Factorization Method of Tree Feynman Amplitudes

Toshiaki KANEKO

2005/5/26 at ACAT 2005

Contents

1 Introduction

- Many final particles $\Rightarrow$ big number of graphs even for tree processes
  ϕ^3 model : $(2E-5)!!$ graphs, ($E = \#$ external particles)

- Faster algorithms

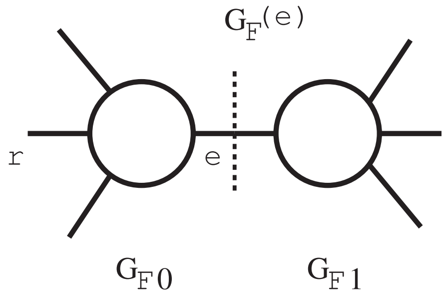
  - `Alpha`
    F. Caravaglios and M. Moretti, *Phys. Lett.* **B358** (1995) 332;
    F. Caravaglios and M. Moretti, *Z. für Physik* **C 74** (1997) 291.

  - `HERAC`
    A. Kanaki, C.G. Papadpouos, *Comput. Phys. Commun.* **132** (2000) 306.

  - `O'Mega`
    M. Moretti, T. Ohl and J. Reuter, `hep-ph/0102195`.

- Traditional calculations with Feynman graphs
  $\Rightarrow$ sub-graphs are calculated repeatedly.

- Basic idea



1. Gather Feynman graphs with the same propagator $e \Rightarrow \mathcal{G}_{\mathcal{F}}(e)$.
2. Cut graphs at propagator e.
3. Calculate amplitudes for the sets of sub-graphs $\mathcal{G}_{\mathcal{F}0}$ and $\mathcal{G}_{\mathcal{F}1}$.
4. Multiply the propagator and the sub-amplitudes.

The number of graphs : $|\mathcal{G}_{\mathcal{F}}(e)| = |\mathcal{G}_{\mathcal{F}0}| \times |\mathcal{G}_{\mathcal{F}1}|$.
The amount of calculations : $|\mathcal{G}_{\mathcal{F}}(e)| = |\mathcal{G}_{\mathcal{F}0}| + |\mathcal{G}_{\mathcal{F}1}|$.

- Over which propagators it should be summed up ?
- Double counting ?
  $\Rightarrow$ need to classify Feynman graphs.
- Generation of $\mathcal{G}_{\mathcal{F}0}$ and $\mathcal{G}_{\mathcal{F}1}$ and then multiplication $\Rightarrow$ generation of $\mathcal{G}_{\mathcal{F}}$?
- Factorize by edges or vertices ?

- Model

  We discuss in a simple model

  - One scalar particle is included.
  - $n > 2$ for n-point vertices.
    No tadpole nor 2-point vertex. ($\Rightarrow$ effective vertices and propagators)
  - At most one kind of n-point vertex for each n.
  - Consider only tree graphs including at least one vertex.

- Terms

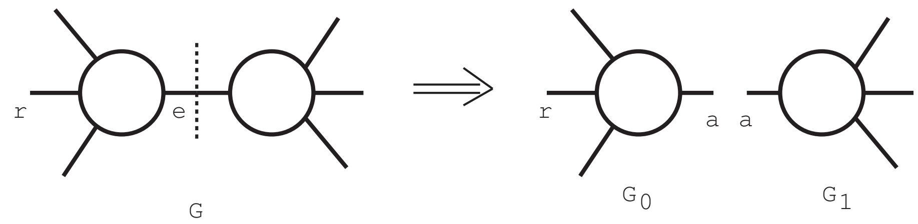
  node : an external particle or a vertex.

  edge : a connection between two edges.

  root : a fixed external particle.

2 Classification of tree Feynman graphs

- Decomposition of a graph $G(\mathcal{E})$



$$G(\mathcal{E}) \; = \; G_0(\mathcal{E}_0) \odot_e G_1(\mathcal{E}_1).$$

– e : edge.

– $\mathcal{E}, \mathcal{E}_0, \mathcal{E}_1$: sets of external particles.

– $\mathcal{E}_0$ includes root r.

– $\mathcal{E}_0 \cap \mathcal{E}_1 = \{a_e\}$: a_e is a new external particle appearing at e. (We call a_e the *cut node* of edge e).

– e and a_e are determined by $\mathcal{E}_0$ or $\mathcal{E}_1$ using momentum conservation law.

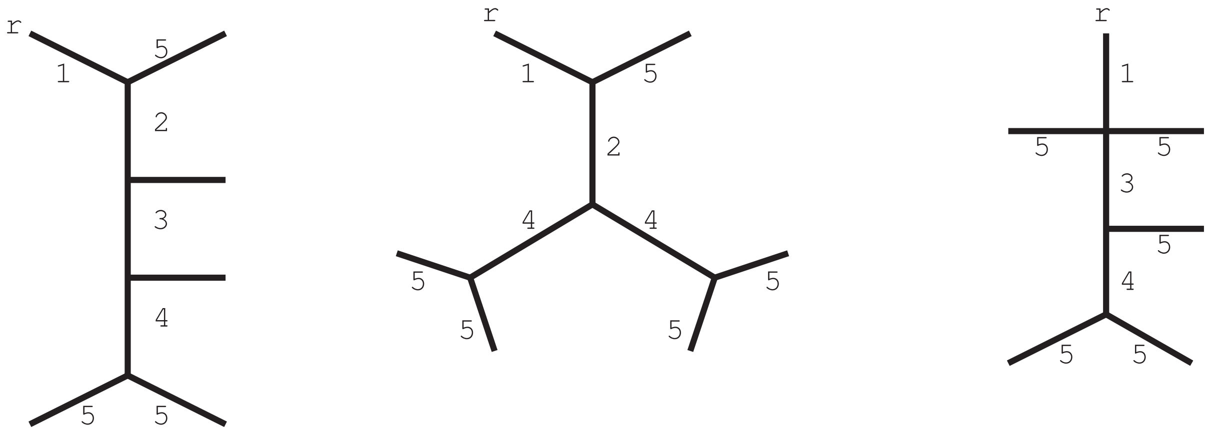
- *D-number* $F(e)$ of edge e :

$$\begin{aligned} F(e) &= |\mathcal{E}_0 - \{a_e\}| = |\mathcal{E}_0| - 1, \\ 1 &\leq F(e) < |\mathcal{E}|. \end{aligned}$$

  – $F(e_r) = 1$ when e_r is adjacent to the root.

  – For given $k \quad (1 < k < |\mathcal{E}|)$, no edge may satisfy $F(e) = k$.

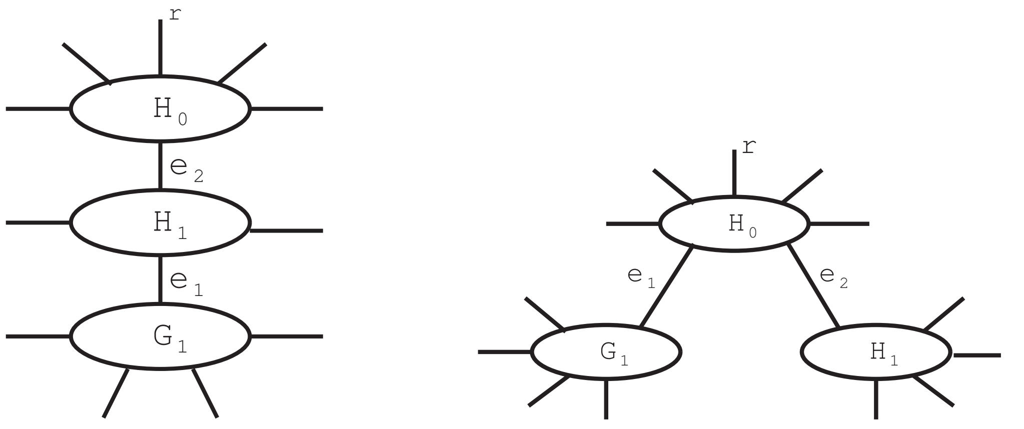
  – More than one edges may have the same value of d-number.



- **Lemma 2.1** *For $k \leq |\mathcal{E}|/2$, there is at most one edge e such that $F(e) = k$.*

Proof

1. Assume that two different edges e_1 and e_2 have the same value of d-number $F(e_1) = F(e_2)$.



2. Edge e_2 cannot be on the path $r - e_1$, since there are no tadpole nor 2-point functions. Edge e_1 cannot be on the path $r - e_2$ by the same reason.

3. Decompose graph G in terms of e_1 and e_2:

$$\begin{aligned} G &= G_0(\mathcal{E}_0) \odot_{e_1} G_1(\mathcal{E}_1), \qquad r \in \mathcal{E}_0, \\ G_0 &= H_0(\mathcal{E}'_0) \odot_{e_2} H_1(\mathcal{E}'_1), \qquad r \in \mathcal{E}'_0. \end{aligned}$$

then ($\mathcal{E}_0$ etc. includes cut nodes)

$$\begin{aligned} k &= F(e_1) = |\mathcal{E}_0| - 1 = |\mathcal{E}'_0| + |\mathcal{E}'_1| - 3, \\ &= F(e_2) = |\mathcal{E}| - |\mathcal{E}'_1| + 1. \end{aligned}$$

We obtain

$$2k = |\mathcal{E}| + |\mathcal{E}_0'| - 2.$$

$|\mathcal{E}_0'| \geq 3 \Rightarrow 2k \geq |\mathcal{E}| + 1.$

$\diamondsuit$

- *Central edge* e_c
  - Definition :

$$F(e_c) \leq |\mathcal{E}|/2, \qquad F(e) \leq |\mathcal{E}|/2 \Rightarrow F(e) \leq F(e_c).$$

  - Central edge uniquely exists in a graph
    $\Leftarrow$ the last lemma + $F(e_r) = 1$ (e_r is adjacent to the root).

- Classification without duplication :

$$\mathcal{G}_{\mathcal{F}}(\mathcal{E}) = \sum_{e, F(e) \leq |\mathcal{E}|/2} \mathcal{G}_{\mathcal{F}}(\mathcal{E}, e),$$

$$\mathcal{G}_{\mathcal{F}}(\mathcal{E}, e) \cap \mathcal{G}_{\mathcal{F}}(\mathcal{E}, e') = \emptyset, \qquad \text{for } e \neq e'.$$

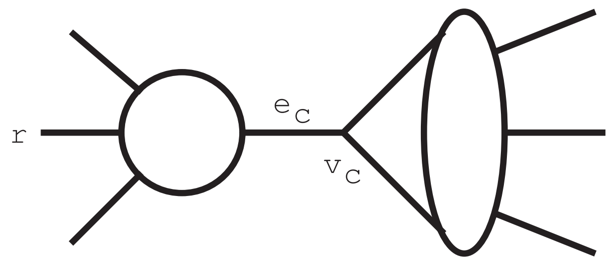
$\mathcal{G}_{\mathcal{F}}(\mathcal{E})$: the set of all possible tree Feynman graphs

$\mathcal{G}_{\mathcal{F}}(\mathcal{E}, e)$: the set of all possible tree Feynman graphs with central edge e.

- Classification in terms of an edge or a vertex ?
  - Condition for $\mathcal{G}_{\mathcal{F}}(\mathcal{E}, e)$ is not simple. $\Rightarrow$ classification of generated of graphs ?
  - Classification in terms of vertex $\Rightarrow$ simpler condition. $\Rightarrow$ generation of graphs ?
- *Central vertex*
  - Define *d-number* $F(v)$ to a node v :

$$
\begin{aligned}
F(v) &= \begin{cases} 0, & \text{for } v \text{ is the root,} \\ \min_{e \in E_v} F(e), & \text{otherwise,} \end{cases} \\
E_v &: \quad \text{the set of all edges adjacent to } v.
\end{aligned}
$$

  - *Central vertex* v_c : vertex satisfying $F(v_c) = F(e_c) \leq |\mathcal{E}|/2$.
  - Central vertex is adjacent to e_c at the far side from root r.
  - Central vertex is uniquely determined in each graph.



  - $\mathcal{G}_{\mathcal{F}}(\mathcal{E}, f, d)$: a set of graphs with central vertex v_c with $f = F(v_c)$ and $d = \deg(v_c)$.

Lemma 2.2 *The set of all the graphs $\mathcal{G}_{\mathcal{F}}(\mathcal{E})$ is decomposed by:*

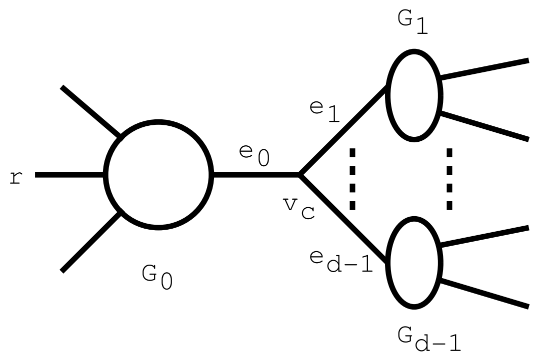
$$\begin{aligned}
\mathcal{G}_{\mathcal{F}}(\mathcal{E}) &= \bigcup_{f=1}^{\left\lfloor \frac{|\mathcal{E}|}{2} \right\rfloor} \bigcup_{d \geq d_{min}} \mathcal{G}_{\mathcal{F}}(\mathcal{E}, f, d), \\
\mathcal{G}_{\mathcal{F}}(\mathcal{E}, f, d) \cap \mathcal{G}_{\mathcal{F}}(\mathcal{E}, f', d') &= \emptyset, \qquad \textit{for } f \neq f' \textit{ or } d \neq d'.
\end{aligned}$$

3 Factorization of Feynman amplitudes

- Decomposition of graph $G \in \mathcal{G}_{\mathcal{F}}(\mathcal{E})$ in terms of a vertex v_c.
  - r : root of G
  - $e_0, ..., e_{d-1}$: edges adjacent to v (degree d).
  - Decompose graph G in terms of $\{e_i\}$.
  - a_i : cut node of e_i.

$$\begin{aligned} G &= V(d, a_0, ..., a_{d-1}) \odot_{e_0} G_0(\mathcal{E}_0) \odot_{e_1} ... \odot_{e_{d-1}} G_{d-1}(\mathcal{E}_{d-1}), \\ r &\in G_0 \\ a_i &\in \mathcal{E}_i \text{ for } i = 0, ..., d-1, \end{aligned} \tag{1}$$

where $V(d, a_0, ..., a_{d-1})$ is a graph obtained by connecting vertex v_c to a_i ($i = 0, ..., d-1$).



Lemma 3.1 *In the decomposition (1) of G in terms of a vertex v, the necessary and sufficient condition for $v = v_c$ is:*

$$
\begin{aligned}
0 \leq |\mathcal{E}_0| &\leq \frac{|\mathcal{E}|}{2} + 1, \\
0 \leq |\mathcal{E}_k| &< \frac{|\mathcal{E}|}{2} + 1, \qquad \text{for} \quad k \geq 1.
\end{aligned} \tag{2}
$$

Proof

– $v = v_c \Rightarrow$ condition (2):

$$
\begin{aligned}
|\mathcal{E}|/2 &\geq F(e_c) = F(e_0) = |\mathcal{E}_0| - 1, \\
|\mathcal{E}|/2 &< F(e_k) = |\mathcal{E}| - |\mathcal{E}_k| + 1.
\end{aligned}
$$

– Condition (2) $\Rightarrow v = v_c$

1. e_c is not included in $G_k, k \geq 1$
   If $e \in G_k \Rightarrow F(e) > F(e_k) = |\mathcal{E}| - |\mathcal{E}_k| + 1 > |\mathcal{E}|/2$.
   $\Rightarrow e_c \in G_0$ or $e_c = e_0$.
2. e_c cannot on the path $r — a_0$ in G_0
   If e is on this path $\Rightarrow F(e) < F(a_0) = F(e_0)$.

3. Let $e \in G_0$ be not on the path r — a_0 in G_0.
   Decompose G_0 in terms of e:

$$G_0 \;=\; H_0(\mathcal{E}'_0) \odot_e H_1(\mathcal{E}'_1), \qquad r \in \mathcal{E}'_0.$$

   As $r, a_0, a_e \in H_0$, $|\mathcal{E}'_0| > 2$ and $|\mathcal{E}'_1| = |\mathcal{E}_0| - |\mathcal{E}'_0| + 2 < |\mathcal{E}_0|$.
   We obtain

$$F(e) \;=\; |\mathcal{E}| - |\mathcal{E}'_1| + 1 > |\mathcal{E}| - |\mathcal{E}_0| + 1 > |\mathcal{E}|/2$$

   $\Rightarrow$ e cannot be e_c.

4. As the result, we get $e_c = e_0$.

$\diamondsuit$

- *Standard decomposition*
  Decomposition of $\mathcal{E}$ into $(\mathcal{E}_0, ..., \mathcal{E}_{d-1})$ which satisfies for $f \quad (0 < f \leq |\mathcal{E}|/2)$ and $d \quad (d_{min} \leq d)$:

$$
\begin{aligned}
\mathcal{E} &= \bigcup_{i=0}^{d-1} \mathcal{E}_i, \\
\mathcal{E}_i \bigcap \mathcal{E}_j &= \emptyset, \qquad \text{for } i \neq j, \\
r &\in \mathcal{E}_0, \\
|\mathcal{E}_0| &= f, \\
|\mathcal{E}_i| &< \frac{|\mathcal{E}|}{2}, \qquad \text{for } i > 0,
\end{aligned}
$$

  where r is the root of $\mathcal{E}$.

- Remove ambiguity in reordering $\{\mathcal{E}_i\}$:
  - Number the external particles from 0 to $|\mathcal{E}| - 1$.
  - Introduce the following condition:

$$e_i < e_{i+1} \qquad \text{for } e_i = \min\{e|e \in \mathcal{E}_i\} \text{ and } i > 0.$$

- $\mathcal{E}(f, d)$: The set of all standard decomposition for given f and d.

Lemma 3.2 *Set of graphs $\mathcal{G}_{\mathcal{F}}(\mathcal{E}, f, d)$ is decomposed as:*

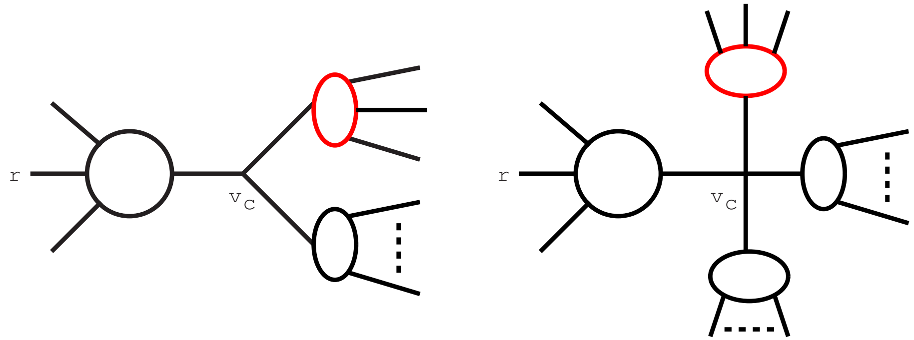
$$\mathcal{G}_{\mathcal{F}}(\mathcal{E}, f, d) \;=\; \bigcup_{\{\mathcal{E}_0,...,\mathcal{E}_{d-1}\}\in\mathcal{E}(f,d)} \{V(d, a_0, ..., a_{d-1})\} \odot\mathcal{G}_{\mathcal{F}}(\mathcal{E}_0 \cup \{a_0\}) \odot ... \odot \mathcal{G}_{\mathcal{F}}(\mathcal{E}_{d-1} \cup \{a_{d-1}\}).$$

In this decomposition, there appear no duplicated graphs.

- The set of all tree Feynman graphs are constructed from sets of sub-graphs.
- $\mathcal{A}(\mathcal{G}_{\mathcal{F}}(\mathcal{E}))$: amplitude of the set of graphs $\mathcal{G}_{\mathcal{F}}(\mathcal{E})$.
- Algorithm
  1. Sum up the result of step 2 for $f = 1, ..., |\mathcal{E}|/2, d = d_{min}, ...$.
  2. Generate all the possible elements of $\mathcal{E}(f, d)$ and sum up the result of step 3.
  3. Calculate momenta of $(a_0, ..., a_{d-1})$ and then $\mathcal{A}(V(v, a_0, ..., a_{d-1}))$.
     Multiply $\mathcal{A}(V)$ and the results of step 4 for $\mathcal{A}(\mathcal{G}_{\mathcal{F}}(\mathcal{E}_i \cup \{a_i\}))$ with propagator e_i $(i = 0, ..., d-1)$.
  4. Calculate $\mathcal{A}(\mathcal{G}_{\mathcal{F}}(\mathcal{E}_i \bigcup\{a_i\}))$ recursively by step 1.

- Comments

1. No need to generate Feynman graphs explicitly.
2. $\mathcal{A}(\mathcal{G}_{\mathcal{F}}(\mathcal{E}_0 \bigcup \{a_0\}))$ can be factored out for common $\mathcal{E}_0$.
   This factorization corresponds to the classification of graphs in terms of e_c. This optimization can be done by the code generator.
3. The same $\mathcal{G}_{\mathcal{F}}(\mathcal{E}_i \bigcup \{a_i\})$ may still appear many times, it can be avoided by keeping a table of pairs $(\mathcal{E}, \mathcal{A}(\mathcal{G}_{\mathcal{F}}(\mathcal{E})))$.



4. This method corresponds to a choice of *"keystones"* which was proposed by `O'Mega` group. It is shown that no double counting appears with our set of keystones.

4 Performance of the algorithm

Performance in $\phi^3 + \phi^4$ theory

Comparison

- The numbers of operations in $\phi^3 + \phi^4$ theory are counted.
- This method is compared with traditional calculation method.

Comments:

- The acceleration ratio increases as the number of external particles increases.
- The number of additions of partial amplitudes shows the worst acceleration.
- In the realistic model, which includes several particles and vertices, partial amplitudes are calculated for each helicity amplitudes.
- The cost of an addition of partial amplitudes increases as the number of external particles increases, while calculations of a vertex and a propagator are kept constant.

	N_e	N_v	N_p	N_a	N_g
M_1	3	1	0	0	1
M_0	3	1	0	0	1
R	3	1	0	0	1
M_1	4	7	3	3	4
M_0	4	7	3	3	4
R	4	1	1	1	1
M_1	5	35	28	24	25
M_0	5	65	40	24	25
R	5	1.86	1.43	1	1
M_1	6	165	165	129	220
M_0	6	755	535	219	220
R	6	4.58	3.24	1.70	1
M_1	7	686	911	594	2485
M_0	7	10605	8120	2484	2485
R	7	15.46	8.91	4.18	1
M_1	8	3283	4781	2967	34300
M_0	8	175035	140735	34299	34300
R	8	53.32	29.43	11.56	1
M_1	9	12895	21932	12060	559405
M_0	9	3322165	2762760	559404	559405
R	9	257.63	125.97	46.39	1
M_1	10	57205	103137	54525	10525900
M_0	10	71307775	60781875	10525899	10525900
R	10	1246.53	589.33	193.05	1
M_1	11	217118	435811	210462	224449225
M_0	11	1708131425	1483682200	224449224	224449225
R	11	7867.29	3404.41	1066.45	1

The numbers of operations are counted for:

- N_e: the number of external particles,
- N_v: the number of vertices to be calculated,
- N_p: the number of connections in terms of propagators,
- N_a: the number of additions of partial amplitudes,
- N_g: the number of corresponding Feynman graphs.

These numbers are compared with:

- M_1: this method,
- M_0: traditional calculation of Feynman graphs,
- R: ratio M_0/M_1 of acceleration.

Table 1: The number of operations in $\phi^3 + \phi^4$ theory

Performance in the standard model

- Fortran code generator `grcft`
  - A new component of `GRACE`.
  - Based on `CHANEL` library (vertices and propagators).
  - Common kinematics library with `GRACE`.
  - Direct generation of Fortran code form process definition.
- Performance test:
  - Compare with traditional `GRACE`
  - Numerical calculation at fixed phase space points.
- Electro-weak theory (no colored particle)

Process	# graphs	# lines		CPU-time
		GRACE	`grcft`	(ratio)
$e^+e^- \Rightarrow (e^+e^-)^2$	654	60933	21909	3.60
$e^+e^- \Rightarrow (e^+e^-)^3$	145128	16253671	150219	83.70
$e^+e^- \Rightarrow e^+e^-\mu^+\mu^-\tau^+\tau^-$	12094	1368735	41648	15.14
$e^+e^- \Rightarrow e^+e^-\mu^+\mu^-\tau^+\tau^-\gamma$	117680	14923326	101629	142.86

- Color factor
  - Traditional method : coefficients of color base for each graph.
  - This method : recursive summation of sub-graphs.
    ⇒ color factors for sub-graphs.
  - Addition and multiplication of sub-amplitudes with color factor.
- Method
  - Amplitudes A_1 and A_2 of sub-graphs.
  - Color bases $B_i^{(1)}$ and $B_i^{(2)}$
  - Coefficients $a_i^{(1)}$ and $a_i^{(2)}$

$$
\begin{aligned}
A_1 &= \sum_i a_i^{(1)} B_i^{(1)}, \\
A_2 &= \sum_i a_i^{(2)} B_i^{(2)}.
\end{aligned}
$$

  - Addition : let $B_i^{(1)} = B_i^{(2)} = B_i$

$$
A_1 + A_2 = \sum_i (a_i^{(1)} + a_i^{(2)}) B_i.
$$

– Multiplication : multiplication table $\{e_{i,j,k}\}$ of color bases

$$B_i^{(1)} B_j^{(2)} = \sum_k e_{i,j,k} B_k.$$

B_k : color bases of multiplied sub-amplitude.

$$\begin{aligned} A_1 A_2 &= \sum_{i,j} a_i^{(1)} a_j^{(2)} B_i^{(1)} B_j^{(2)} \\ &= \sum_k \sum_{i,j} a_i^{(1)} a_j^{(2)} e_{i,j,k} B_k. \end{aligned}$$

- FORTRAN code generation
  - Multiplication table is prepared in the code generator.
  - Coefficients are calculated in FORTRAN code.

$$a_k = \sum_{i,j} a_i^{(1)} a_j^{(2)} e_{i,j,k}.$$

- Performance

process	# graphs	# of lines		CPU-time
		GRACE	`grcft`	(ratio)
$u\bar{u} \Rightarrow 5g$	1240	310053	171944	1.07
$u\bar{u} \Rightarrow u\bar{u}c\bar{c}t\bar{t}$	200	28329	26375	0.81
$u\bar{u} \Rightarrow u\bar{u}c\bar{c}t\bar{t}g$	2658	386977	76459	1.16
$u\bar{u} \Rightarrow u\bar{u}c\bar{c}gg$	1454	301646	55475	1.10
$u\bar{u} \Rightarrow u\bar{u}c\bar{c}ggg$	19808	3353808	431866	1.32
$gg \Rightarrow gggg$	220	126865	114486	1.09
$gg \Rightarrow ggggg$	2485	730198	3862290	0.032

- Overhead of this method is large. $\Rightarrow$ needs improvement.

5 Summary

- Classification of tree Feynman graphs without duplication.
- Factorized calculation of Feynman amplitudes.
- Good acceleration for electro-weak theory.
- Overhead of color factor is large $\Rightarrow$ need improvement.
- This method corresponds to a choice of "*keystones*" proposed by `O'Mega` group.