Optimizing automatic differentiation using activity analysis
I’m from originally from Kyiv, Ukraine

Studied Applied Mathematics at the Taras Shevchenko National University of Kyiv

Currently live in Wuerzburg, Germany

Study Mathematics at Julius Maximilians Universität Würzburg

Fluent in Ukrainian and English and can speak some German

Often cycle and love jogging

Enjoy cooking our national cuisine
So what is activity analysis (AA)?

First a bit of motivation…
Sometimes Clad produces adjoints that are useless for the desired final derivative. Let’s call those variables *passive*. Otherwise, the variable is called *active*. Now Clad assumes all variables are active, but we can do much better using AA.

Let’s see the example:

<table>
<thead>
<tr>
<th>code</th>
<th>forward mode</th>
<th>fm+aa</th>
</tr>
</thead>
<tbody>
<tr>
<td>( f(a, b, c): )</td>
<td>( f_c(darg0(a, b, c)): )</td>
<td>( f_c(darg0(a, b, c)): )</td>
</tr>
<tr>
<td>( x = a \times b )</td>
<td>( d_a = 1 )</td>
<td>( d_a = 1 )</td>
</tr>
<tr>
<td>( d = a \times c )</td>
<td>( d_b = 0 )</td>
<td>( d_b = 0 )</td>
</tr>
<tr>
<td>( \text{return } x )</td>
<td>( d_c = 0 )</td>
<td>( d_c = 0 )</td>
</tr>
<tr>
<td></td>
<td>( d_x = d_a \times b + a \times d_b )</td>
<td>( d_x = d_a \times b + a \times d_b )</td>
</tr>
<tr>
<td></td>
<td>( x = a \times b )</td>
<td>( x = a \times b )</td>
</tr>
<tr>
<td></td>
<td>( d_d = d_a \times c + a \times d_c )</td>
<td>( d = a \times c )</td>
</tr>
<tr>
<td></td>
<td>( d = a \times c )</td>
<td>( \text{return } d_x )</td>
</tr>
<tr>
<td></td>
<td>( \text{return } d_x )</td>
<td></td>
</tr>
</tbody>
</table>
AA is the combination of a forward and a backward analysis.

It propagates forward the \textbf{Varied} set of the variables that depend in a differentiable way on some independent input. Similarly, it propagates backwards the \textbf{Useful} set of the variables that influence some dependent output in a differentiable way.

Since the relation “depends in a differentiable way of” is transitive on code sequences, the essential equations of the propagation are:

\begin{align*}
\text{Varied}^+(I) &= \text{Varied}^-(I) \times \text{Diff} - \text{depp}(I) \\
\text{Useful}^-(I) &= \text{Diff} - \text{dep}(I) \times \text{Useful}^+(I)
\end{align*}

Where \text{Varied}^-(I), \text{Varied}^+(I) \text{ are sets of } \textbf{Varied} \text{ variables before and after } I - th \text{ instruction,}

\((v_1, v_2) \in \text{Diff} - \text{dep}(I) \text{ iff } v_2 \text{ depends on } v_1 \text{ after } I - th \text{ instruction,}

v_2 \in S \times \text{Diff} - \text{dep}(I) \iff \exists v_1 \in S, (v_1, v_2) \in \text{Diff} - \text{dep}(I)\)
And finally we define the set of all active variables as follows:

$$Active^+(I) = Varied^+(I) \cap Useful^+(I)$$
Note:

After AA is implemented and both AA and TBR analysis are default, there is a potential in modifying TBR using AA.

References