Computing 1000's of Integrals and Sums

J. Linser[1] R. Mekhe[2]

Department of Mathematics
Ohio University

3/11/2011 Pi Day

Outline

Outline

Purpose

- The overall goal of our work was to "write code to write code" that would eventually compute all the cases stated in the paper. We quickly found that in order to achieve this we needed to break the problem down in smaller, more manageable problems. To begin we picked specific propositions.

Outline

Class IS

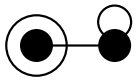
- PYTHON code:
- Defined a class IS
- Input values for the class were geminals, curves, W function, phi's(list of strings) and theta's(list of strings) and extras
- For the input geminal and curves, the code generates the formula
- It is then printed in LATEX

Outline

Propositions

- The propositions were created to simplify the number cases encountered.
- For each proposition we wrote functions to apply the propositions to a user's input. The functions' goals are the following:
- 1) See if the function could be applied to the input
- 2) If condition (1) was satisfied, apply the proposition to the input. Otherwise return false and go to the next function.
- 3) Record what the function did
- 4) Compute the resulting expression
- 5) Output in LateX what was done and the resulting expression

First Reduction Proposition:

- It is called ‘gencircontract’
- It looks for the following subgraph: 

Description of proposition code:

- For given geminal and index connections, the code first searches for a sub graph loop with one connecting geminal in the given graph structure
- If there is no such sub graph, then it returns false as output and no reduction takes place
- If there is any, then the code finds it and selects the last one
- It also eliminates the connecting geminal

Description of proposition code: Cont'd

- But while doing so, it takes care of the other end of connecting geminal
- If it is connected with some other index, then we don't have to worry for an isolated node being lost and in the output we get the tuples of new geminals and curves. It is then printed in Latex as part of the code.
- But if the other end is not connected , then we get an isolated node which is appended back to the tuple of geminals already existing and we get the new tuples of geminals and curves . The output gets printed in Latex as a mathematical formula
- In the output all indexes are shifted down by one unit.

Example

- Start with

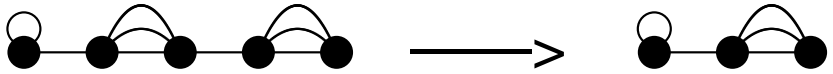
$$\sum_a \sum_b \sum_c \sum_d \sum_e \int \int \int \int \int \phi_a(\gamma_0)\theta_a(\gamma_0)\phi_b(\gamma_1)\theta_b(\gamma_2)\phi_c(\gamma_2)\theta_c \tag{1}$$

Then do

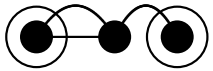
$$g(\gamma_1) = \int \sum_a \phi_a(\gamma_0)\theta_a(\gamma_0) w_0(|\gamma_0 - \gamma_1|) d\gamma_0 \tag{2}$$

End with

$$\sum_a \sum_b \sum_c \sum_d \int \int \int \int \phi_a(\gamma_0)\theta_a(\gamma_1)\phi_b(\gamma_1)\theta_b(\gamma_0)\phi_c(\gamma_2)\theta_c(\gamma_3)\phi_d \tag{3}$$

- This does the following reduction:
- 

Second Reduction Proposition:

- It is called ‘genericgabbmgeneric contract’
- It looks for the following subgraph: 

Third Reduction Proposition:

- It is called ‘genericabbmgeneric contract’
- It looks for the following subgraph: