Chemometrics and Intelligent Laboratory Systems 153 (2016) 106–109

Contents lists available at ScienceDirect

Chemometrics and Intelligent Laboratory Systems

journal homepage: www.elsevier.com/locate/chemolab

Short Communication

Parallel formula generator based on branch-and-bound algorithm for elucidating high resolution mass spectra

Mingjing Zhang, Zhimin Zhang *, Chen Chen, Hongmei Lu *, Yizeng Liang

College of Chemistry and Chemical Engineering, Central South University, Changsha 410083, PR China

ARTICLE INFO

Article history:
Received 5 November 2015
Received in revised form 1 March 2016
Accepted 2 March 2016
Available online 10 March 2016

Keywords:
Formula generator
Mass spectrometry
Parallel computing

ABSTRACT

The identification of unknown molecules by mass spectrometry is one of the most challenging problems despite the development of the instrument. One of the crucial steps is to obtain the possible elemental compositions within the limit of the measurement of mass-to-charge ratio and the mass tolerance. However, as the number of possible elements and the molecular weight increase, the more calculation time is needed. Here, a formula generator based on template metaprogramming and parallel computing is proposed to generate the possible candidate formulas. The template metaprogramming has been applied to replace the inefficient recursion to create the nested loops at compile-time for enumerating the possible elements. To accelerate the computation speed, the branch-and-bound algorithm is used to constrain the number of loop for each element. The parallel computing procedure is based on the Open Multi-Processing (OpenMP). The calculation time for calculating the candidates in the mass ranges especially for the higher ones can be significantly reduced when comparing with the popular HR2 program. PFG is implemented in C++ and available at https://github.com/zmzhang/PFG. It can be compiled easily and run smoothly in both Windows and Linux.

1. Introduction

In recent years, mass spectrometry is usually applied to identify unknown compounds, such as proteins and metabolites [1,2]. Mass spectrometers commonly measure the mass-to-charge ratio (m/z) of an ion. With the development of mass spectrometric techniques, the modern high resolution mass spectrometers with time-of-flight, Orbitrap or Fourier transform ion cyclotron resonance mass analyzers can obtain high mass accuracy. Based on the instrument and calibration, the mass tolerance can be <1 ppm [3]. However, the mass spectral libraries are limited by the fact that they do not contain all the substance, especially for LC–MS [4]. Thus, one of the crucial steps is to obtain the possible elemental compositions. And the potential formulas can be greatly filtered by some heuristic rules and the calculation of isotopic distribution [5–7]. In order to obtain the possible elemental compositions of the measurement of m/z, the process of generating candidates has been developed as a module in many packages or tools [5,8–14]. However, as the number of the elements and molecular weight increase, the more calculation time for the possible elemental compositions is needed. The calculation time for enumerating a high mass with more than 6 possible elements may cost several seconds, which can be time-consuming. In order to accelerate time-consuming chemometric algorithm, Zhang has introduced multicore-computing and proposed the parallel leave one out cross-validation [15].

Here, we present a parallel formula generator method (PFG) which is based on template metaprogramming and parallel computing to generate the possible candidate formulas within the given mass range. The template metaprogramming has been applied to create the nested loops at compile-time for enumerating the possible elements. To accelerate the computation speed, the branch-and-bound algorithm is used to constrain the number of loops for each element, and parallel computing based on the Open Multi-Processing (OpenMP) has been applied to improve the efficiency of the formula generator. Finally, to demonstrate the advantage of the PFG, it was compared with the HR2 approach.

2. Theory and methods

In this section, the characteristics of the parallel formula generator (PFG) will be described in detail. The mass of the ionization adduct is removed from the measured ion mass. The formulas are searched within the tolerance range from the calculated neutral molecular mass. The ionization adduct type, the types of the elements, the minimal and the maximal counts of each element are as the input variables.

2.1. Template metaprogramming for nested loops

Metaprogramming can be defined as the writing of programs which can manipulate other programs or themselves. Template metaprogramming is a metaprogramming technique which allows the compiler to generate codes at compile-time [16]. The construct

* Corresponding authors. Tel.: + 86 731 88830831.
E-mail addresses: zmzhang@csu.edu.cn (Z. Zhang), Hongmeilu@csu.edu.cn (H. Lu).

http://dx.doi.org/10.1016/j.chemolab.2016.03.002
0169-7439/

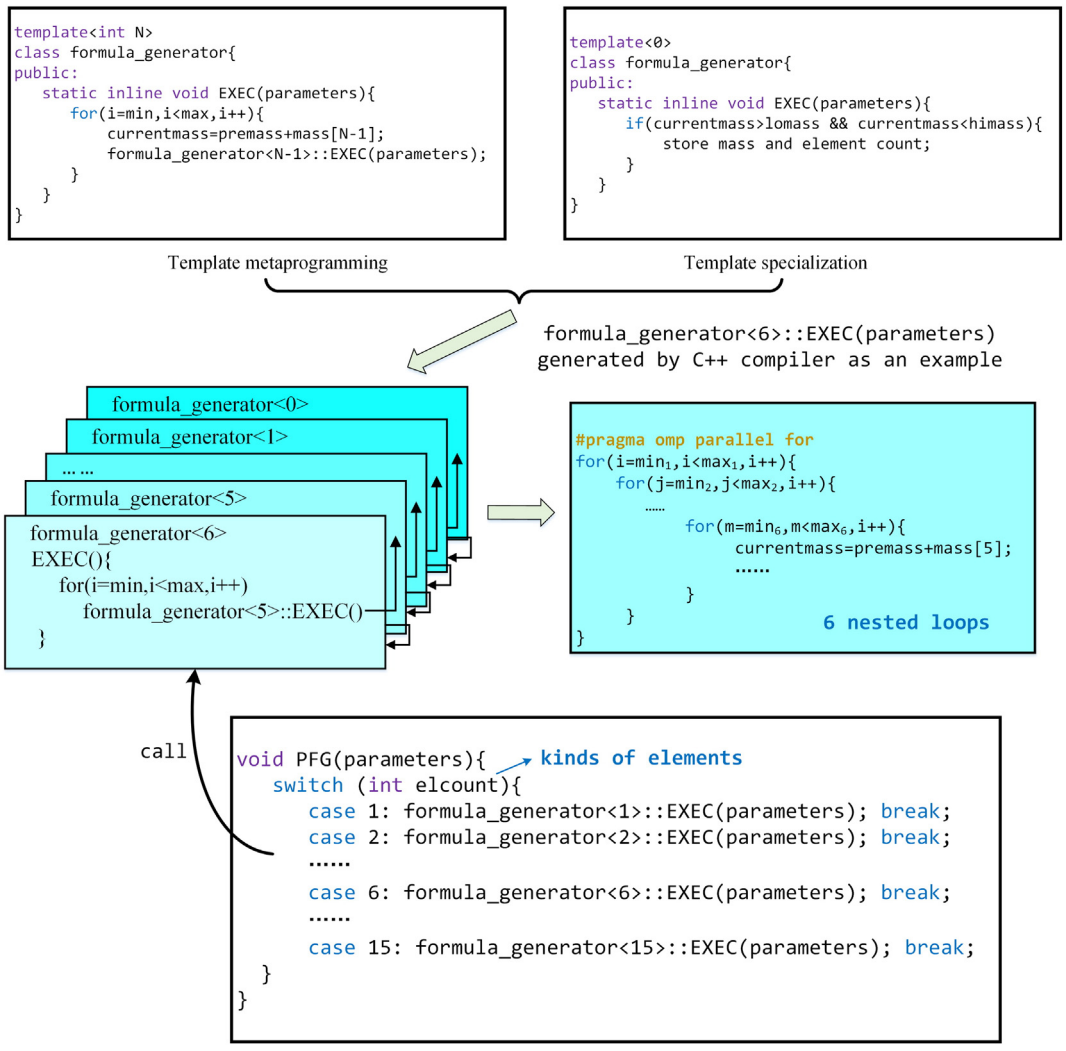

Fig. 1. The process of the template metaprogramming for the PFG.

of the template metaprogramming requires a template to be defined and a specialization template to make the defined template out of several alternatives. The nested loops generated by template metaprogramming in compile-time can be used to replace the inefficient recursion. The generated nested loops are used to enumerate the possible elemental compositions. The process of the template metaprogramming for generating nested loops in the PFG is shown in Fig. 1. Firstly, dozens of functions with different levels of nested loops have been generated by C++ compiler according to the template metaprogramming and specialization template. Based on the different number of elements, the switch statement of PFG function is used to call the function with right level of nested loops for enumerating the elements. An example with 6 kinds of elements is shown in Fig. 1. The C++ compiler generates 6 nested loops at compile-time for enumerating each element with one level loop. It can be seen from Fig. 1 that the formula_generator<6> performs the loop unrolling to produce 6 nested loops, and each level processes an element. The sixth level calls the fifth level until the bottom level. In each level, the nested loop can make the corresponding element looped from its minimum count to its maximum count. The formula_generator<0> is the specialization template function which is used to determine if the mass of the enumerated elemental composition is within the given mass range. Through all

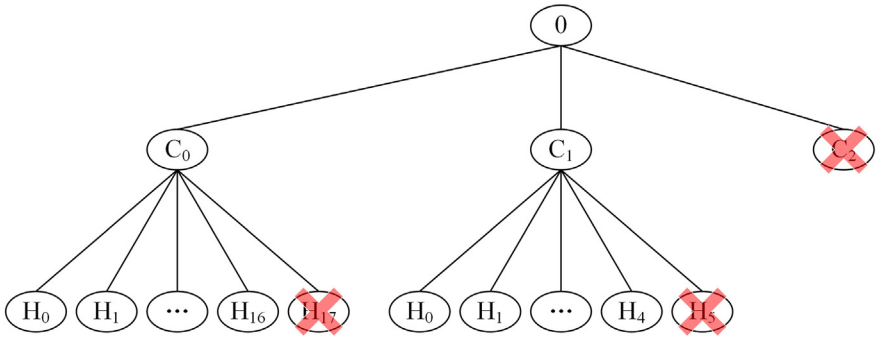

Fig. 2. The branch-and-bound algorithm for searching the mass of 16 with only carbon and hydrogen elements.