[redacted]

2 sided / F & B

Bell Laboratories — **Cover Sheet for Technical Memorandum**

[redacted]

Title: **Ode — A Program for the Numerical Solution of Ordinary Differential Equations**

Date: **1 April 1983**

TM: **83-52321-39**

Other Keywords:

Author(s)	Location	Extension	
N. Tufillaro	**MH 2A-443**	**5065**	Charging Case: **12152-6504**
G. A. Ross*			Filing Case: **40106**

ABSTRACT

Ode solves the initial value problem for a family of first-order differential equations when provided with an explicit expression for each equation. Ode parses a set of equations, initial conditions, and control statements and then provides an efficient numerical solution. Ode makes the initial value problem easy to express; for example, the ode program,

```
# an ode to Euler
y  = 1
y' = y
print y from 1
step 0, 1
```

prints 2.718282.

The *Ode User's Manual* contains a guide to applying ode and a discussion of its design and implementation.

* Reed College, Portland, Oregon 97202

Pages Text: 3	Other: 29	Total: 32
No. Figures: 0	No. Tables: 0	No. Refs.: 3

Bell Laboratories

subject: **Ode — A Program for the Numerical Solution of Ordinary Differential Equations**
Charge Case 12152-6504
File Case 40106

date: **1 April 1983**

from: **N. Tufillaro**
MH 52321
2A-443 x5065

G. A. Ross*

TM 83-52321-39

MEMORANDUM FOR FILE

I. INTRODUCTION

Ode is a program which provides a numerical solution to ordinary different equations. The user need only be familiar with the fundamentals of UNIX operating system in order to access and run this numerical software. Ode provides:

— A simple problem-oriented user interface

— A table-driven grammar, simplifying extensions and changes to the language

— A structure designed to ease the introduction of new numerical methods

— Remarkable execution speed and capacity for large problems, for an "interpretive" system.

Ode currently runs under the UNIX operating system found on the DEC family of 11/70, Vax 750, and Vax 780 computers. Ode was developed at Reed College in the summer of 1981 under a UNIX operating system and is public domain software because of a licensing agreement between Bell Labs and non-profit educational institutions which have acquired UNIX. The program is currently in use at numerous educational and industrial sites (Reed

* Reed College, Portland, Oregon 97202

College, Tektronix Inc., U.C. Berkeley, Univ. Utah, Bell Labs, etc.) and a tape containing the program, documentation and copious examples is available from the first author. Ode has been of use in many research problems and incorporated into course work. The first author has used it, for example, to investigate the solution of rate-equations which arise in the modeling of semiconductor lasers.

II. ODE USERS MANUAL

This memorandum contains the *Ode Users Manual* (Appendix) which provides both a pedagogical introduction as well as a comprehensive overview of Ode's design and implementation. Suggestions for extensions and improvements are found at the end of this memorandum. The Ode Users Manual was originally written for the Reed College computing environment. The manual contained here is slightly modified to reflect the Murray Hill computing environment. The reader is advised to use a graphics terminal such as the Tektronix 4014 to work the example problems found in the manual. Those unfamiliar with UNIX should obtain the UNIX starter package available from the Murray Hill Computing Library.

III. FUTURE WORK

This is the first version of Ode, and there is a lot of room for growth and improvements. The first few ideas are quite reasonable. Some other numerical methods could be included. They would make Ode more efficient or more accurate for certain classes of problems. Some candidates are:

— Variable-order codes (e.g. Shampine and Gordon).
— Special codes for solution of so-called "stiff" systems (e.g. Gear).
— Routines to handle other special systems, such as highly oscillatory ones.

Two of the three methods currently supported provide adaptive step sizes, but the step size is always chosen for the dynamic quantity most in error and then applied to all the dynamic quantities in the system. It would be a feasible task to redesign those methods to use different step sizes for each dynamic quantity.

Recent work suggests that the analysis of accumulated error is more tractable than it once was. Some hooks are left in the language to facilitate the addition of accumulated error estimates.

The rest of these suggestions are more speculative. It would be nice to be able to integrate around singularities in the solution. Codes exist that can find and characterize singularities. There are some complex functions on the real domain that would be nice to analyze; however the changes to the numerical routines would be extensive.

N. Tufillaro
Nicholas B. Tufillaro

MH-52321-NT/GAR-slc

G. A. Ross*

Atts.
ODE Users Manual
Appendix
References 1-3