

地震作业数据处理并行化的几个问题

赵凤治 (中国科学院计算中心)

我们知道,地震数据处理工作模块的工作是按如下步骤进行的。它们必须在地震数据作业子操作系统的控制下运行。一个处理过程往往要处理数以千记的记录道;另一方面,同一道记录又要用很多个工作模块做处理。这些工作模块构成一个作业。在同一个作业中有些工作模块的处理顺序是不能变动的。而有些则是可以变化的。工作模块一般由分析功能和执行功能组成。分析功能系指对于地震作业卡进行译码、分析,从而确定本模块的工作方式。在一个作业卡中,不管这个工作模块要做多少处理,而分析功能却只进行一次。执行功能系指工作方式确定之后对地震记录进行处理的功能。有一点应该特别注意,即工作模块执行部分的核心计算往往是用专门的库程序来完成,如科学库、褶积库等。而且其所占的计算时间几乎是工作模块的全部时间。由上面描述,不难看出并行化有下述几种方式:

- (1)对于子操作系统的功能并行化;
- (2)对于各专门的库程序并行化;
- (3)各工作模块的程序并行化;
- (4)全面并行化。

本文将着重讨论(1)~(3),最后顺便指出(4)的优劣。得出的结论是(1)的处理可以收到事半功倍的效果。相对而言,(2)~(4)是不可取的。特别是单独实行(3)是最拙劣的方案。我们受某种思想框框的约束,曾力图把(2)~(3)结合起来做一点并行化试验。受各种因素的影响,这种试验很少真实性。现在看来尽管我们做了一些工作,也还是忍痛割爱了。但在这些工作的启发下,我们进行了本文的讨论。对于本文的观点在机器上实现是一个具有相当规模的软件工程,没有足够的人力、物力、财力的支持是不能实现的。这正是本文的工作到此终止的原因。

1.在子操作系统部分并行化

地震数据处理作业子操作系统为地震记录处理工作模块的运行创造了各种条件,并且进行了全面的管理。

从另一方面来说,工作模块的工作也必须在子操作的控制下才能运行。

并行机进行并行处理是在配置程序的安排下进行的。无论串并行关系、等待关系、信息交换关系,都受配置程序所制约。

近年来一个作业中同一个工作模块进行相同处理的不是几十道记录,而是上千道记录。在同一个作业中,有时几个工作模块的处理顺序是可以交换的。对于同一道记录,有时对于它进行处理并不以另一道记录状态如何有关,即它们并不要求有同步考虑。同一个工作模块分析功能只要在执行功能执行前进行完毕就可以了,中间距离多长时间是无关紧要的。

所有这些,说明了地震数据处理工作具有很强的并行特性。而且这种并行性质与各工作模块的自身特性是无关的。为了均衡并行机各处理器的工作量,只须知道各工作模块对于各道记录的大致处理时间就可以了。

一个较为合适的并行方案是用子系统控制实行同一工作模块对于不同道及不同工作模板在各个处理器上的并行。

例如一个有 m 个处理器的并行机处理 n 道记录。我们考虑可交换顺序的 r 个工作模块来处理这些记录。假定第 k 个工作模块处理第 j 道记录所用的时间为 a_{kj} 秒,则用单个计算机完成这几个工作模块所用的时间为:

$$\sum_{k=1}^r \sum_{j=1}^n a_{kj}$$

设所考虑的是最简单的情形,即 n 为 m 的整数倍时,如为 P 倍。记

$$a_k = \max\{a_{kj}\}$$

则处理时间最多为:

$$P \sum_{k=1}^r a_k$$

如果 n 不是 m 的整数倍,我们记

$$n = lm + h$$

则我们可以首先完成 $(l-1)m$ 道记录,再对 $m+h$ 道记

录做各种安排。可以让 $(l-1)m$ 道记录先处理完,而再安排其它模块的处理。也可以完成一些工作模块的分析部分。总之,时间最大为:

$$(P+1) \sum_{k=1}^r a_k$$

这样做子操作系统是比原来要复杂一些,而且对于各工作模块完成各道记录的处理时间要有一个统计值。但所有这些是不难办到的。这样处理有很多优点。它比在其它部分进行并行化所考虑的问题要简单得多。

在子操作系统进行并行化处理,则各工作模块的程序,各库程序中的各程序无须做改变。因为地震勘探数据处理软件系统各程序库及工作模块的工作量是非常大的,相对而言子操作系统的工作量是小的。很多工作模块及库程序很少做大的改动,即使在一个新的机器上实现,也往往没有什么实质性改动。但子系统往往与机器联系比较密切,在新的机型上往往要重新实现。所以这种处理方式工作量是小的。

进行了操作系统的设计及实现的人相对的比较少,让部分人去掌握并行机及并行处理手段,对于我们这个方案就够了。对于其它人来说仍可按以前串行处理的方式进行软件开发。因此可以看出,用这种方式人员培养可以省力,研制周期短。

总结上述,我们认为这是一种可行的方案。

2. 库程序功能并行化

从一些研究报告可以看出,有一些专门从事地震数据处理的单位或个人,把并行化的精力花在这里,他们的观点产生于下列基本事实。库程序是地震数据处理的核心工作,做地震数据处理很多机时都支付给这些程序的运行上了。库程序的各程序一般都比较简单,解剖、分析、移植、改动、维护都容易实现。库程序中的各程序一般都有较强的数学背景,而且很多都是不太复杂的代数计算,这些计算有大量的并行处理结果可以借鉴。

为了说明他们考虑问题的方式,现举一个例子。

假定有一道记录为:

$$\{a_i\}, \quad i=1,2,\dots,n,$$

我们把它们分成浅层:

$$\{a_i\}, \quad i=1,2,\dots,n_1,$$

及深层:

$$\{a_i\}, \quad i=n_1+1,\dots,n,$$

对其实现如下的平滑滤波。

$$a'_i = \frac{1}{T+1} \sum_{r=i-\frac{T}{2}}^{i+\frac{T}{2}} a_r, \quad i=1+\frac{T}{2}, \dots, n_1,$$

$$a'_i = \frac{1}{2T+1} \sum_{r=i-T}^{i+T} a_r, \quad i=N_1+1, \dots, n,$$

其中 $T+1$ 称为时窗长度。

我们记:

$$T_1 = \sum_{i=1}^{T+1} a_i, \quad T_2 = \sum_{i=n_1-T-1}^{n_1} a_i,$$

$$T_3 = \sum_{i=n_1+1}^{n_1+T} a_i, \quad T_4 = \sum_{i=n_1-T-1}^n a_i,$$

$$S_1 = \sum_{i=n-2T-1}^{n_2} a_i, \quad S_2 = \sum_{i=n_2+1}^{n_3} a_i,$$

$$S_3 = \sum_{i=n_3+1}^{n_4} a_i, \quad S_4 = \sum_{i=n_4+1}^n a_i,$$

其中 S_1, S_2, S_3, S_4 求和长度相等,它们的长度之和刚好为 T 。于是各处理器做如下分工。即:

第 1 处理器求 T_1, S_1 , 做 $a'_i (i=1, 2, \dots, i_1)$ $i=1, 2, \dots, T/2$ 为斜坡处理等, i_1 的取法是使第 1 个处理器与第 2 个处理器工作量平衡。

第 2 处理器求 T_2, S_2 , 做 $a'_i (i=i_1+1, \dots, n_{21})$

第 3 处理器求 T_3, T_2+T_3, S_3 , 做 $a'_i (i=n_1+1, \dots, i_2)$, i_2 取法是使第 3 个处理器与第 4 处理器工作量平衡。

第 4 处理器, 求 $T_4, S_4, T_4+S_1+S_2+S_3+S_4$, 做 $a'_i (i=i_2+1, \dots, n-T)$, 做尾部处理。

这时 4 个处理器的工作量大体是平衡的。

但我们发现,这要求所给的参数 T 及 m_1 必须都是较好的值。而且深层时窗刚好是浅层时窗长度的 2 倍。对于一般的情形,安排 4 个处理器工作量均衡也并不算困难。

现在来讨论一下如何处理的优劣。其优点在前面已经叙述过了。下面看一看它的主要缺点及相关问题。

如果在这种考虑下进行并行化,那么各个库程序中的所有程序均要进行并行化。因而工作量很大。而且要求所有搞库程序的人都必须对并行机及并行处理有所了解。因而人员的培训工作量是大的。

这种并行化与参数有一定关系,于是并行程序可能是一成不变的,造成配置程序放在什么地方有一定困难。如果放在子操作系统,则子操作系统必须深入到各模块的内部结构,如果放在各模块,搞工作模块的人又要去了解各程序库的内部结构。特别当一工作模块调用

多个库程序时,程序会显得十分复杂。如果配置程序放在各库中,也是十分杂乱的。

总之,这样干工作量大,问题复杂,效果也不会很好。

3.各工作模块并行化

从形式上看,地震数据处理作业都是通过各工作模块来完成的。从各工作模块的功能介绍尤其给人以这种印象。看来对各工作模块的并行化似乎是提高地震数据处理的关键所在。即使不能这样,也有人认为对地震作业模块进行并行化试验会对地震作业并行化处理的优劣有说明作用。为了回答这些问题,要做一些讨论。

首先介绍在一般情况下各工作模块的工作内容。为了具体,不妨以 RECOV 为例来讨论这一问题。

RECOV 是一个振幅恢复程序,同时对于输入的一道地震记录可进行浮点到定点的转换。振幅恢复功能如下:

(1)在给定的时窗内计算振幅绝对值之和。一般情况下,把一道记录分成两段,浅层区及深层区。各段用不同的时窗长。常常规定深层时窗长度是浅层时间长度的2倍。有了时窗长度后,在时窗内求振幅绝对值之和。依次对整道记录进行滑动做如下处理。如是求出一条曲线,称为振幅曲线。

(2)对振幅曲线进行平滑。为了消除(1)中求出的振幅曲线的高频成分,还要对它进行平滑滤波。但这里并不采用通常的平滑滤波,而是对振幅曲线进行30个点的平滑。我们记平滑前各点的振幅为 a_i ,平滑后各点的振幅为 a'_i ,则有:

$$a'_i = \frac{1}{30} \sum_{r=i-14}^{i+15}$$

经过平滑后的振幅曲线称为权函数曲线。我们将利用这个权函数曲线进行振幅恢复。

(3)对地震道记录进行振幅恢复。对于要处理的记录,其中有些在权函数曲线上找到相同的点,于是两个值相除便得出了恢复后的振幅。

格式转换功能指如果地震记录为浮点数时,而一般库程序只是处理定点数,因而要把浮点数换成定点数。

除以上主要功能外,RECOR 还有下述几项功能。

判别是否为无效道;

道头字处理;

如一道记录中0记录多于1/3个时视为无效道;

水低处理功能;

切除功能;

做斜坡功能。

下面再来看一下程序组织。程序分成两大部分,即分析模块和执行模块。

分析模块的主要功能是对作业卡做译码工作,且对译出的码做参数加工。其次还有申请缓冲区,以及给缓冲区赋初值,调水底库及切除库,并将这些值,通过地震记录处理操作子系统送入约定好的存储片,以便执行模块调用。对于每一个作业卡,分析模块只执行一次,而不管执行模块执行多少次。

RECOV 模块的所有处理功能均由执行模块执行。但,全部振幅恢复功能均是由 SP_1 库中的专门程序完成的。

从前面介绍中可以看出,尽管执行模块中有3个子块,但大部分计算时间都是在 SP_1 库中用掉。3个子块为处理水底切除模块;处理前切除模块;振幅恢复模块。形式上看振幅恢复模块只占1/3,但其工作时间占19/20左右,这个数据是只做一道记录处理时经多次统计得出的。若要处理道数增加,这个比例还要增加。由此看来,改进模块功能,提高计算速度关键在 SP_1 库及科学库。

从上面介绍不难看出,并行工作在模块上进行是工作量最大的,但其效果几乎为零。所以这是极不可取的。

总之,我们的观点是,地震数据处理工作并行化最好是在子操作系统上进行。这样各模块及各专门库几乎不必做什么工作,可以收到事半功倍的效果。在 SP_1 库及科学库等专用库进行,有一定价值,但效果不大。在各模块进行,工作量最大,而效果最小,所以是最不可取的。

如果进行全面并行化,第1,有些工作根本不会发挥什么作用,正如前面所介绍的。第2,要用大量的人力、物力、财力、投入多而产出少,周期长,而收效甚微。第3,要进行复杂的信息传输,组织不好,极易发生混乱。总之,全面并行化也是不可取的。

可以看出,如果在子操作系统部分,做良好的并行处理,并行机中的各处理器效率都会得到发挥。因而提高计算速度是很可观的。当然,要想并行处理做得好,必须付出艰苦努力。

至于有关的其它问题,如适用于地震数据处理的并行机的特征等问题,在此就不做讨论了。