

実行起動前最適化層を有する自動チューニングソフトウェア構成方式の提案

片桐 孝洋^{†,††} 吉瀬 謙二^{†,††}
本多 弘樹[†] 弓場 敏嗣[†]

本発表では、インストール時、実行起動前、および実行時に応用プログラムのパラメタのチューニングを行う新しい自動チューニングソフトウェア構成方式である FIBER の提案を行う。また FIBER による最適化機能の一部について、PC クラスタを用いて評価した結果を示す。

Proposition of the Framework of Before Execution-Invocation Optimization Layer for Auto-tuning Software

TAKAHIRO KATAGIRI,^{†,††} KENJI KISE,^{†,††} HIROKI HONDA[†]
and TOSHITSUGU YUBA[†]

In this presentation, we propose a new auto-tuning framework for application programs, named FIBER – Framework of Installation, Before Execution-invocation, and Run-time optimization layers. We also evaluate the effect of a FIBER's optimization layer by using a PC cluster.

1. はじめに

近年、ATLAS¹⁾に代表される「自動チューニング機能付き」の数値計算ソフトウェア(以降、自動チューニング機能付きソフトウェア、Software with Auto Tuning Facilities (SATF) とよぶ)が多数開発されるようになった。ところが SATF の汎用性は、機能が限定されていることから問題があるといえる。

そこで我々は、SATF に幅広く適用できるソフトウェアの確立を目指し、(1) ライブラリインストール時に行うインストール時最適化階層 (Installation Optimization Layer, IOL)、(2) ユーザが指定するパラメタ(たとえば問題サイズなど)が固定した地点で行う実行起動前最適化階層 (Before Execution-invocation Optimization Layer, BEOL)、そして (3) 実際にライブラリが実行された地点で行う実行時最適化階層 (Run-time Optimization Layer, ROL)、の3種の最適化階層に分けて SATF を構成するソフトウェア構成方式を提案する。このソフトウェア構成方式のことを、FIBER (Framework of Installation, Before Execution-invocation, and Run-time optimization layers, ファイバー)²⁾ とよぶ。

2. FIBER のソフトウェア構成

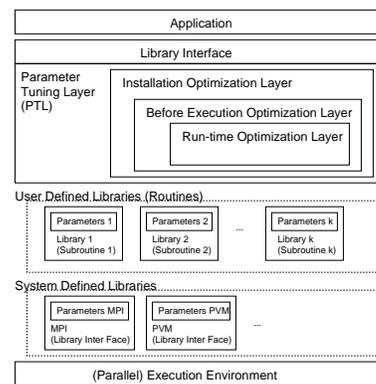


図 1 FIBER のソフトウェア構成

図 1 は FIBER のソフトウェア構成を示している。FIBER は以下に示す 2 種の機能を有するソフトウェア構成方式である：(1) ユーザが記述したライブラリやサブルーチン(またはプログラムの一部分)に専用言語などを用いて指示を与えることで、自動チューニングを保証するコードの自動生成、およびそのコードに関するパラメタ化と登録を自動に行う、コード開発支援機能；(2) パラメタチューニング階層 (Parameter Tuning Layer, PTL) におけるパラメタの最適化機能；PTL では引き渡されたパラメタの最適化に関して 3 階層を有するが、これらは最適化情報の伝達に関して図 2 のような処理手順がある。この主な目的は、下位層で決まるパラメタの精度を高めるためである。

[†] 電気通信大学大学院情報システム学研究所
Graduate School of Information Systems, The University of Electro-Communications

^{††} 科学技術振興事業団 戦略的創造研究推進事業 さきがけプログラム「情報基盤と利用環境」領域
“Information Infrastructure and Applications”, PRESTO, Japan Science and Technology Corporation (JST)

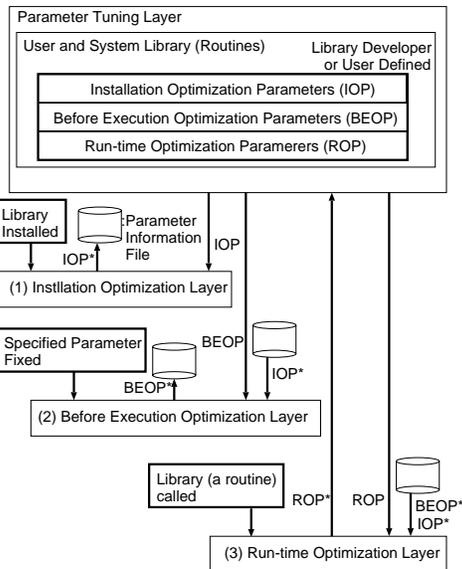


図 2 FIBER における 3 種の最適化階層の処理手順

次に図 2 の PTL では、各ライブラリやルーチンからユーザが指定したパラメタが引き渡されるが、それらは IOL に引き渡すパラメタ (IOP), BEOL に引き渡すパラメタ (BEOP), そして ROL に引き渡すパラメタ (ROP), というように指定されている。

各最適化階層によるパラメタ最適化は、以下に示す状況、かつ順序で実行される: [1 番目] IOL: ライブラリインストール時; [2 番目] BEOL: ユーザが指定するパラメタ (たとえば問題サイズなど) の指定後; [3 番目] ROL: BEOL によるパラメタ最適化が終了し、かつ対象のライブラリやルーチンの実行時;

3. FIBER による自動チューニングおよびユーザによるパラメタ指定方式

FIBER では、実行のために設定不可欠なパラメタを基本パラメタ (BP) とよぶ。また BP と性能とに影響するが、実行時に指定が不要なパラメタを性能パラメタ (PP) とよぶ。FIBER での自動チューニングは BP の一部を固定した上で、実行時間などを定義した関数 (コスト定義関数) が最小となる PP を推定する処理をさす。

以下にユーザによる PP の指定方式の例を示す。

```
!ABCLib$ static unroll(j) region start
!ABCLib$ varied (j) from 1 to 16
!ABCLib$ fitting polynomial 5
!ABCLib$ sampled (1-4,8,16)
do j=0, local_length_y-1
  tmpu1 = u_x(j)
  tmpu1 = mu * tmpu1 - y_k(j)
do i=0, local_length_x-1
  A(i_x+i, i_y+j) = A(i_x+i, i_y+j)
  + u_y(i)*tmpu1 - x_k(i)*tmpu1
```

```
enddo
enddo
!ABCLib$ static unroll(j) region end
```

上記の例では、`region start ~ region end` で囲まれた領域について、 j ループをアンローリングするコードを自動生成し、その段数をパラメタ化して PP とする。また実行起動前最適化 (static で指定) をする。その PP , すなわちここでは BEOP, の定義域は $\{1, \dots, 16\}$ である。コスト定義関数に線形多項式を利用し、 PP の標本点としては $\{1-4, 8, 16\}$ を指定する。

4. 実行起動前最適化の性能評価

例で示される固有値ソルバのカーネルにおいて、 PP である j ループのアンローリング段数パラメタを iud とする。ここでは PC クラスタを用いて、FIBER の実行起動前最適化機能を評価した結果をのせる。

PC クラスタのノードとして、Intel Pentium4 2.0GHz を 4 ノード用いた。ノード当たりの搭載メモリは 1GB (Direct RDRAM/ECC 256MB*4)、ネットワークカードは Intel EtherExpressPro100+ である。使用 OS は Linux 2.4.9-34 である。通信ライブラリは MPICH 1.2.1 を利用している。また PGI 社の Fortran90 コンパイラ 4.0-2, オプションは `-fast` を指定した。パラメタ推定の方法には、列選択ピボットングを用いた Householder QR 分解による最小二乗法を用いた。

本評価で想定する条件は、 $[iud \text{ の標本点 } 1]: \{1, 2, 3, 4, 8, 16\}$, $[問題サイズ n の標本点 (BP): $\{200, 400, 800, 2000, 4000, 6000, 8000\}$, $[コスト定義関数]: 5 次線形多項式, である。$$

ユーザによる指定次元 BP が 123, 1234, 9012 のとき、IOL による推定パラメタ iud とその実行時間は、14 (0.1286 秒), 6 (7.838 秒), 6 (973.6 秒) であった。これに対し、パラメタ全探索による FIBER の BEOP での推定結果は、10 (0.1269 秒), 10 (6.090 秒), 2 (845.6 秒) であり、従来のインストール時最適化のみの場合に対して、それぞれ、1.3%, 28.7%, 15.1% の速度改善が認められた。

現在、例に示したようなユーザによる自動チューニングの指定ができるプリプロセッサ ABCLibScript を開発中である。また当日は、FIBER 機能の一部を搭載した固有値ソルバのプロトタイプ ABCLibDRSSED リリース版 (<http://www.abc-lib.org/>にてソースコード公開中) のデモを行う予定である。

参考文献

- 1) ATLAS project;
<http://www.netlib.org/atlas/index.html>.
- 2) 片桐孝洋: プログラム、記録媒体およびコンピュータ、日本国特許出願、特願 2003-022792 (平成 15 年 1 月 30 日)。