

Richmond のモーメント法解析におけるベクトル化率の向上

東北大学大学院工学研究科

陳 強 澤谷 邦男

1. はじめに

Richmond のモーメント法[1]はアンテナや電磁界の数値解析によく用いられる手法である。この手法は分割セグメント数が少なくても計算精度がよいというすぐれた特徴[2]を持っているが、大型のアレーアンテナや平面アンテナ解析などの大規模な数値解析において CPU 時間が長いという問題点がある。そのため、大規模な数値解析に対して高速な計算を行えるアルゴリズムの開発が必要である。スーパーコンピュータはベクトル演算の機能[3]を持っており、ベクトル化率の高いプログラムに対して高速な計算が行える。本報告ではスーパーコンピュータのベクトル演算機能を利用して Richmond のモーメント法による数値解析を高速に行うためのベクトル化率の高いアルゴリズムを開発した結果を述べる。

2. 高いベクトル化率のアルゴリズムの開発

Richmond のモーメント法は、基底関数と重み関数として正弦状関数を用いた手法である。そのプログラムのフローチャートを図 1 に示す。まずアンテナの形状をダイポールセグメントに分割して、各セグメントの位置情報を計算しておく。その後位置情報に基づいて、各セグメント間の自己、相互インピーダンスを計算してインピーダンス行列を生成する。さらにインピーダンス行列の逆行列を求めることにより、アンテナ上の電流分布、入力インピーダンス、放射電磁界などを求めることができる[1]。

モーメント法による数値解析の一連の流れで、計算時間の大部分はダイポールセグメント間のインピーダンス行列を求めるための積分計算と、インピーダンス行列の逆行列計算で占められている。逆行列計算は東北大学大型計算機センターにスーパーコンピュータ用のベクトル化率の高いサブルーチンが用意されているので、それを用いることにより高速化を図ることができる。プログラム全体の計算時間の高速化を図るには、いかにインピーダンス行列を生成するサブルーチンに対してベクトル化率を向上させて、高速化するかが問題となる。

一般的にプログラムのベクトル化率を上げるにはプログラム中にベクトル演算対象となる DO ループ文を多くつくることと、DO ループの繰り返し数（ループ長）を大きくすることが考えられる。繰り返し数を大きくする方法としては

- ・多重 DO ループは最深の DO ループのループ長が大きくなるようにループを入れかえる。
- ・小さなループ長の DO ループは分離させる。
- ・多重 DO ループを一重の DO ループに変換する。

という手法がある。

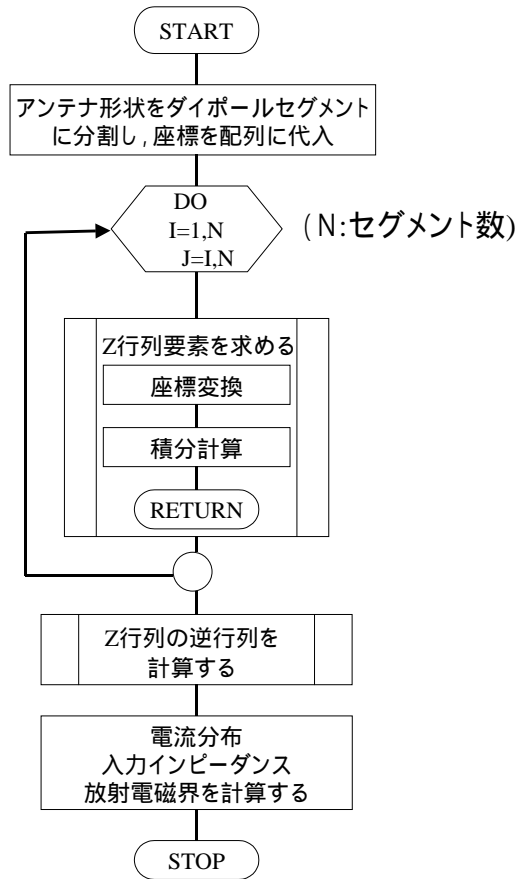


図 1 Richmond のモーメント法のフローチャート

図 2 に従来のモーメント法のアルゴリズムを示す。外側のループは分割セグメントの相互インピーダンスを計算するものであり、計算の規模が大きいくほど、ループ長が大きい。それに対して、内側のループはインピーダンス行列要素を計算する積分点数であり、4~6 ぐらいの短いものである。外側の大きなループはベクトル演算の対称とはならず、最深の小さなループに対してのみベクトル化されるのでプログラム全体のベクトル化率はわずか 10~30% である。そのため、ベクトル化率が上がるような新しいアルゴリズムの開発が必要である。そこで、図 3 に示すように、まず繰り返し数の小さい積分計算のループを分離し、積分計算に必要なセグメント間の位置情報をあらかじめ計算して配列に入れ、その後、ループ間の配列の受け渡しにより大きな繰り返し数のループ内で積分計算を行うようにした。さらに、積分計算のループの繰り返し数を大きくするために、二重ループを一重ループに変換した。これによりインピーダンス行列要素を求める部分のベクトル化率が 99% 以上に向上し、プログラム全体のベクトル化率も 99% 以上になった。

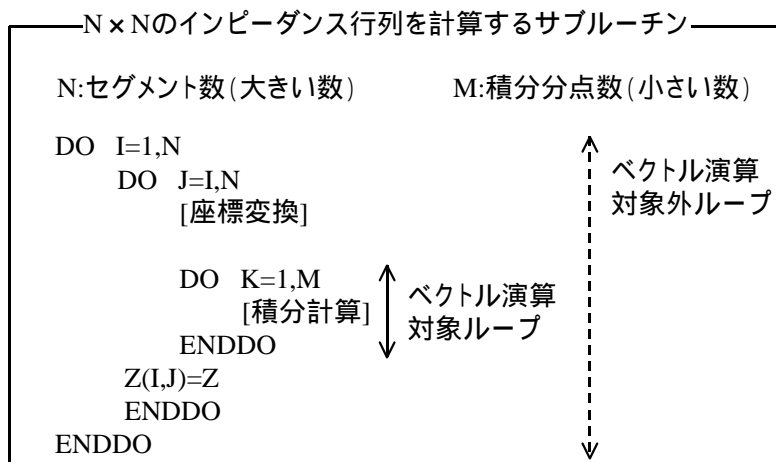


図 2 従来のアルゴリズム

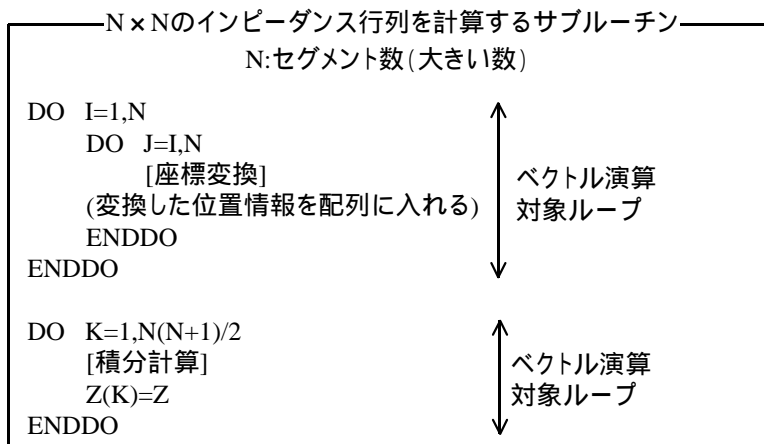


図 3 開発したアルゴリズム

分かりやすく説明するために，図 4と図 5に 9 行× 9 列のインピーダンス行列の計算例を示す．白い部分は計算されていない行列要素を表し，黒く塗られた部分は計算された行列要素を表す．従来のアルゴリズムにおけるインピーダンス行列計算では，図 4に示すように行

列の各要素を一つずつ計算している。一方，開発したアルゴリズムにおけるインピーダンス計算では，プログラムのベクトル化によって 10 倍高速になったとすると図 5 に示すように一命令で 10 個要素を計算できる。つまり，従来のアルゴリズム計算ではインピーダンス行列を計算するのに 15 命令必要だったのに，開発したアルゴリズムによる計算ではわずか 2 命令で計算できることがわかる。

Richmond のモーメント法のプログラムがベクトル化によって高速化を実現できるとすると，多くのセグメント数で分割された大型のアンテナ解析でもインピーダンス行列が少ないベクトル計算命令で求められ，モーメント法によるアンテナ解析の計算時間を大幅に減少することが期待できる。

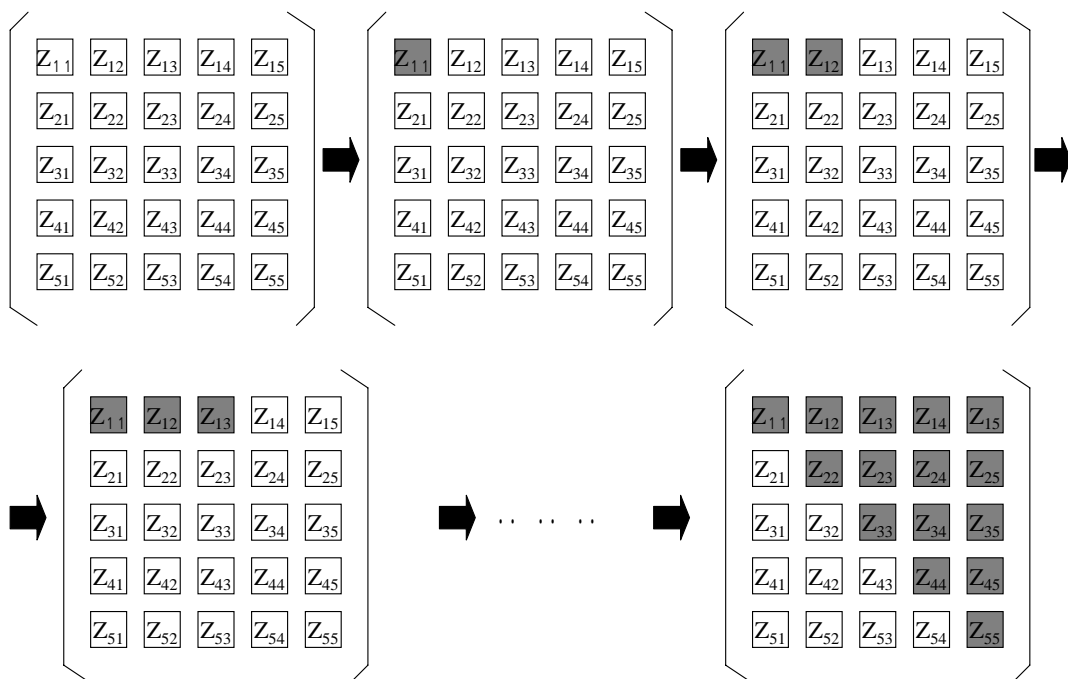


図 4 従来のアルゴリズムにおけるインピーダンス行列計算

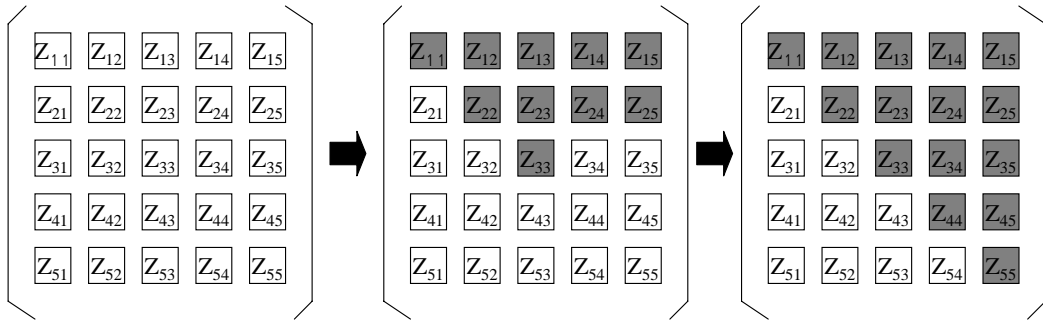


図 5 開発したアルゴリズムにおけるインピーダンス行列計算

3. 数値解析結果

従来のアルゴリズムのプログラムを WS と SX で実行し、計算時間の比較を行った。その結果を図 6 に示す。横軸がアンテナのダイポールセグメント分割数 N 、縦軸が CPU TIME [sec] を表す。図から WS の計算時間が SX の約 10 分の 1 であることがわかる。この結果から、SX を使用して計算の高速化を図るためには高いベクトル化率のプログラムが不可欠であることがわかる。

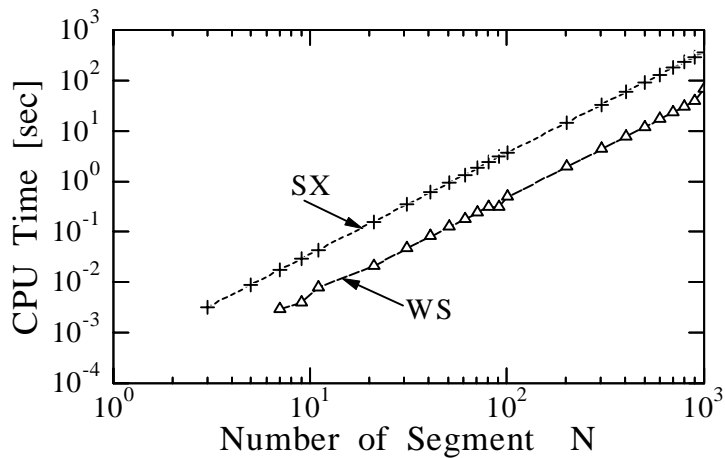


図 6 従来のアルゴリズムにおける WS と SX での計算時間比較
(WS:Alpha21164A-600MHz, SX:Super Computer SX4 in Tohoku University)

開発したアルゴリズムと従来のアルゴリズムのプログラムを SX で実行し、インピーダンス行列計算時間の比較を行った。その結果を図 7 に示す。横軸がアンテナのダイポールセグメント分割数 N 、縦軸が CPU TIME [sec] を表す。この図から、開発したアルゴリズムの方が従来のアルゴリズムより約 40 倍計算時間が高速化できたことがわかる。

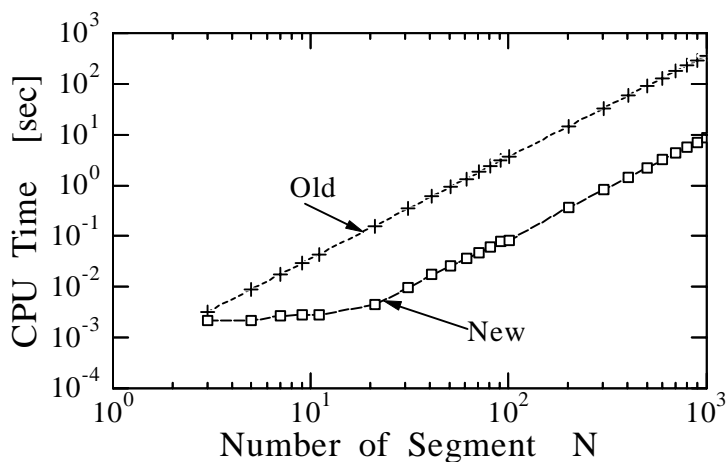


図 7 開発したアルゴリズムと従来のものによる
インピーダンス行列計算に要する CPU Time の比較
(Old : Old Algorithm , N=601 の時ベクトル化率 : 0.0% ,
New : New Algorithm , N=601 の時ベクトル化率 : 99.69% .)

次に開発したアルゴリズムと従来のアルゴリズムのプログラムを SX で実行し、逆行列計算時間を含めたプログラム全体の計算時間の比較を行った。その結果を図 8 に示す。横軸がアンテナのダイポールセグメント分割数 N、縦軸が CPU TIME [sec] を表す。この図から、開発したアルゴリズムの方が従来のアルゴリズムより約 50 倍計算時間が高速化できたことがわかる。

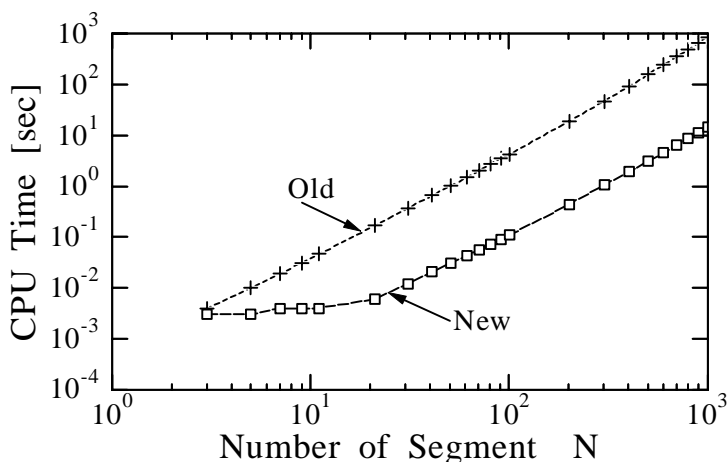


図 8 開発したアルゴリズムと従来のものによる MoM の CPU Time の比較
(Old : Old Algorithm , N=601 の時ベクトル化率 : 34.18% ,
New : New Algorithm , N=601 の時ベクトル化率 : 99.55%)

4. まとめ

Richmond のモーメント法におけるインピーダンス行列を計算するサブルーチンに対して高いベクトル化率で計算するためのアルゴリズムを開発した。その結果、ベクトル化率は99%以上になり、インピーダンス行列の計算時間で従来のアルゴリズムに対して約 40 倍高速になった。また、逆行行列計算時間も含むプログラム全体の計算時間でも約 50 倍高速になった。このような解析結果から、今回開発したアルゴリズムはスーパーコンピュータのハードウェア性能を十分に引き出せたものと考えられる。今後、スーパーコンピュータのハードウェア性能がさらに向上することによって、今回開発したアルゴリズムはさらに高速な計算を実現できるものと考えている。

謝辞

本文は東北大学大学院工学研究科電気・通信工学専攻古屋聡士氏（平成 12 年博士前期課程修了）が在学中に行った研究をまとめたものである。

本研究の一部は東北大学大型計算機センターのスーパーコンピュータを利用し、同センターとの共同研究で行われたものである。また、研究にあたっては同センターの有益な指導と多大な協力をいただいた。

参考文献

- [1] J. H. RICHMOND and N. H. GEARY, "Mutual Impedance of Nonplanar-Skew Sinusoidal Dipoles", *IEEE Trans. Antennas Propagat.*, Vol. AP-23, No. 3, pp. 412-414, May 1975.
- [2] 澤谷, "モーメント法によるアンテナ解析中級コース", 電子情報通信学会第 12 回アンテナ・伝搬における設計・解析手法ワークショップ, 1998.
- [3] 吉田, *FORTRAN90/SX の自動ベクトル化について*, 東北大学大型計算機センター広報, SENAC, Vol. 30, No. 4, pp. 27-42, 1997.