·开发研究与工程应用· 文章编号：1000—3428(2013)04—0314—04 文献标识码：A 中图分类号：TP301.6

一种 H.264/AVC 视频编码并行算法

夏 龄，舒 涛
(四川民族学院网络信息中心，四川 康定 626001)

摘 要：针对 H.264/AVC 视频编码串行算法编解码耗时长的问题，提出一种新的基于动态调度的 H.264/AVC 视频编码并行算法。以画面组为并行编码单元，各个并行计算单元之间采用全局通信模式，即 Master-Worker 模式进行互相通信，可降低并行编码的通信成本。实验结果验证了该算法在 H.264/AVC 视频编码应用中的优越性，能够保证视频的质量，且具有较高的编码加速比。

关键词：动态调度；静态调度；H.264/AVC 编码；并行算法；画面组级别；视频编码

A Parallel Algorithm of H.264/AVC Video Encoding

XIA Ling, SHU Tao
(Network Information Center, Sichuan University for Nationalities, Kangding 626001, China)

【Abstract】Aiming at long encoding time of H.264/AVC video encoding serial algorithm, a new parallel algorithm based on dynamic scheduling strategy is proposed. The algorithm takes Group of Picture(GOP) as parallel encoding units and uses Master-Worker mode for communication with each other, reducing the communication costs of parallel encoding. Experimental results demonstrate its effectiveness and superiority of the algorithm in video encoding application, ensuring the quality of the video, getting high speedup. The algorithm has the ability to adapt to various status of node easily.

【Key words】dynamic scheduling; static scheduling; H.264/AVC encoding; parallel algorithm; Group of Picture(GOP)-level; video encoding
DOI: 10.3969/j.issn.1000-3428.2013.04.072

1 概述

H.264/AVC 视频压缩编码标准是国际电信联盟的视频编码专家组和国际标准化组织的活动图像专家组联合开发的视频编码新标准，又称 MPEG-4 AVC。H.264 标准与 MPEG-4 AVC 经常联合起来使用，称为 H.264/AVC[1]。

与以往标准相比，在相同的图像失真度下，H.264/AVC 相对 MPEG-2 可节省 60%~70%的码率，相对 H.263 及 MPEG-4 也可节省 40%~50%的码率[2]。可见，H.264/AVC 的编码性能是相当高的，但是其获得优越性能的代价是计算复杂度的增加。H.264/AVC 采用相当复杂的编码技术和模式选择，因此，运算复杂度远高于以往的编码标准。根据 JVT 会议文件的评估结果，H.264/AVC 相比 MPEG-4 编码复杂度约为 10 倍以上，解码复杂度则为 3 倍以上[3]。

如此高的复杂度导致 H.264/AVC 的编解码耗时长，难以满足某些实时性要求较高的应用需求。如何在不牺牲 H.264/AVC 编码效率的前提下，降低其运行时间，成为相当重要的研究方向。因此，在不改变编码效率的前提下，采用并行方式编码视频，缩短编码时间成为研究趋势。

关于 H.264/AVC 视频编码并行算法的相关文献较少。文献[1]将 Slice 单元作为并行算法单元，提出了一种并行编码算法，在 4 个节点时，其加速比为 2.7；文献[3]仅在机群上简单实现了视频编码并行化，并没有给出详细的过程；文献[4]提出了一种新的并行编码体系结构；文献[5]在构建工作站机群系统上，给出了 H.264 视频编码的并行实现方法，采用联合调度策略对视频进行并行编码，但是该方法过于复杂，难以实现；文献[6]利用超线程技术对视频编码并行化；文献[7]基于多核技术提出视频编码并行算法。

本文在分析 H.264/AVC 编码原理的基础上，提出一种新的基于机群的 H.264/AVC 视频编码并行算法。选择画面组(Group of Picture, GOP)作为并行编码单元，将编码任务分配到各并行单元上。在各并行单元间采用全局通信模式——Master- Worker 模式进行互相通信。Master 节点采用动态调度策略对编码任务进行分配与回收，完成视频的编码。

基金项目：四川省教育厅科研基金资助项目(10ZC025)
作者简介：夏 龄(1970－)，男，副教授、硕士，主研方向：视频编码，高性能计算；舒 涛，讲师、硕士
收稿日期：2012-07-17 **修回日期：**2012-09-18 **E-mail：**miaomiao066257@126.com

2 H.264/AVC 视频编码并行算法

视频编码并行算法就是用多台处理器联合对视频进行编码，故并行编码要比串行编码所花费的时间少得多。对于 H.264/AVC 视频编码并行化来说，并不是简单地将编码任务分配到若干个处理器上就可以完成编码。需要分析 H.264/AVC 标准，找出在编码过程中可以并行运行的编码单元，将该单元分配到各个处理器上，而且还要考虑各个处理器之间的通信模式、调度方式，以及有效获取处理器的状态等以决策处理单元是否能够分配编码任务。

2.1 并行编码单元选择

如图 1 所示，采用 H.264/AVC 标准并行编码一个视频序列，任务单元可以选择在 GOP、Frame、Slice、Macro Block、Block 以及 Block 以下级别上考虑[8]。

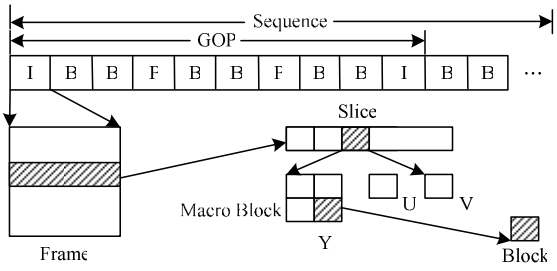


图 1 视频数据结构

GOP-级别：GOP 是一个可以独立编码的帧序列，所以 GOP 是一个很好的并行任务单元。但是 GOP 作为任务单元颗粒较大，而且 GOP 自身的编码速度各不相同，因此，难于平衡节点之间的负载。

Frame-级别：I、P、B 帧之间存在着强烈的依赖关系和顺序问题，节点之间需要大量的信息通信才能维持这种关系，使得运行时间大幅提高。

Slice-级别：Slice 是一个独立编码单元，不需要参考同一帧中其他 Slice 进行解码，也是一个大小适合的任务单元。由于一幅图像只能划分为有限数目的 Slice，因此限制了并行任务的数目。而且随着划分 Slice 数目的增加，会更多地打断帧中宏块的相关性，而宏块不能利用和另一个 slice 中的宏块间的相关性来编码，编码的效率就会降低[3]。

Macro Block-级别、Block-级别以及以下级别：编码每个任务单元都需要大量的参考信息，导致节点间通信量巨大，影响总体运行速度[5]。

考虑以上各个任务单元的优缺点，不牺牲 H.264/AVC 编码效率，并且不影响视频压缩质量的情况下，本文的并行算法选择 GOP 作为并行编码任务单元，并且采用一般串行编码序列 IBBPBBPBBPBBPBBI 作为一个 GOP 单元。

选择 GOP 作为并行任务单元主要有以下优点：(1)GOP 可以被独立并行编码，而且在编码时不需要其他 GOP 信息；(2)节点之间的通信量少；(3)节点数量不受任务的限制，可以任意增加与减少节点；(4)节点数量改变不会影响编码效率以及编码质量；(5)易于平衡各节点之间的负载。

2.2 并行通信模式选择

全局通信模式——Master-Worker 模式如图 2 所示。该模式适用于具有完全并行性(即能分成完全独立的小任务)的算法。原数据集可简单地被划分成互不相交的子集，不同的子任务对不同的数据子集进行运算，子任务间没有约束关系，因此，Master-Worker 模式添加或减少节点都十分方便。

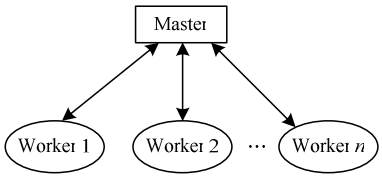


图 2 Master-Worker 模式

GOP 是一个可以独立编码的帧序列，不需要互相参考信息，本文选择 Master-Worker 模式十分适合。对于每一个视频序列(Sequence)来说，只存在一个 Master 节点，负责视频编码初始化、GOP 的分割、分发、与 Worker 节点通信、码流收集、汇总；存在多个 Worker 节点，分别执行不同 GOP 的编码，并将码流返回给 Master 节点，它们由 Master 节点进行控制，只与 Master 节点进行通信，Worker 节点之间没有直接的通信。

2.3 并行调度算法选择

任务调度问题是根据一定的调度规则和调度策略，把组成并行算法的一组任务，按照一定的执行时序分配到并行分布系统的多个计算节点上，以期望取得较好的系统执行性能。按照何时执行任务调度操作来划分，任务调度算法可分作静态调度算法和动态调度算法。

静态调度[9-10]算法是在程序执行前已经得知所有任务的调度情况。基于静态调度的 H.264/AVC 视频编码并行算法，以视频编码过程为运行依据。Master 节点依次发送一组 GOP 的运行参考信息给 Worker 节点，获得任务的 Worker 节点对 GOP 进行编码，将码流返回给 Master 节点，当所有任务节点的 GOP 码流全部返回给 Master 节点后，Master 节点才能进行下一组 GOP 信息分发，如图 3 所示。

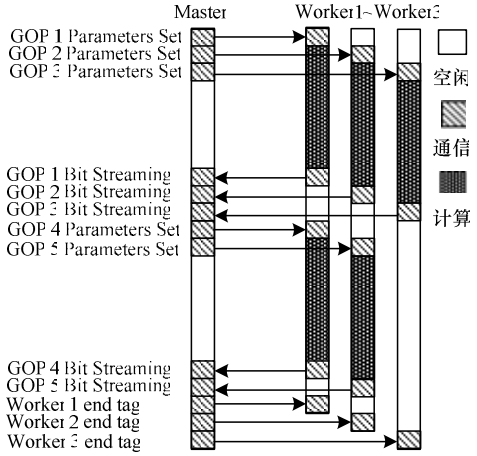


图 3 静态调度的 H.264/AVC 视频编码并行算法

这种静态调度算法维护了整个视频序列编码码流的完整性和正确性，而且步骤清晰，易于掌握和调试，但是只适合于大小相同的任务单元和理想的运行环境，不能适应于不可预测的系统行为。由于并行算法的运行环境选择在 PC 机群上，PC 机自身的配置、运行速度和工作状态各不相同，而且 GOP 自身的编码速度也各不相同，因此 Worker 负载不平衡问题严重，静态调度算法无法适应这种不可预测的系统行为。

动态调度算法在运行过程中进行任务分配，能够灵活地适应系统的变化，因此，采用一种合理高效的动态调度算法，可有效缓解负载不平衡所引起的并行算法运行效率下降问题。具体过程为：首先 Master 节点依次发送一组 GOP 的运行参考信息给 Worker 节点，然后循环对各个 Worker 节点的编码情况进行查询，当查询到某一个 Worker 节点的 GOP 编码结束时就马上接收码流，当查询到某一个 Worker 节点的码流接收完成时就马上发送下一个 GOP 的运行参考信息给该 Worker 节点，这样循环查询直到整个视频编码完毕，如图 4 所示。

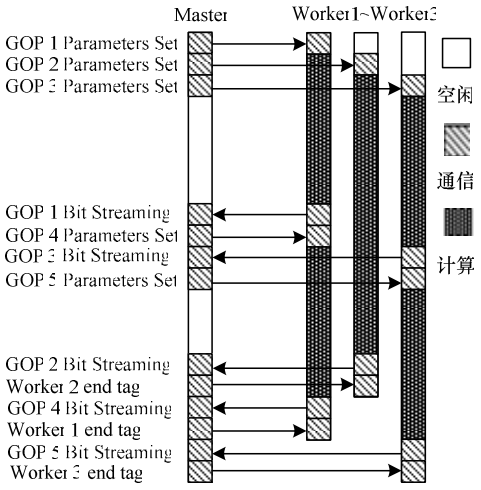


图 4 动态调度的 H.264/AVC 视频编码并行算法

这样，面向节点的动态调度算法在编码过程中动态调整负载平衡，避开了不同 GOP 在不同 PC 机上编码时间不一致的问题，省略了同步等待时间，从而提高了运行效率。动态调度算法虽然保证了视频编码码流的完整性，但是乱序接收问题却导致了码流的不正确性，因此，需添加码流整合功能，设置码流位置记录器，使接收到的 GOP 码流可以插在整个编码码流的正确位置上，保证码流的正确性。

3 门限优化

在用 PC 机群搭建的并行算法运行环境中，PC 机所处的工作状态各不相同，有的可能正被用户使用，处于繁忙状态，有的可能处于空闲状态，也有的可能正被用户使用，但并不是非常繁忙。

当运行本文提出的基于动态调度的 H.264/AVC 视频编码并行算法时，处于繁忙状态的 Worker 节点仍然会被分配 GOP 进行编码，这样会出现该 GOP 编码时间过长的问题，甚至会出现只等待某 Worker 节点结束首次被分配的 GOP 编码，而整个视频中其他 GOP 都已编码结束的特殊情况。虽然这已经是动态调度 GOP 任务单元的最好结果，但是这种特殊情况仍然会降低并行算法的运行效率和合理性，影响码流整合的速度，使算法无法在实际中应用。因此，增加门限设置以及在 Master 节点循环查询各个 Worker 节点的编码情况时增加对 Worker 节点工作状态的查问机制，以避免这种特殊情况的发生。

门限设置将 Worker 节点的工作状态划分为繁忙和非繁忙 2 种，在每次分配 GOP 任务单元之前，Master 节点先查问 Worker 节点的工作状态，以 Worker 节点的 CPU、内存、硬盘的使用情况，判断 Worker 节点所处的工作状态。如果 Worker 节点处于繁忙状态，则不分配任务，如果处于非繁忙状态，则分配任务。这种 Worker 节点工作状态的查问机制是添加在循环查询各个 Worker 节点的编码情况中的，使得 Worker 节点的 GOP 编码这一操作可以随着每次循环随时开始和随时停止。

本文使用 CPU 利用率和内存使用率定义 Worker 节点的工作状态。当 Worker 节点 CPU 利用率大于 75%或内存使用率大于 75%时，定义节点为繁忙状态，否则为非繁忙状态。如果 Worker 节点为繁忙状态，则 Master 节点不对该 Worker 节点分配编码任务，否则分配。

4 实验结果与分析

为了验证本文方法的有效性，对该算法进行了验证。实验的运行环境搭建在本实验室内局域网 PC 机群上，以 MPI 消息传递实现节点间数据的通信。在 VC++6.0 编程环境下采用非阻塞式消息传递方式将 H.264/AVC 的参考模型 JM8.6[2]改写成本文提出的动态调度视频编码并行算法，并以 QCIF(176×144 像素)格式的 Akiyo 视频[11]作为测试序列，如图 5 所示。



图 5 Akiyo 测试视频的第一帧

为了衡量视频编码并行算法性能，本文从编码加速比、编码效率和编码时间等方面对不同算法做了对比。编码加速比[12]和编码效率[13]的定义如下：

$$S_p = \frac{T_s}{T_p} \tag{1}$$

$$E_p = \frac{S_p}{p} \tag{2}$$

在式(1)中，T_S 为最快的串行算法在单台处理器上的运行时间；T_p 为并行算法在 p 台处理器上的运行时间，加速比可以用来衡量算法的并行性对计算时间的减少程度。如果不考虑通信等开销，即在最理想的情况下，若 $T_s = t$，则 $T_s = \frac{t}{p}$，有 $S_p = p$，此时称为线性加速比。由于并行算法总有一些额外的通信开销(如同步等待等)，因此不可能有 $S_p = p$。在式(2)中，$0 < E_p \leqslant 1$，效率 E_p 可以度量并行算法对处理器的利用率，并且可以补偿 S_p 评价标准的不足，S_p、E_p 是一对互补指标。

实验中将本文算法与串行编码算法、静态调度算法做了对比，其中，本文算法和静态调度算法中 Worker 节点的数量分别为 2 个~4 个。表 1 所示为编码前 181 帧得出的实验数据。

表 1 不同算法的实验结果比较

算法	节点数量	比特数	运行时间/s	加速比	效率
串行	1	341 400	551.358	1.000	0.494
静态调度	2	341 400	558.447	0.987	0.598
	3	341 400	307.531	1.792	0.639
	4	341 400	215.859	2.554	0.489
本文算法	2	341 400	563.956	0.977	0.659
	3	341 400	280.572	1.965	0.494
	4	341 400	189.064	2.916	0.729

以上数据为算法运行在不同数目的处理器(各处理器性能和状态并不完全一致)上的结果统计。采用并行算法所得的视频编码码流与采用串行算法所得的视频编码码流完全一致，说明对编码效率和视频质量没有任何影响。

从数据中发现，当采用 2 个节点并行编码时，加速比为 0.977，反而下降了，主要是因为 Master 只负责发送和接收信息，不参与真正的编码操作，与串行编码相比，多了通信时间，所以加速比下降了。但是当 Worker 节点数量增加为 3 个、4 个时，加速比迅速增加，编码时间也远小于串行算法和静态调度算法。当并行编码单元有 4 个节点时，加速比已达到 2.916，非常接近理想线性加速比 3。从表中不难看出，本文算法的加速比和效率总高于静态调度并行算法的加速比和效率，而且编码时间要远少于串行编码算法、静态调度算法。

5 结束语

本文在机群系统的基础上，提出一种新的 H.264/AVC 视频编码并行算法。考虑到编码效率、负载平衡、运行环境、节点状态等多方面因素，选择 GOP 作为并行编码单元，机群中的各个计算单元采用 Master-Worker 模式进行互相通信，而且在编码任务调度中选择动态调度策略，有效地利用了机群中的闲置资源。实验结果表明，该算法是一种高效的自适应 H.264/AVC 视频编码并行算法。

另外本文采用的门限优化方法有些简单，可增加预测机制。对于预测 Worker 节点的工作状态，则可以将多个 GOP 一起分配给某 Worker 编码，进一步减少通信量和通信次数。

参考文献

[1] 宁 华, 梅 铮, 李锦涛. 基于 Slice 的 H.264 并行视频编码算法[J]. 计算机工程, 2005, 31(4): 181-183.

[2] Ralf S, Thomas W, Sehwarz H. The Emerging H.264/AVC Standard[Z]. EBU Technical Review, 2003.

[3] Miguel R, Oliver S. MPEG-4 Natural Video Parallel Implementation on a Cluster[C]//Proc. of the 12th Portuguese Conference on Pattern Recognition. Aveiro, Portugal: [s. n.], 2002.

[4] Peng Chungan, Yu Dunshan, Cao Xixin, et al. A Novel Parallel VLSI Architecture for H.264/AVC Scalar Quantization[J]. Acta Scientiarum Naturalium Universitatis Pekinensis, 2007, 2(4): 1-5.

[5] 邢向磊, 都思丹. 基于工作站机群系统的 H.264 并行视频编码[J]. 东南大学学报: 自然科学版, 2008, 38(6): 949-954.

[6] 王 晗, 林 涛. 超线程技术以及 H.264 编码器中的并行运算分析[J]. 计算机工程, 2006, 32(7): 224-226.

[7] 冯飞龙, 陈耀武. 基于 H.264 实时编码的多核并行算法[J]. 计算机工程, 2010, 36(24): 226-229.

[8] 李 锦, 钟 诚. 异构机群上高效可扩展的 Motif 发现并行算法[J]. 计算机科学, 2012, 39(3): 279-282.

[9] 常 静, 周 宇, 李晓辉, 等. 一种改进的 H.264/AVC 空时编码方案及仿真实现[J]. 系统仿真学报, 2012, 24(4): 907-911.

[10] 罗常英. 基于机群的 H.264/AVC 并行算法研究[D]. 阜新: 辽宁工程技术大学, 2008.

[11] 苏洪磊, 李兵兵, 杨付正. H.264/AVC 网络视频编码失真评估的比特流层模型[J]. 西安交通大学学报, 2012, 46(6): 36-41.

[12] 陈 科. 基于高性能计算机的并行计算研究[D]. 大连: 大连理工大学, 2011.

[13] 吴本涛, 吴敏渊. 一种新的 H. 264/AVC 帧内编码快速算法[J]. 光电子·激光, 2010, 21(12): 1845-1849.

编辑 顾逸斐