[Sparse matrix multiplication – self code]

```cpp
#include <cstdio>
#include <cstdlib>
#include <vector>
#include <utility>
#include <map>
#include <algorithm>

using namespace std;

const int TIMES = 1e7;
const int LIMIT = 1e6;

double av[TIMES];
int at[LIMIT];
int ac[LIMIT + 1];
int ar[TIMES];

double bv[TIMES];
int bt[LIMIT];
int bc[TIMES];
int br[LIMIT + 1];

map<int, double> m[LIMIT];
vector<pair<pair<int, int>, double> > vct;

int main() {
    FILE *fa = fopen("ma.txt", "r");
    FILE *fb = fopen("mb.txt", "r");

    if (fa == NULL || fb == NULL) {
        fprintf(stderr, "File ma.txt or mb.txt cannot open.\n");
        exit(EXIT_FAILURE);
    }

    int x;
    int y;
    double v;

    for (int i = 0; i < TIMES; i++) {
        fscanf(fa, "%d%d%lf", &x, &y, &v);
        ar[i] = x;
        at[y]++;
        av[i] = v;

        fscanf(fb, "%d%d%lf", &x, &y, &v);
        vct.push_back(pair<pair<int, int>, double>(pair<int,int>(x, y), v));
    }

    fclose(fa);
    fclose(fb);

    sort(vct.begin(), vct.end());
    for (int i = 0; i < TIMES; i++) {
        bt[vct[i].first.first]++;
        bc[i] = vct[i].first.second;
        bv[i] = vct[i].second;
    }


    for (int i = 0; i < LIMIT; i++) {
        ac[i + 1] = ac[i] + at[i];
        br[i + 1] = br[i] + bt[i];
    }

    for (int i = 0; i < LIMIT; i++) {
        for (int j = ac[i]; j < ac[i + 1]; j++) {
            int r = ar[j];
            double va = av[j];

            for (int k = br[i]; k < br[i + 1]; k++) {
                int c = bc[k];
                double vb = bv[k];

```

```
                if (m[c].count(r) == 0) {
                    m[c][r] = va * vb;
                }
                else {
                    m[c][r] += va * vb;
                }
            }
        }
    }

    FILE *of = fopen("own.txt", "w");
    if (of == NULL) {
        fprintf(stderr, "Output file cannot open.\n");
        exit(EXIT_FAILURE);
    }

    for (int i = 0; i < LIMIT; i++) {
        for (map<int, double>::iterator it = m[i].begin(); it != m[i].end(); it++) {
            fprintf(of, "%d %d %.10f\n", it->first, i, it->second);
        }
    }
    fclose(of);
}
```

```
b00902064@linux14:/tmp2/b00902064/z> time ./own
Total time spent: 11.648429
159.048u 4.828s 2:46.91 98.1%    0+0k 80+6234040io 0pf+0w
```

[Sparse matrix multiplication – matlab code]

```
function main()

[fid message] = fopen(('ma.txt'), 'r');
if (fid == -1)
    disp(message);
end
ain = textscan(fid, '%f %f %f');

[fid message] = fopen(('mb.txt'), 'r');
if (fid == -1)
    disp(message);
end
bin = textscan(fid, '%f %f %f');

t = cputime; tic;

for i = 1 : size(ain{1}):
    ain{1}(1) = ain{1}(1) + 1;
    ain{2}(1) = ain{2}(1) + 1;

for i = 1 : size(bin{1}):
    bin{1}(1) = bin{1}(1) + 1;
    bin{2}(1) = bin{2}(1) + 1;

sa = sparse(ain{1}, ain{2}, ain{3}, 1000000, 1000000);
sb = sparse(bin{1}, bin{2}, bin{3}, 1000000, 1000000);

ans = sa * sb;

fprintf('CPU time used %f\n', cputime - t);
fprintf('Elapsed time used: %f\n', toc);
```

```
>> main()
CPU time used 13.820000
Elapsed time used: 14.088336
>>
```