A Feynman graph selection tool in GRACE system

Fukuko YUASA[a]*, Toshiaki KANEKO[b], Tadashi ISHIKAWA[a]

a)KEK, 1-1 OHO Tsukuba Ibaraki, Japan 305-0801
b)Meiji-Gakuin Univ., Kamikurata 1518, Totsuka, Yokohama, Kanagawa Japan 244-0816

Abstract. We present a Feynman graph selection tool `grcsel`, which is an interpreter written in C language. In the framework of `GRACE`, it enables us to get a subset of Feynman graphs according to given conditions.

INTRODUCTION

Using an automatic Feynman graph calculation package, we can generate the information of all Feynman graphs for given processes. Sometimes it is necessary to select graphs from the set of all graphs by some conditions. However, it is not so easy to select them correctly by hand when a huge number of graphs is involved, such as higher order corrections or SUSY processes.

A program `grcsel` selects out a subset of Feynman graphs from the set of graphs, generated by GRACE(1), according to given selection conditions. The output information of selected graphs is written in the same format as that of the original set. This enables us to generate Feynman amplitudes within GRACE in the same procedure as for all graphs. So we can perform cross section calculation, gauge invariance check, event generation and so on for the selected ones. `grcsel` helps us:

1. to find decay-graphs and evaluate signal/background ratio,
2. to check the accuracy of approximated calculation,
3. to confirm precision of the calculation,
4. to reduce the calculation time,
5. to develop kinematics routine.

OVERVIEW OF `GRCSEL`

`grcsel` consists of three parts: a steering-part defines basic functions of graph selections and reads input files, an interpreter-part parses and evaluates commands, and a utility-part handles subsets of particles, vertices and graphs.

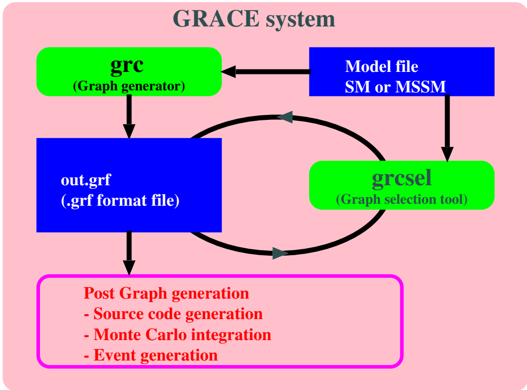

FIGURE 1.

Once a physics process and the order of calculation are fixed, Feynman graphs are generated by `grc` program with specified Feynman rules described in physics model file(2). The information on graphs generated are stored in a file named `out.grf` (we call the format of this file *`.grf` format*).

`grcsel` reads the physics model file and `out.grf` and selects graphs according to a kind of propagator, characteristics of graph topology, a type of vertex or a graph number. `grcsel` outputs those selected graphs in the same format as `out.grf`. Successively this output file can be used as the input to source code generation for Monte Carlo integration or event generation. We can also use `grcsel` again reading output of previous execution of `grcsel`. The schematic view of how `grcsel` works in GRACE system is shown in Fig. 1.

* E-mail:fukuko.yuasa@kek.jp

Table 1. Basic functions to select and output graphs.

Function	Description
`cutprop`	Select graphs with a specified propagator.
`selvlegs`	Select graphs with a vertex consisting specified particles.
`selvertex`	Select graphs with a specified vertex.
`outgset`	Output a set of graphs.
`renumgset`	Renumber and output a set of graphs.

Running `grcsel`

Graph selection starts by the program `grcsel`:

```
grcsel
```

This program requires `out.grf` file by default. The graph selection commands are read through standard input, which may be given interactively or by a script file. With a script file where `grcsel` commands are prepared, we can redirect that file:

```
grcsel < command.in
```

To use another input `.grf` format file, e.g. `out1.grf`, instead of `out.grf`, we can add the filename after `grcsel` command as:

```
grcsel out1.grf < command.in
```

`grcsel` command

In a script file there are a series of `grcsel` commands such as declaration of variables and basic functions to specify the selection conditions or operators. `grcsel` has 14 basic functions in total. Three of them return a subset of graphs in accordance with specified selection condition and two functions output set of graphs. They are summarized in Table 1.

In `grcsel`, graphs, particles or vertices are treated as elements of a *set* of type `gset`, `pset` or `vset`, which are defined as a set of graphs, particles or vertices, respectively. Set variables have to be declared at first with their types. Operations on sets are available and are shown in Table 2.

Table 2. *set* operators

Operator	Function
&	Set intersection
|	Set union
~	Complement of set

EXAMPLE

In the following example, graphs with ν_e propagator connected to the initial electron and final W^- are selected among graphs of $e^+e^- \to W^+W^-\gamma$ process. Selected graphs are output into a file named `out1.grf`.

```
% e+ e- --> W+ W- Photon
%
% out1.grf : with neutrino propagator
% at the vertex of initial elec-
tron and
% final W-.
%

gset gs0, gs1;

gs0 = ~[];   % all graphs

gs1 = cutprop(gs0, [``nu-e''], [0,3]);

outgset(``out1.grf'', gs1);

quit;
```

In Fig. 2, selected graphs are shown.

REMARKS

`grcsel` has been developed in the framework of GRACE 2.1.7.4. It can handle tree and 1-loop graphs and it supports standard and MSSM physics model. `grcsel` is included in a distribution kit of GRACE 2.1.7.4.

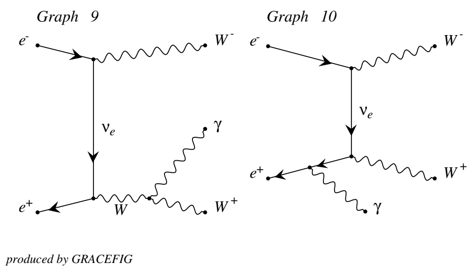

FIGURE 2.

ACKNOWLEDGMENTS

We wish to thank the members of MINAMI-TATEYA collaboration for continuous discussions and many kinds of support. We are also grateful to express our sincere gratitude to Prof. Y.Shimizu for the valuable suggestions and continuous encouragements. Authors appreciate Prof. Y.Watase for the encouragements. This work was supported in part by the Grant-in Aid (No. 12680363, 10640285, 10680366 and 11440083) of Monbu-sho, Japan.

REFERENCES

1. MINAMI-TATEYA group: `GRACE` manual. KEK Report 92-19.
   F.Yuasa *et al.*: Prog. Theor. Phys. Suppl. **138** (2000) 18-23.
2. T.Kaneko: Comput. Phys. Commun. **92** (1995) 127-152.