

Definite Integral Approximator

Gage Golish

Fall 2019

Contents

1 Overview	1
2 Highlights	1
2.1 Extendable Equation Parser	1
2.2 Flexible Mesh Size	2
2.3 N-Dimensional Integrals	2
3 Usage	2
3.1 Input	2
3.1.1 Supported Operators	2
3.1.2 Supported Functions	2
3.2 Build	2
3.3 Command Line Arguments	3
3.4 Default Behavior	3
3.5 run_tests.sh	3

1 Overview

A program for approximating definite integrals, written for MATH 613 - Applied Linear Algebra course at Indiana State University. The program uses Simpson's one-third rule for the approximation:

$$\int_a^b f(x) dx \approx \frac{\Delta x}{3} (f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + 2f(x_4) + \cdots + 4f(x_{n-1}) + f(x_n)).$$

This program uses the extended form of Simpson's rule for approximating n-dimensional integrals.

2 Highlights

The following are some implementation details that are of note.

2.1 Extendable Equation Parser

The program contains a custom tokenizer (`scanner.c`) and parser (`parser.c`) that converts an infix input equation to an abstract syntax tree (`ast.c`). Representing the equation as an AST allows for computing the equation on many values efficiently. It is also a simple matter to extend the scanner and parser to handle new operators and functions.

2.2 Flexible Mesh Size

The program uses an adaptive approach for choosing the mesh size. Starting at a low value, the mesh size is doubled until the approximation converges to within four decimal places (`simpson.c`). The mesh size is part of the output of the program.

2.3 N-Dimensional Integrals

The program was built from the ground up to accomodate n-dimensional integrals. The program accepts as input the integral dimension and then a list of integral specifications that consist of the start and end of the integral, as well as the variable the integral should be performed on (`args.c`, `simpson.c`). See the Usage section for more details.

3 Usage

3.1 Input

The program accepts as input an infix equation that can contain numbers, functions, one-character variables, and operators.

3.1.1 Supported Operators

- + (addition)
- - (subtraction)
- * (multiplication)
- / (division)
- ^ (exponentiation)
- ((open parenthesis)
-) (closing paranthesis)

3.1.2 Supported Functions

- `sin` (sine)
- `cos` (cosine)
- `tan` (tangent)
- `ln` (natrual logarithm)
- `sqrt` (square root)

3.2 Build

To build the project, run the following command in the root directory of the project:

```
make integrate
```

This command produces the target executable `integrate`.

3.3 Command Line Arguments

- `-n <dimension>`: Specifies the integral dimension as `<dimension>`.
- `-p <params-list>`: Specifies the list of integral specifications. A colon (`:`) separated list of comma (`,`) separated specifications e.g. `0,3,x:0,3,y` specifies a double integral in which the variable `x` (the outer integral) is integrates over the range `[0, 3]` and variable `y` (the inner integral) is integrated over the range `[0, 3]`.
- `-e <equation>`: Specifies the equation to integrate.

3.4 Default Behavior

If no command line arguments are given, the program will expect the user to first input the integral specification, then the equation. One or the other may be passed as command line arguments, while the other is read from `stdin`. For example,

```
./integrate
0 3.14 x
sin(x)
```

is the `stdin` way of specifying $\int_0^\pi \sin(x)dx$. The command line way would be:

```
./integrate -n 1 -p '0,3.14,x' -e 'sin(x)'
```

3.5 run_tests.sh

Run the interactive program `run_tests.sh` to perform some test examples using the program. The test examples are as follows:

- $\int_0^{2\pi} \sin(x)dx$
- $\int_0^{2\pi} \cos(x)dx$
- $\int_0^{10} \sin(x^2 + 1)dx$
- $\int_{10}^{20} \sin(x^2 + 1)dx$
- $\int_{-1}^1 \int_1^2 x^2y + xy^2 dx dy$