Solution to an ODE**
Input	$y' = f(x, y), h, x_0, y_0, x_{\mathrm{k}}$
Output	y_{k} such that $y_{\mathrm{k}} = y(x_{\mathrm{k}})$ using the recursive formulas: $x_{\mathrm{n+1}} = x_{\mathrm{n}} + h, n = 0, 1, 2, ...$ $y_{\mathrm{n+1}} = y_{\mathrm{n}} + h * f(x_{\mathrm{n}}, y_{\mathrm{n}}), n = 0, 1, 2, ...$
Description	$y' = f(x, y)$ is the first order ODE. h is the constant step size. x_0 is the initial value of x. $x_{\mathrm{n+1}}$ is the value of x in the next equation iteration. x_{k} is the final value of x. y_0 is the initial value of y, such that $y_0 = y(x_0)$. $y_{\mathrm{n+1}}$ is the value of y in the next equation iteration. y_{k} is the final value of y. n is the reference recursion step. The above equations are used recursively until $x_{\mathrm{n+1}} = x_{\mathrm{k}}$ and $y_{\mathrm{n+1}} = y_{\mathrm{k}}$.
Sources	Nagle, et al, "The Approximation Method of Euler," in *Fundamentals of Differential Equations and Boundary Value Problems*, 3rd ed. USA: Addison Wesley Longman, 2000, ch. 1, sec. 1.5, pp. 31-32.
Ref. By	IM2

Detailed derivation of Euler's Method

Let $y' = f(x, y)$, $y(x_0) = y_0$, h as a fixed positive number (step-size), and consider the equally spaced points in the domain:

$$x_n = x_0 + nh,\ n = 0, 1, 2...$$

The construction of values y_n that approximate the solution values proceeds as follows. At the point (x_0, y_0), the slope of the solution to $y' = f(x, y)$ is given by $dy/dx = f(x_0, y_0)$. Hence the tangent line to the solution curve at the initial point is:

$$y = y_0 + (x - x_0) * f(x_0, y_0).$$

Using this tangent line to approximate the next point in the solution set, we find that for the point $x_1 = x_0 + h$, we can assume the following approximation:

$$y_1 = y_0 + h * f(x_0, y_0).$$

Repeating the process (recursion), until we reach y_k yields the derivation of Euler's Method.

Number	IM2
Label	**Heun's Method of Finding the Solution to an ODE**
Input	$y' = f(x,y), h, x_0, y_0, x_{\mathrm{k}}$
Output	y_{k} such that $y_{\mathrm{k}} = y(x_{\mathrm{k}})$ using the recursive formulas: $x_{\mathrm{n+1}} = x_{\mathrm{n}} + h, n = 0, 1, 2, ...$ $y_{\mathrm{n+1}} = y_{\mathrm{n}} + \frac{h}{2}\{[f(x_n, y_n) + f(x_n + h, y_n + h * f(x_n, y_n)]\}, n = 0, 1, 2, ...$
Description	$y' = f(x,y)$ is the first order ODE. h is the constant step size. x_0 is the initial value of x. $x_{\mathrm{n+1}}$ is the value of x in the next equation iteration. x_{k} is the final value of x. y_0 is the initial value of y, such that $y_0 = y(x_0)$. $y_{\mathrm{n+1}}$ is the value of y in the next equation iteration. y_{k} is the final value of y. n is the reference recursion step. The above equations are used recursively until $x_{\mathrm{n+1}} = x_{\mathrm{k}}$ and $y_{\mathrm{n+1}} = y_{\mathrm{k}}$.
Sources	Nagle, et al, "Improved Euler's Method," in *Fundamentals of Differential Equations and Boundary Value Problems*, 3rd ed. USA: Addison Wesley Longman, 2000, ch. 3, sec. 3.5, pp. 124-129.
Ref. By	IM**??**

5.2 Input Assumptions

This section focuses on the variabilities and assumptions in the inputs of LODES.

Input Variability	Parameter of Variation
Allowed program family calls	Set of {Euler's Method, Heun's Method, Fourth-Order Taylor Series, Fourth-Order Runge-Kutta}
Allowed order of $f(x, y)$	Set of {First}
Allowed type of $f(x, y)$	Set of {Linear, Nonlinear, Homogeneous, Non-homogeneous}
Allowed dimensions of $f(x, y)$	Set of {Single}
Possible entries of h	set of positive $\mathbb{R}$
Possible entries of x_0	set of $\mathbb{R}$
Allowed dimensions of x_0	Set of {Single}
Possible entries of y_0	set of $\mathbb{R}$
Allowed dimensions of y_0	Set of {Single}
Possible entries of x_k	set of $\mathbb{R}$
Allowed dimensions of x_k	Set of {Single}

5.3 Calculation

This section focuses on the variabilities in the calculations used in LODES.

Variability	Parameter of Variation
Check input?	boolean (false if the input is assumed to satisfy the input assumptions)

5.4 Output

This section focuses on the variabilities in the output of LODES.

Variability	Parameter of Variation
Destination for output y_k	boolean set of {to a file, to the screen, to memory}
Encoding of output y_k	Set of binary, text
Possible values of y_k	set of $\mathbb{R} \cup \{-\infty, \infty,$ undefined$\}$
Output program success	boolean (true if the program successfully solves for the solution to the ODE problem)

6 Appendix

[Your report may require an appendix. For instance, this is a good point to show the values of the symbolic parameters introduced in the report. —SS]

6.1 Symbolic Parameters

[The definition of the requirements will likely call for SYMBOLIC_CONSTANTS. Their values are defined in this section for easy maintenance. —SS]