# Enhanced Automatic Multi-nested Loop Unimodular Parallelization with Array Reduction Technique

YU,Yijun, LI,Jianhui, HUANG,Bo, ZHU Chuan-Qi
Institute of Parallel Processing,
Shanghai 200433, China

## Abstract

Unimodular transformation is a kind of loop parallelizing transformations that keeps the normality of the transformed loop nest. However, there are two drawbacks that makes the UT method not practical. First, the dependence distance must be kept lexicographically positive after transformation, which makes the computation of unimodular matrix for multi-fold loops difficult. Secondly, the array reduction dependence is not constant distance in its nature while most UT method deal with constant distance matrix.

The paper aims at finding approaches to overcome these drawbacks.

First, an automatic UT approach for multi-fold loops with constant distance matrix is introduced to parallelize outer and inner multi-nested loops, which reaches the first aim.

Then, the paper defines a way to express array reduction dependence as mimimal constant distance vector such that the previous UT method can apply to the loops with array redution dependence, which fulfil the second aim.

To illustrate the theory of enhancing UT method with array reduction technique, a complete example is given.

**Keywords** Dependence Analysis, loop parallelization, unimodular transformation, array reduction

# 同归约变换结合的自动幺模变换技术

俞一峻 李剑慧 黄波 朱传琪
复旦大学并行处理研究所
上海 200433，中国

## 摘要

幺模变换是一类适合对循环并行化的循环变换，它能使规范循环仍变换为规范循环。应用自动幺模变换有两大困难，首先如何自动找出使多重循环并行化的恰当的幺模变换矩阵；其次如何解决妨碍幺模矩阵计算的非常数归约相关距离。本文首先介绍了如何对给定常数距离矩阵，自动找出使循环并行化的恰当的幺模变换矩阵的技术；本文然后提出了将数组归约相关表示为最小常数距离向量，从而使存在归约相关的多重循环也能够应用自动幺模变换技术，为自动幺模变换技术走向实用化提供了理论依据。

关键词：相关性分析，循环并行化变换，幺模变换，数组归约

## 1. 引言

有效的相关性分析**[1]**[**2**]和自动并行化变换方法**[8**]是并行化编译成功的关键。循环往往占据程序主要的计算时间，因此循环并行化变换自然成为并行化变换研究的重点。近年来人们发现了许多有效的循环变换手段**[1**][**3**][**4**][**7**]，其中一些基本循环变换(斜错(skewing) [**7**]、交换(interchange)[**6**] 、反序(reversal)等)都是幺模循环变换的特例。幺模变换具有保持整数循环迭代空间体积和维数不变的性质，不仅如此，步长为 1 的规范循环在幺模变换后仍是步长为 1 的规范循环，变换后程序书写格式仍较为规整。幺模变换是适合对循环并行化变换的一种重要手段。

但是，使用幺模变换进行循环自动并行化的困难在于如何对给定多重串行循环，自动找出使循环并行化的恰当的幺模变换矩阵。现有的大多数自动并行化编译器[9]没有实现一般的自动并行化幺模变换，就是因为无法对多重循环自动计算能使之并行化的幺模变换矩阵。而利用交互并行程序设计环境[**4**]提供的迭代空间相关图可视化工具[**5**]对循环程序的分析经

验表明：即使由人来寻找恰好能对循环并行化的幺模变换矩阵也是很困难的 。对给定多重串行循环，文**[9]**中提出了自动找出使循环并行化的恰当的幺模变换矩阵的算法。

除了缺乏自动计算么模矩阵的方法外，么模变换在过去的并行编译系统中所起作用不大的另一个主要原因是没能同归约识别技术相结合 。么模变换要求各相关都有确定的相关距离，但当循环中出现归约语句时，由于大多数归约引起的相关是杂乱无章的，根本不存在相关距离，致使现有的么模变换无法应用 。即使归约相关的相关距离确定的情况下，如果将归约所带来的相关等同于一般的相关，将使么模变换的有效性大为下降，以致产生出低效的并行代码。在**[9]**基础上我们提出有效地结合归约识别和多重循环么模变换的方法，可以开发出更大的并行度。

## 2. 基本概念

发生在两个程序段之间变量引用的数据相关性定义了相关程序段之间正确执行的时间偏序关系。在两个冲突的存储单元(程序变量)引用之间存在三种数据相关：先写后读的流相关，先读后写的反相关，先后都写的输出相关。一个循环迭代是循环体实例。发生在同一循环迭代内两个变量引用之间的相关称为 循环无关相关。发生在两个不同循环迭代内两个变量引用之间的相关称为跨循环相关。

数组归约相关是跨循环流相关的特例：假设 a 为变量(标量或数组元素), e 为表达式，若⊕为具有单位元且满足结合律及交换律的运算符，具有形式 $a = a \oplus e$ 的语句成为具有归约形式的语句，⊕为归约运算符, a 为归约变量, e 为归约表达式。循环 L 中具有归约形式的语句称为归约语句，若满足归约变量与不具有归约形式的语句及所有的归约表达式之间不存在跨循环流相关，而且同一归约变量的所有归约形式语句之间归约运算符相同 。若归约语句本身或归约语句之间具有跨循环 L 的相关，则称归约语句对循环 L 有效，否则称其对循环 L 无效。循环 L 中所有有效的归约语句构成的语句集，称为循环 L 的归约语句集。归约语句集中语句之间的相关，称为循环 L 的归约相关。

一个 n 重循环 L 的<u>规范迭代空间</u>是由各层循环的下标变量 $i_1,...,i_n$ 在给定上下界内所有取值向量所组成的集合

$$\{\mathbf{i}=(i_1,...,i_n)^T\in \mathbf{Z}^n \mid L_1\le i_1\le U_1, L_2(i_1)\le i_2\le U_2(i_1), ..., L_n(i_1,...,i_{n-1}) \le i_n\le U_n(i_1,...,i_{n-1})\},$$

串行循环的迭代空间是有序的（按照规范化迭代空间的<u>向量字典序</u>从小到大依次执行）。并行循环的迭代空间是无序的，因此，循环并行化变换后必须保证存在跨循环相关的两个循环迭代保持在原来规范迭代空间中串行执行顺序。

考察导致相关引用 $A(\mathbf{f}(\mathbf{I}))$和 $A(\mathbf{g}(\mathbf{I}))$的两组下标表达式 $\mathbf{f}(\mathbf{I})$和 $\mathbf{g}(\mathbf{I})$，如果 $A(\mathbf{f}(\mathbf{I}))$跨循环相关于 $A(\mathbf{g}(\mathbf{I}))$，那么在循环迭代空间 **I** 中必定存在两个迭代实例 $\mathbf{i_1}$ 和 $\mathbf{i_2}$，使得<u>相关方程</u>成立：$\mathbf{f}(\mathbf{i_1})=\mathbf{g}(\mathbf{i_2})$。<u>相关距离向量</u> **d** 定义为两个迭代向量 $\mathbf{i_1}$ 和 $\mathbf{i_2}$ 之间的差。迭代空间 I 中，$\mathbf{i_1}\angle\mathbf{i_2}$ 当且仅当**距离向量按字典序大于 0：$\mathbf{0}\angle\mathbf{d}$，如果按字典序大于 0 的距离向量称为 <u>合法距离向量</u>**。如果 $\mathbf{f}(\mathbf{i_2}-\mathbf{d})=\mathbf{g}(\mathbf{i_2})$对所有相关方程的解都成立，向量 $\mathbf{d} = \mathbf{i_2} - \mathbf{i_1}$ 是不依赖循环迭代变量 **i** 的常数向量，称为<u>常数相关距离</u>。同一个循环的一组合法常数相关向量组成<u>合法相关距离矩阵</u>。

<u>幺模矩阵</u> **U** 是行列式为±1 的整数方块矩阵。显然幺模矩阵的乘积、逆阵仍是幺模矩阵。对应于幺模矩阵 **U** 的<u>循环幺模变换</u> $\mathbf{U}:\mathbf{I}\rightarrow\mathbf{J}$ 是对原循环迭代空间 **I** 构造新的迭代空间 **J**:

$$\{\mathbf{j}=\mathbf{U}\mathbf{i}, \mathbf{i}=(i_1,...,i_n)^T\in \mathbf{Z}^n \mid L'_1\le j_1\le U'_1, L'_2(j_1)\le j_2\le U'_2(j_1), ..., L'_n(j_1,...,j_{n-1}) \le j_n\le U'_n(j_1,...,j_{n-1})\}$$

```
Loop1：DO i1=L1,U1,1
   ...
    Loopn: DO in =L(i1,...,in-1),U(i1,...,in-1),1
          Body(i1,...,in)
```

(a) 幺模变换前的循环

```
Loop1: DO j1=L'1,U'1, 1
   ...
    Loopn: DO in =L'(i1,...,in-1),U'(i1,...,in-1),1
          (i1,...,in)=(j1,...jn)U^-1
             Body(i1,...,in)
```

(b) 幺模变换后的循环

由于绝大多数循环使用线性边界表达式，可以将**循环(a)**边界条件表示为 $\mathbf{A}\mathbf{i}\le\mathbf{b}$ 的形式，

其中 $\mathbf{A}_{2n\times n}$ 是一个整系数矩阵，$\mathbf{b}_{2n\times 1}$ 是整系数向量：$\mathbf{L}_k = -\mathbf{b}_{2k} + \sum_{j=1}^{k-1}\mathbf{a}_{2k,j}\mathbf{i}_j$，$\mathbf{a}_{2k,k} = -1$，

$\mathbf{U}_k = \mathbf{b}_{2k} - \sum_{j=1}^{k-1} \mathbf{a}_{2k-1,j}\mathbf{i}_j$，$\mathbf{a}_{2k-1,k} = 1$，k=1,...,n。因为 **i=U$^{-1}$j**，所以在 J 空间中边界条件变为 **AU$^{-1}$j≤b**，用整数 Fourier-Motzkin 消元算法可以计算出 L'$_k$ 和 U'$_k$ 的线性表达式。从而得到循环**(b).**

对相关距离向量已知的循环，寻找合法的并行化变换一般有两种方法：侧重于使内层循环并行的 wavefront 方法和侧重于使最外层循环并行的 partition 方法。[9]探讨的 n 重循环幺模并行化算法是对距离矩阵 $\mathbf{D}_{nxm}$ 发掘最多可并行外层循环的幺模变换: 使得 n-rank(**D**)个外层循环一定能够并行化(定理 1)，而且 rank(**D**)-1 个内层循环也能并行化(定理 2)。

定理 1(合法的消元幺模矩阵存在性)：在 n 维迭代空间中，只要所有 m 个相关距离向量都是常数，给定的合法距离矩阵 $\mathbf{D}_{m\times n}$ 的秩 r<n，一定存在一个合法的行变换幺模矩阵 $\mathbf{U}_{nxn}$，使得 $\mathbf{UD}=[\mathbf{0}_{m-rxn} \mid \mathbf{D}_{rxm}'^{T}]^{T}$，**D'**是合法距离向量矩阵。

定理 2：在 n 维迭代空间中，给定距离向量矩阵 $\mathbf{D}_{n\times m}$ 的秩 rank(**D**)=n，一定存在一个合法的行变换幺模矩阵 $\mathbf{U}_{nxn}$，使得 **UD 是**合法距离向量矩阵且所有距离向量的第 1 个元素大于零(**UD** 的首行元素大于 0)。

如果 **D** 为满秩可逆矩阵，行列式|**D**|>1，还可以用 E.H.D'Hollander[2]的迭代划分和求代表迭代点的方法，通过增加一个外层并行循环，使所有内层串行循环的相关迭代链是紧凑的。

# 3. 结合归约变换增强么模并行化变换

复杂的数组归约识别是并行化研究的另一项实用技术，将它同么模变换理论相结合，具有理论和实践的重要意义。因为对归约语句的并行化处理可以采用特定的技术，所以在进行么模变换之前，可以不考虑归约引起的数据相关，这就扩展了么模变换技术的适用范围。若一个循环除了归约相关外没有跨循环相关，则可通过归约变形实现循环的并行化；若存在其它跨循环相关，可以考虑引入么模变换来消除这种相关，以实现并行化。

## 3.1 归约变换

前面我们谈到归约语句具有一定的并行性，在并行执行循环的不同迭代时只要保证归约语句不被同时执行，就可以保证结果的正确性。现在我们介绍两种使含归约的循环并行化的归约变换方法。

第一种方法就是让相应循环直接并行化，仅对归约单句加锁使归约变量的操作互斥即可。为保证最大并行性，我们只需对同一组归约语句，即变量名和操作符相同的归约语句，加相同的锁，对不同组的归约语句加不同的锁。证明：S1(i),...,Sk(i)为 k 维数组 A 的线性下标表达式，若相关语句 A(S1(i),...,Sk(i))= A(S1(i),...,Sk(i))⊕e(i)引起跨循环流相关，则相关联立方程为：Fi1=Fi2。因为 d=i2- **i1，所以 Fd=F(i2- i1)=0，又因为是跨循环流相关，所以按词典序 d>0，且 d 最小；若 d 不唯一，即存在两个最小的距离向量 d1，d2，那么对于某些迭代相关距离为 d1,** 对于另一些迭代相关距离为 d2，这与常数相关距离的要求矛盾。证毕。

表 1a 描述了这种方法。

加锁操作需要并行语言的支持，但有些并行计算机的并行语言不支持锁操作，这时我们需采用变量扩张技术。这种技术对每一组归约，都为每个并行进程分别引入的一个临时数组，该临时数组每个元素的初值为归约运算的单位元；然后各进程完全并行执行，并将每组归约的部分结果记录在各自的临时数组元素内；在并行循环结束后，再通过每个的临时数组的部分结果求出归约语句的最终结果。应该注意的是，变量扩张方法在引入临时数组前需要计算出临时数组的大小，当归约变量是数组而我们又无法确切计算出临时数组的大小时，我们可以根据整个归约数组的大小来进行数组扩张。证明：S1(i),...,Sk(i)为 k 维数组 A 的线性下标表达式，若相关语句 A(S1(i),...,Sk(i))= A(S1(i),...,Sk(i))⊕e(i)引起跨循环流相关，则相关联立方程为：Fi1=Fi2。因为 d=i2- **i1，所以 Fd=F(i2- i1)=0，又因为是跨循环流相关，所以按词典序 d>0，且 d 最小；若 d 不唯一，即存在两个最小的距离向量 d1，d2，那么对于某些迭代相关距离为 d1,** 对于另一些迭代相关距离为 d2，这与常数相关距离的要求矛盾。证毕。

表 1b 描述了这种方法：其归约运算符为 +，归约运算的初值为 0，数组的归约区域为

A[1:10]。

### 3.2 结合归约识别技术增强么模 变换

所有的并行变换要保证每一相关在变换下的不变性，也就是说，假设存在数据相关 $S_1 \angle S_2$，那么变换后 $S_1$ 必在 $S_2$ 之前执行，但对归约相关不必保持这种不变性，因为按任意顺序执行归约语句都不会改变最终结果 。所以，我们在进行么模变换时可以忽略并行化外层循环的归约相关，但必须同时保证归约的不变性，即归约语句在么模变换后仍然是归约语句，并称这种么模变换为增强么模变换 。增强么模变换时要保证变换后归约语句不被同时执行，只要对变换后的程序进行归约并行化即可 。下面我们通过一个定理证明增强么模变换保证归约语句的不变性，从而保证变换的合法性。

引理 1：对任意一层循环，该循环中具有归约形式的单句经增强么模变换仍为具有归约形式的单句，反之亦然。

引理 2（归约相关计算）：在 n 重循环中，相关语句 $A(S_1(\mathbf{i}),...,S_k(\mathbf{i}))= A(S_1(\mathbf{i}),...,S_k(\mathbf{i}))\oplus e(\mathbf{i})$ 产生的归约常数相关距离 $\mathbf{d}$ 为满足齐次方程 $\mathbf{Fd=0}$ 唯一的最小非零正向量。其中 $\mathbf{i}=(i_1,..., i_n)$ 为循环的迭代下标变量，$S_1(\mathbf{i}),...,S_k(\mathbf{i})$为 k 维数组 A 的线性下标表达式，$e(\mathbf{i})$为不含数组 A 引用的任意归约表达式。$\mathbf{F}$ 为一个 $k\times n$ 阶矩阵，其第 l 行向量由 $S_l(\mathbf{i})=\sum_{m=1}^{n}\mathbf{f}_{lm}\mathbf{i}_m$ 定义。

证明：$S_1(\mathbf{i}),...,S_k(\mathbf{i})$为 k 维数组 A 的线性下标表达式 ，若相关语句 $A(S_1(\mathbf{i}),...,S_k(\mathbf{i}))= A(S_1(\mathbf{i}),...,S_k(\mathbf{i}))\oplus e(\mathbf{i})$引起跨循环流相关，则相关联立方程为：$\mathbf{Fi_1=Fi_2}$。因为 $\mathbf{d=i_2-\ i_1}$，所以 $\mathbf{Fd=F(i_2-\ i_1)=0}$，又因为是跨循环流相关，所以按词典序 $\mathbf{d>0}$，且 $\mathbf{d}$ 最小；若 $\mathbf{d}$ 不唯一，即存在两个最小的距离向量 $\mathbf{d}_1$，$\mathbf{d}_2$，那么对于某些迭代相关距离为 $\mathbf{d}_1$, 对于另一些迭代相关距离为 $\mathbf{d}_2$，这与常数相关距离的要求矛盾。证毕。

表 1、归约变换

```
DO I=1,9
    Critical_section_A
    A(I)=A(I)+F(I)
    end section
    Critical_section_A
    A(I+1)=A(I+1)+G(I)
    end section
ENDDO
```

a 利用加锁进行归约变换

```
REAL FX1(1:10,1:9)
DOALL   I=1,p,1        /*p 为处理机个数 */
    DOI1=1,10
        A1(I1,I)=0   /*0 为"+"的单位元*/
    ENDDO
    DO J=⌊9/p*(I-1)⌋+1,⌊9/p*I⌋,1
        A1(J,I)=A1(J,I)+F(J)
        A1(J+1,I)=A1(J+1,I)+G(J)
    ENDDO
ENDDO
DO I=1,p,1
    DO I1=1,10,1
        A(I)=A1(I1,I)+A(I)
    ENDDO
ENDDO
```

b 利用数组扩张来进行归约变换

例如，在下述的二重循环中

```
DO I1=0,10,1
   DO I2=0,10,1
S1:    A(I1+I2)=A(I1+I2)+1
S2:    B=B+1
S3:    C(5)=C(5)+1
   ENDDO
ENDDO
```

归约相关语句 $S_1$ 产生跨循环流相关，则相关距离向量 **d**=$(d_1,d_2)^T$ 可以这样计算：

$$\begin{pmatrix}1 & 1\end{pmatrix}\begin{pmatrix}d_1\\ d_2\end{pmatrix}=\begin{pmatrix}0\\ 0\end{pmatrix}，d_1>0 \text{ 或 } d_1=0 \text{ 且 } d_2>0。$$

显然，取 $\mathbf{d}=\begin{pmatrix}1\\ -1\end{pmatrix}$ 为满足上式唯一的最小合法距离向量。在归约语句 $S_2$，$S_3$ 产生跨循环流相关，但是，由于不存在唯一的最小合法距离向量，无法计算常数相关距离。

对于能够计算常数距离向量的归约相关，幺模变换对归约相关距离向量的改变可以由下列定理计算。

定理 3（幺模变换后的归约相关计算）：n 重循环中的归约语句 S: $A(S_1(\mathbf{i}),\ldots,S_k(\mathbf{i}))=A(S_1(\mathbf{i}),\ldots,S_k(\mathbf{i}))\oplus e(\mathbf{i})$如果产生跨循环流相关 **d**, 在幺模变换 **U** 后归约语句 S'产生跨循环流相关 **d'=Ud**。

证明：因为幺模变换 **U** 后，$\mathbf{j}=\mathbf{U}\mathbf{i}$, $\mathbf{i}=\mathbf{U}^{-1}\mathbf{j}$。所以归约语句 S'为:

$$A(S_1(\mathbf{U}^{-1}\mathbf{j}),...,S_k(\mathbf{U}^{-1}\mathbf{j}))= A(S_1(\mathbf{U}^{-1}\mathbf{j}),...,S_k(\mathbf{U}^{-1}\mathbf{j}))\oplus e(\mathbf{U}^{-1}\mathbf{j})$$

原相关联立方程 $\mathbf{Fi}_1=\mathbf{Fi}_2$ 变换为为相关联立方程：$\mathbf{F\,U}^{-1}\mathbf{j}_1=\mathbf{F\,U}^{-1}\mathbf{j}_2$，所以：

$\mathbf{d}'= \mathbf{j}_2- \mathbf{j}_1= \mathbf{U}^{-1}\mathbf{i}_2- \mathbf{U}^{-1}\mathbf{i}_1=\mathbf{U}^{-1}(\mathbf{i}_2- \mathbf{i}_1)= \mathbf{U}^{-1}\mathbf{d}$。□

如果归约相关的距离可以计算，下面给出对定理 2 的改进算法，以便适当选择内层么模变换矩阵使用增强么模变换，也可以对内层循环消除归约相关。

算法 1: 寻找消除常数相关距离的内层归约相关的内层并行化么模矩阵 U：

输入：在 n 维迭代空间中，给定常数相关距离矩阵 $\mathbf{D}_{n\times(m+r)}$的秩为 n，其中 $m\geq n, r\geq 0$,分别是非归约相关距离向量和归约相关距离向量数。

输出：找到这样一个合法的行变换幺模矩阵 **U**，使得循环内层可并行且完全消除常数相关距离的归约相关。

步骤：

1) 令 k=1, $u_{n-1}=\max\{0, \left\lceil \dfrac{-\mathbf{d}_{n,j}}{\mathbf{d}_{n-1,j}} \right\rceil \mid 1\leq j\leq m+r, d_{1j}=...=d_{n-2\,j}=0, d_{n-1\,j}>0\}$

,....,

n-1) 令 k=n-1, $u_1=\max\{\left\lceil \dfrac{-\sum_{k=1}^{n-2}\mathbf{u}_{n-k}\mathbf{d}_{n-k,j}-\mathbf{d}_{n,j}}{\mathbf{d}_{1,j}} \right\rceil \mid 1\leq j\leq m+r, d_{1j}>0\}$。

n) 这些非负整数 $u_1,...,u_{n-1}$，满足 $\mathbf{d}_{n,j}+\sum_{i=1}^{n-1}\mathbf{u}_i\mathbf{d}_{i,j}>0 \mid 1\leq j\leq m+r$，所以幺模矩阵

$\mathbf{U}=\begin{pmatrix} \mathbf{u}_1 & ... & \mathbf{u}_{n-1} & 1 \\ 0 & 0 & 1 & 0 \\ 0 & \iddots & \iddots & \vdots \\ 1 & 0 & \cdots & 0 \end{pmatrix}$即为所求。□

### 3.3 对归约相关增强么模并行化的算法总结

设任意 n 重嵌套循环相关距离向量矩阵 $\mathbf{D}=[\mathbf{D}_1;\mathbf{D}_2]$，其中 $\mathbf{D}_1$ 和 $\mathbf{D}_2$ 分别是非归约相关距离向量矩阵和归约相关距离向量矩阵。

忽略外层循环的归约相关，我们首先由定理 1 对非归约相关距离向量矩阵 $\mathbf{D}_1$ 作幺模 $\mathbf{U}_1$ 变换得到 $\mathbf{U}_1\mathbf{D}_1=[\mathbf{0}^T;\mathbf{D}_1'^T]^T$，$\mathbf{D}_1'$为 $s=\mathrm{rank}(\mathbf{D}_1)$阶行满秩阵，现在根据算法 1对包含归约相关

距离向量的矩阵 $\mathbf{D'}=\mathbf{U_1D}$ 计算得到的 $\mathbf{U}_s$ 作 $\mathbf{U}_2=\begin{pmatrix}\mathbf{I}_{(n-s)x(n-s)} & \mathbf{0}_{(n-s)xs}\\ \mathbf{0}_{sx(n-s)} & \mathbf{U}_{sxs}\end{pmatrix}$，则 $\mathbf{U_2U_1D}$ 完成进一步对内层循环并行化的任务。若么模变换之后外层可并行化，则检验所有的归约语句，是否对外层有效(存在跨循环流相关)，有效的归约变量需做归约变形，若都无效，则外层可直接并行化。

# 4. 实例

表 2为一个消除归约的多重循环么模变换的实例，表 2a 为一个三层循环，循环中第二句：b(i1+i2,5*i1+i3)= b(i1+i2,5*i1+i3)+i1 为外层循环的归约语句：它引起的归约相关齐次方程如下：

$\begin{pmatrix}1 & 1 & 0\\ 5 & 0 & 1\end{pmatrix}\begin{pmatrix}d_1\\ d_2\\ d_3\end{pmatrix}=\begin{pmatrix}0\\0\\0\end{pmatrix}$，如果 $d_1=0$，则 $d_2=d_3=0$，但 $\mathbf{d}=\begin{pmatrix}0\\0\\0\end{pmatrix}$不是跨循环相关距离，所以 $d_1>0$。要使 $\mathbf{d}$ 取最小值(词典序意义上)，必须 $d_1=1$。从而 $\mathbf{d}=\begin{pmatrix}1\\-1\\-5\end{pmatrix}$为所求的归约相关距离向量。类似地可以计算数组 c 规约相关距离向量，不再赘述。

我们在外层么模并行化变换之前忽略了它的影响，根据定理 1 求得幺模矩阵 $\mathbf{U_1}$，使表 2b 循环外层并行化($\mathbf{U_1D}$ 首行为 0)。内层使用定理 2 并行化（表 2c），但是得到幺模矩阵 $\mathbf{U_2}$ 不能消除由第二句引起的归约相关，如表 2c 所示。在采用了消除内层归约算法 1之后，么模矩阵将修改为 $\mathbf{U_3}$，我们可以从表 2d 中看到，内层归约相关现象被完全消除。我们注意到，在牺牲部分并行度的情况下实现了归约消除。最后根据 Fourier-Motzkin 算法求得循环边界表达式，完成了增强幺模变换(表 2bcd 右栏)。

表 2、增强么模变换

| 计算说明 | 程序 |
| --- | --- |

a)根据引理 2，计算包括 b,c 归约相关在内，所有跨循环相关：

$$D=\begin{array}{c}\begin{pmatrix}1 & 0 & 1 & 1\\ 2 & 3 & -1 & 0\\ -3 & 2 & -5 & 0\end{pmatrix}\\ \begin{matrix}a_1 & a_2 & b & c\end{matrix}\end{array}$$

```
   DO i1=1,10
      DO i2=1,10
         DO i3=1,10
            a(i1+1,i2+3,i3+2)=a(i1,i2+1,i3+5)+a(i1+1,i2,i3)
            b(i1+i2,5*i1+i3)= b(i1+i2,5*i1+i3)+i1
            c(i2,i3)=c(i2,i3)+(i2+i3)
         ENDDO
      ENDDO
   ENDDO
```

b)忽略 b,c 引起的归约相关：

$$D_1=\begin{pmatrix}1 & 0\\ 2 & 3\\ -3 & 2\end{pmatrix}$$ 进行外层并行化么模变换

$$U_1=\begin{pmatrix}13 & -2 & 3\\ 1 & 0 & 0\\ -4 & 1 & -1\end{pmatrix},$$

$$D'_1=\begin{pmatrix}0 & 0\\ 1 & 0\\ 1 & 1\end{pmatrix}$$

```
  DOALL j1=-4,158
     DO j2=max(1,ceil((j1-28)/13.)),min(10,floor((17+j1)/13.))
        DO j3=max(ceil((1-j1+j2)/3.),ceil((1-j1+5*j2)/2.)),
1             min(floor((10-j1+j2)/3.),floor((10-j1+5*j2)/2.))
           i1=j2
           i2=j1-j2+3*j3
           i3=j1-5*j2+2*j3
           a(i1+1,i2+3,i3+2)=a(i1,i2+1,i3+5)+a(i1+1,i2,i3)
           Critical_Section_B
           b(i1+i2,5*i1+i3)= b(i1+i2,5*i1+i3)+I1
           End_ Section
           Critical_Section_C
           c(i2,i3)=c(i2,i3)+(i2+i3)
           End_Section
        ENDDO
     ENDDO
  ENDDOALL
```

c)忽略归约相关，
进行内层并行化么模变换

$$U_2=\begin{pmatrix}1 & 0 & 0\\ 0 & 0 & 1\\ 0 & 1 & 0\end{pmatrix},$$

$$U=U_2U_1=\begin{pmatrix}13 & -2 & 3\\ -4 & 1 & -1\\ 1 & 0 & 0\end{pmatrix},$$

$$D'=\begin{array}{c}\begin{pmatrix}0 & 0 & 0 & 13\\ 1 & 1 & 0 & -4\\ 1 & 0 & 1 & 1\end{pmatrix}\\ \begin{matrix}a_1 & a_2 & b & c\end{matrix}\end{array}$$ 。最内层循环对 b 的归约相关仍旧存在。

```
   DOALL j1=-4,158
      DO j2= max(ceil((2-j1)/3.),ceil((6-j1)/2.),ceil((-5-4*j1)/13.)),
1        min(floor((20-j1)/3.),floor((60-j1)/2.), floor((49-4*j1)/13.))
         DOALL j3=max(1,j1+j2-10,ceil((j1+2*j2-10)/2.)),
1                  min(10,j1+j2-1,floor((j1+2*j2-1)/5.))
            i1=j3
            i2=j1+3*j2-j3
            i3=j1+2*j2-5*j3
            a(i1+1,i2+3,i3+2)=a(i1,i2+1,i3+5)+a(i1+1,i2,i3)
            Critical_Section_B
            b(i1+i2,5*i1+i3)= b(i1+i2,5*i1+i3)+i1
            End_section
            Critical_Section_C
            c(i2,i3)=c(i2,i3)+(i2+i3)
            End _Section
         ENDDOALL
      ENDDO
   ENDDOALL
```

d)消除内层 b 的归约相关的么模变换(增加一次斜错变换）：

$$U_3=\begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 1\\ 0 & 1 & 0\end{pmatrix}$$

$$U=U_3U_2U_1=\begin{pmatrix}13 & -2 & 3\\ -3 & 1 & -1\\ 1 & 0 & 0\end{pmatrix},$$

$$D'=\begin{array}{c}\begin{pmatrix}0 & 0 & 0 & 13\\ 2 & 1 & 1 & -3\\ 1 & 0 & 1 & 1\end{pmatrix}\\ \begin{matrix}a_1 & a_2 & b & c\end{matrix}\end{array}$$ 。

所以还要对 c 归约变形

```
  DOALL j1=-4,158
     DO j2=max(ceil((5-j1)/3.),ceil((-3*j1-3)/13.),ceil((8-j1)/2.)),
1       min(floor((50-j1)/3.),floor((80-j1)/2.), floor((66-3*j1)/13.))
        DOALL j3=max(1,ceil((j1+3*j2-10)/4.),ceil((j1+2*j2-10)/7.)),
1             min(10,floor((j1+3*j2-1)/4.), floor((j1+2*j2-1)/7.))
           i1=j3
           i2=j1+3*j2-4*j3
           i3=j1+2*j2-7*j3
           a(i1+1,i2+3,i3+2)=a(i1,i2+1,i3+5)+a(i1+1,i2,i3)
           b(i1+i2,5*i1+i3)= b(i1+i2,5*i1+i3)+i1
           Critical_Section_C
           c(i2,i3)=c(i2,i3)+(i2+i3)
           End_Section
        ENDDOALL
     ENDDO
  ENDDOALL
```

## 5. 结论

本文提出了一种多重串行循环并行化的幺模变换方法，不仅从理论上证明满足外层并行化要求的合法幺模矩阵是存在的，而且给出了计算并行化幺模变换矩阵的可行算法，并提出了将多重循环幺模并行化变换与归约变换相结合的有效途径，由于所有算法都是自动的，可以在自动并行化变换工具中实现，为自动幺模变换技术走向实用化提供了理论依据。

## 参考文献


[1] Banerjee,U., "Unimodular transformations of double loops",in Advances in Languages and Compilers for Parallel Processing, pp.30-44,Cambridge, MA: MIT Press, 1991.

[2] Banerjee,U., *Dependence Analysis for Supercomputing*, Kluwer Academic Publishers, 1988.

[3] D'Hollander,E.H., "Partitioning and labeling of loops by unimodular transformations", IEEE Trans. on Parallel and Distributed Systems, 3(4):465-476,1992.

[4] Wang,Q., Yu,Y., D'Hollander,E.H., "Visualizing the Iteration Space in PEFPT",in Proc. High Perf. Computing and Network Intl. Conf. and Exhibition,Viena, Austria, pp908-915, Apr. 1997.

[5] Wang,Q., Yu,Y., D'Hollander,E.H., "Interactive Programming using PEFPT", Proc. Seminar Parallel Computing: Software,Architecture and Operating Systems, pp123-129,Oct. 1996.

[6] Wolfe,M., "Advanced Loop Interchange",Proc. 1986 Int'l Conf. on Parallel Processing,pp536-543.

[7] Wolfe,M., "Loop Skewing: The Wavefront Method Revisited", Proc. 1986 Int'l Journal on Parallel Programming, 15(4):279-293.

[8] Wolfe,M., *High Performance Compilers for Parallel Computing*, Addison-Wesley Publishing Company, 1996.

[9] 俞一峻，施武，臧斌宇，朱传琪，"自动寻找使多重串行循环并行化的幺模变换"，软件学报, No. 3, 1999.

[10] 朱传琪，臧斌宇，陈彤，"自动并行化编译工具 AFT ".软件学报, Mar 1996, Vol.7, No.3, pp180-186.