Automatic C Code Generation for Parallel Compilation

Amit Barve[1] & Brijendra Kumar Joshi[2]

[1]CSE, Malwa Institute of Technology, Indore, India
[2]MCTE Mhow, India
E-mail : barve.amit@gmail.com[1] , brijendrajoshi@yahoo.com[2]

Abstract – **This paper presents an Automatic C code generator which generates the syntactically correct code in c language, which is helpful for the performance evaluation and analysis of parallel programs.**

Keywords – Source Code Generator, Parallel Compilation.

I. Introduction

Observation of recent research scenario on parallel processing and compilation shows that the automatic source code generation is essential which generates bench mark programs to check the practical performance of variety of parallel compilation algorithms. An automatic source code generator is a tool which generates the source code based on given specifications. These tools can be developed using various methods like model-based code generation, simulation-model, etc. Detailed description of these models can be found in [1][2].

II. Automatic Source Code Generators

In the past, there have been some interesting attempts made to generate code automatically. Kirkland and Mecrer[3] presented an algorithm for automatic test pattern generation which produce tests for combinational circuits. Mukherjee and Chakrabarti [4] developed a translator which automatically translates an algorithm to specification. Xiang et al [5] developed an algorithm to generate code from flow chart. Apart from these there are many tools like Automatic C code generation in Hypersignal RIDE[6], Netbeans[7], IBM Rational Rose[8] and other IDEs. They normally generate the part of code based on specifications. But still there is lack of availability of tools which generate syntactically correct bench mark programs to check the parallel compilation. Barve and Joshi[9] developed a parallel lexical analyzer for multicore machines. Their work is based on detection of loops and decision constructs in the program for parallel lexical analysis. For such kind of program a tool is needed that would generate random number of programs with random number of lines of code to check the real time performance of algorithms. In this paper we present a tool which does the job of producing random number of programs as per specifications given by the user.

III. Code Generation Algorithm

Input: User choice of the program like program with loops, decision constructs or simple statements or the combination of all these, number of occurrences of choice and range of number of variables.

Output: A *.c* files with above specifications.

Method:

Create a .c file say *source.c* and open it in write mode and consider the following things to be written:

a) Call *select case* method for selecting the choices given by users.

b) Call *write code* method for writing number of statements in selected choice option.

IV. Experimental Results

Like Example 1 and Example 2. The experiment was performed on the above algorithm in Ubuntu 10.04 LTS on Sony Vaio Core I7 Laptop with 4GB RAM and Processor Speed 1.73GHz having 8 cores in total. The main screen in Figure 1. shows the possible choices available to the user for producing the source code. The choices can be *for, while, do..while, if..else, switch..case*. The code can be generated for single as well as combination of choices. Following are few examples are:

Example1:

```
void main()
{
    int t0,t1,t2;
    int x;
    for(x=0;x<5;x++)
    {
        t1=67;
        t0=68;
    }
    switch(x)
    {
        case 0 :
        {
        t1=76;
        break;
        }
    }
printf("Welcome to Automatic Source Code Generation");
printf("A Unique Tool for programmers ");
}
```

Example 2

```
void main()
{
    int t0,t1,t2;
    int x;
    for(x=0;x<5;x++)
    {
        t1=67;
        t0=68;
    }
    if(x<3)
    {
        t1=73;
        t0=74;
    }
    else
    {
        t2=75;
    }
    switch(x)
    {
        case 0 :
        {
            t1=76;
            break;
        }
    }
printf("Welcome to Automatic Source Code Generation");
printf("A Unique Tool for Programmers ");
}
```

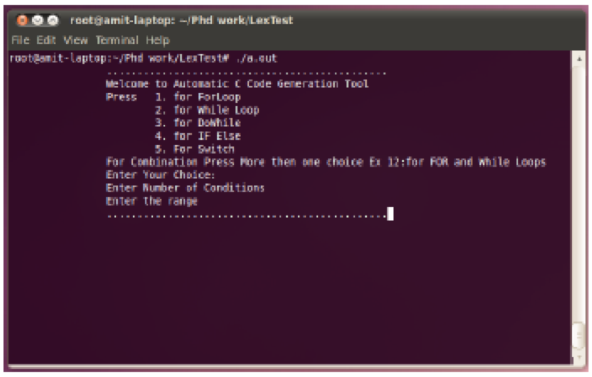
Like example 1 and 2 Table 1. Shows the code generated for 1- 512 constructs and total number of lines in the code.

Table 1: Code generated 1-512 constructs.

Sr. No.	Number of Constructs in Code	Total Line of Code
1	1	13
2	2	21
3	4	33
4	8	59
5	16	192
6	32	317
7	64	649
8	128	1348
9	256	2543
10	512	5635

V. CONCLUSION AND FUTURE WORK

Since bench mark programs are not commonly available for researchers to test the performance and analysis of algorithm, this tool is more helpful for research and academic purposes. The current version of tools generates only looping and decision making constructs with only assignment and print statements. In future it can be further modify to generate functions and pointers and other programming standards.


root@amit-laptop: ~/Phd work/LexTest
File Edit View Terminal Help

```
root@amit-laptop:~/Phd work/LexTest# ./a.out
                .............................................
                Welcome to Automatic C Code Generation Tool
                Press   1. for ForLoop
                        2. for While Loop
                        3. for DoWhile
                        4. for IF Else
                        5. For Switch
                For Combination Press More then one choice Ex 12:for FOR and While Loops
                Enter Your Choice:
                Enter Number of Conditions
                Enter the range
                .............................................
```

Fig. 1: Main screen of automatic c source code generator

VI. References

[1] http://www.ichmaschine.de/autocode.html

[2] http://www.mathworks.com/tagteam/ 36386_91391v00_Auto_Embedded_Code_Gen.pdf

[3] Tom Kirkland, M. Ray Mercer”; Algorithms For Automatic Test Pattern Generation;Volume 5 Issue 3, pp. 43-55, IEEE Computer Society Press Los Alamitos, CA, USA May 1988.

[4] Suvam Mukherjee, Tamal Chakrabarti. ”Automatic Algorithm Specification To Source Code Translation” Indian Journal Of Computer Science And Engineering (Ijcse), Vol. 2 No. 2 Apr-May 2011.

[5] Xiang-Hu Wu, Ming-Cheng Qu, Zhi-Qiang Liu, Jian-Zhong Li,” Research and Application of Code Automatic Generation Algorithm Based on Structured Flowchart” Journal of Software Engineering and Applications, 2011, 4, 534-545.

[6] www.ni.com/white-paper/2708/en.

[7] www.netbeans.org/

[8] www.ibm.com/software/awdtools/developer/rose/

[9] Amit Barve and Brijendra Kumar Joshi;”A Parallel Lexical Analyzer for Multi-core Machine”; Proceeding of CONSEG-2012,CSI 6th International confernece on software engineering;pp 319-323;5-7 September 2012 Indore,India.

◈◈◈