

改进的基于 FFT pruning 的窄带高分辨率频谱算法

王 琰, 万 群, 杨万麟

WANG Yan, WAN Qun, YANG Wan-lin

电子科技大学 电子工程学院, 成都 610054

Dept. of Electronic Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China

E-mail: wangyan1509@163.com

WANG Yan, WAN Qun, YANG Wan-lin. Improved high-resolution narrow-band spectra by FFT pruning. Computer Engineering and Applications, 2007, 43(26): 54-55.

Abstract: An improved method of computing high-resolution narrow-band spectra is presented. This method is a generalization of Sreenivas's FFT pruning algorithm and in combination with Nagai's pruning with frequency shift. By dividing the output points into several series, non-power of two output points in arbitrary narrow band can be achieved. This improved algorithm has lower computational cost than Sreenivas's algorithm and has a simpler flow-graph.

Key words: FFT pruning; frequency shift; narrow band

摘 要: 提出一种改进的基于 FFT pruning 的窄带高分辨率频谱计算方法。该方法是对 Sreenivas's FFT pruning 算法和 Nagai 的利用频移变换的 FFT pruning 算法的推广。同时提出输出点分级思想, 可实现任意窄带上非 2 的整数幂次频点输出。该算法比 Sreenivas's FFT pruning 算法具有更小的计算量和更简单的信号流程图。

关键词: FFT pruning; 窄带; 频移

文章编号: 1002-8331(2007)26-0054-02 文献标识码: A 中图分类号: TN91

1 引言

离散傅立叶变换(DFT)是信号处理中的重要变换。要实现点DFT变换需要 N^2 次复数乘法及 $N(N-1)$ 次复数加法。Cooley-Tukey提出的快速傅立叶变换(FFT)使点DFT的乘法量由 N^2 降为 $\frac{N}{2} \log_2 N$ 次^[1]。当非零输入点数远小于FFT阶数或频域上需输出的点数远小于FFT阶数时, Markel^[1]和 Skinner^[2]的FFT pruning算法提高了FFT的运算效率。

当需要获得高分辨率的窄带频谱时, 通常需要在时域数据后填零, 这时FFT的输入端引入大量零点; 而此时只需要某个窄频带上的输出, 输出点数也远小于FFT阶数。Sreenivas, Rao^[3,4]提出的FFT pruning算法将FFT pruning同时作用于输入端与输出端, 提高了运算效率。但这一方法的缺点在于引入求余数运算用于蝶型路径的确定, 增加了算法的复杂度和计算量。

本文结合Nagai提出的“频移”思想^[5], 在对输入/输出采用FFT pruning的同时, 利用频移变换将任意窄带的输出转化为等效的“低频”分量, 避免了求余运算, 使信号流程图得到简化。针对传统的FFT pruning算法均要求输出的点数为2的整数幂次这一缺点, 本文提出对输出点数分级的思想, 可完成任意窄频带, 非2整数幂次点数的输出。

2 算法原理

若输入数据序列长度 $N=2^M$, 其中前 2^L 个数据为非零点。输出只需要频域上序号为 k_1 到 k_2 的点。假设取 $N=16, M=4, L=3, k_1=3, k_2=5$, 采用去除无关路径的方法对DIT-FFT结构图进行化简, 可以得到如图1所示的信号流程图, 其中虚线表示FFT pruning部分。

由于输入点数为2的整数幂次, 故前 $M-L$ 级可采用Skinner's pruning算法^[2], 即将每个非零点复制到 2^{M-L} 个相邻位置。注意到此时输出的点数不再是2的整数幂次, 且不位于低频部分, Markel's pruning算法^[1]对输出端不再适用, 需对其进行推广。

将需输出的频域点进行分组, 使得分组后的点数都为2的整数幂次。对于图1中输出为3点的情况, 可分为两组: $k=3, 4$ 和 $k=5$ 。此时, 两个子输出序列的点数都是2的整数幂次, 分别为 $2^{LF_1}, 2^{LF_2}$ ($LF_1=1, LF_2=1$)。

分组后的子输出序列计算可采用文献[4]中的方法实现。但该方法额外引入了求余数运算, 增加了算法的运算量和复杂度。考虑采用频移方法进行处理。

DFT变换的表达式如下:

$$F_k = \sum_{n=0}^{N-1} f_n W_N^{kn} \quad (k=0, 1, \dots, N-1)$$

基金项目: 国家自然科学基金 the National Natural Science Foundation of China under Grant No.60372022; 新世纪优秀人才支持计划 (No. NCET-05-0806)。

作者简介: 王琰 (1983-), 女, 在读硕士研究生, 从事参数估计、定位等信息获取与探测领域的研究; 万群 (1971-), 男, 教授, 博士生导师, 从事阵列信号处理与移动台定位技术研究。

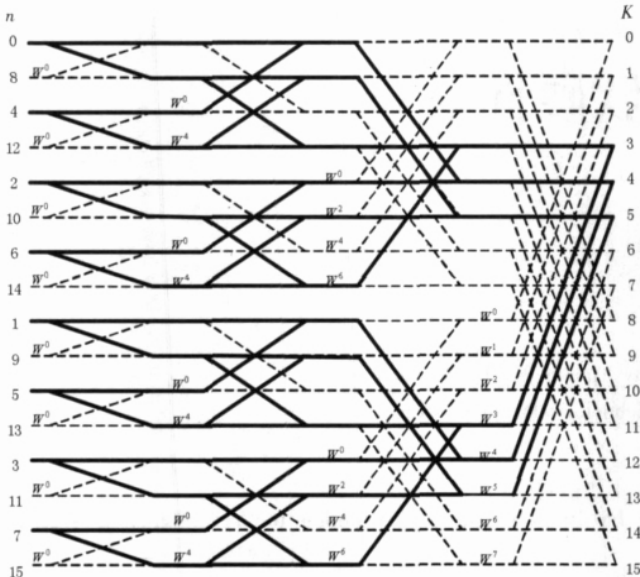


图 1 输出点数为非 2 整数幂次的 DIT-FFT pruning 信号流图

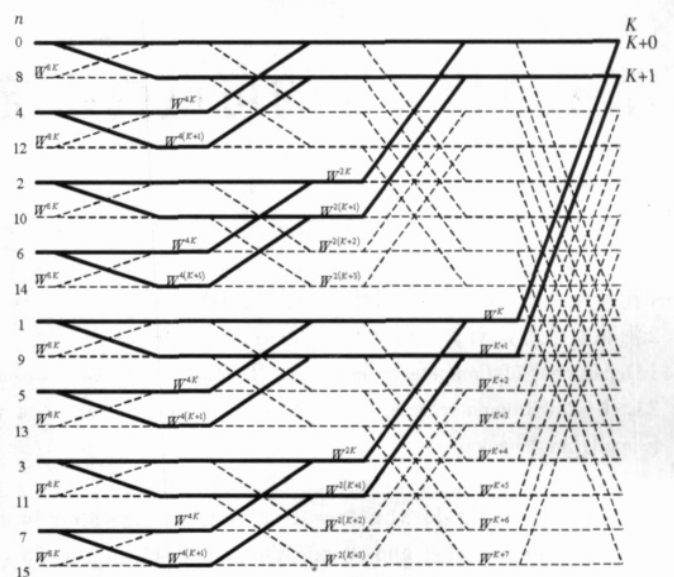


图 2 采用频移变换后第一组子输出序列的 DIT-FFT pruning 信号流图

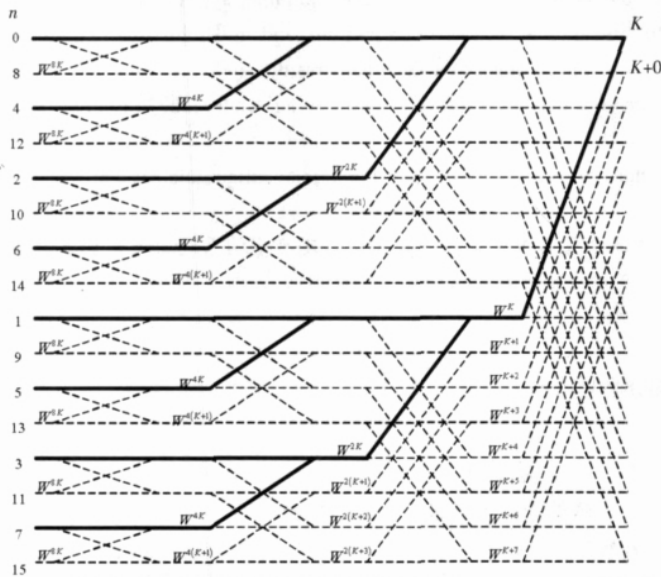


图 3 采用频移变换后第二组子输出序列的 DIT-FFT pruning 信号流图

其中: $W = \exp(j2\pi/N)$, N 为输入时间序列长度。

引入频移变换, 即将上式中的 k 用 $k+K$ 代替:

$$F_{k+K} = \sum_{n=0}^{N-1} f_n W^{(k+K)n} \quad (k=0, 1, \dots, N-1)$$

文献[5]分析可得, 与未作频移时相比, 频移后只是作了旋转因子的改变, 旋转因子由 $W^{2^{i-1}k}$ 变为 $W^{2^{i-1}(k+K)}$, 其中 i 表示级数, $N=2^M$ 。

分别对两个子输出序列作频移变换, 可将任意频段上的输出转化到“低频”部分, 信号流图如图 2、图 3 所示。图 2 中 $K=3$, 图 3 中 $K=5$ 。

由分析可得, 频移并不改变实数乘法次数, 但由于可将任意频段上的点“搬移”到基带部分, 从而简化了流图结构。

3 算法运算量及复杂度分析

由于乘法运算的时间远大于加法运算时间, 故加法运算时间相对于乘法运算时间可忽略。下面对裁减前后的实数乘法次数进行分析。

N 点的 DFT 变换需要 N^2 次复数乘法, N 点 FFT 需要 $0.5N \log_2 N$ 次复数乘法, 每次复数乘法又包含 4 次实数乘法, 故 N 点 FFT 需要 $2N \log_2 N$ 次实数乘法^[6]。

本文涉及对输入、输出端同时采用 FFT pruning 算法, 下面分两种情况进行讨论:

(1) $L+LF \geq M$ 时:

$N=2^M$ 个输入点中仅有 2^L 个非零点, 前 $M-L$ 级可采用 DIT-FFT pruning; 仅有 2^{LF} 个输出点, 后 $M-LF$ 级可应用 DIT-FFT pruning。图 4 给出 $M=4, L=3, LF=2$ 时的信号流图。

此时, 蝶型图的前 $M-L$ 级 FFT 运算可简化为复制的步骤, 前 $M-L$ 级不再需要乘法运算。

蝶型图的后 $M-LF$ 级变为半蝶型单元:

$$\begin{matrix} (a_3, b_3) & \text{---} & [c^* a_{11} + s^* b_{11} + a_2] & (c^* b_{11} - s^* a_{11} + b_2) \\ (a_{11}, b_{11}) & \text{---} & (c, -s) \end{matrix}$$

注: 各数用实部和虚部分别表示。

(下转 141 页)

入了局部搜索算子, 针对已调度的任务对位置作出细微的调整, 局部的调整使它的求精能力也得到了很大地提高, 同时, 整个遗传算法结构也进行了调整, 而且采用了间接编码方式, 不会使不同的染色体对应到相同的调度方案, 从而染色体的多样性增加能直接使调度方案多样性增加, 使算法的搜索空间加大, 因此得到了更好的调度效果, 而且收敛性能也很好。

5 结论

针对网格环境下独立任务的调度问题, 人们提出了很多改进的算法。本文提出了一混合遗传算法, 重新组织了算法结构, 在交叉操作时, 将种群数目加倍, 增加了种群中染色体的多样性, 并在变异操作之后, 加入了针对特定问题的调整操作, 使算法具有了很好的局部求精能力。因此, 该算法很好地克服了早熟收敛的问题。仿真实验表明, 无论在何种异构情况下, 本文的算法的调度性能都比其他算法优异, 而且改进的程度很大, 有很好的收敛性能。(收稿日期: 2007 年 1 月)

参考文献:

[1] Ibarra O, Kim C. Heuristic algorithms for scheduling independent tasks on non-identical processors[J]. Journal of the ACM, 1977, 77 (2): 280- 289.
 [2] Braun T D, Siegel H J. A comparison study of static mapping heuristics for a class of meta-tasks on heterogeneous computing

systems[C]//8th IEEE Heterogeneous Computing Workshop, 1999: 15- 19.
 [3] Maheswaran M, Ali S, Siegel H J, et al. Dynamic mapping of a class of independent task onto heterogeneous computing systems[J]. Journal of Parallel and Distributed Computing, 1999, 50(2): 107- 131.
 [4] Atakan D, Fusun O. Genetic algorithm based scheduling of meta-Tasks with stochastic execution times in heterogeneous computing systems[J]. Cluster Computing, 2003, 7(2): 177- 190.
 [5] Lee Wang, Howard Jay Siegel. Task matching and scheduling in heterogeneous computing environments using a genetic-algorithm-based approach[J]. Journal of Parallel and Distributed Computing, 1997, 47: 8- 22.
 [6] 钟一文, 杨建刚. 异构计算系统中独立任务调度的混合遗传算法[J]. 北京航空航天大学学报, 2004, 30(11): 1080- 1083.
 [7] Wu M Y, Shu W, Zhang H. Segmented min-min: a static mapping algorithm for meta-tasks on heterogeneous computing systems[C]// IPDPS Workshop on Heterogeneous Computing, Canc ún, Mexico, May 2000: 375- 385.
 [8] Wu Min-You, Shu Wei. A high-performance mapping algorithm for heterogeneous computing systems[C]//International Parallel and Distributed Processing Symposium, San Francisco, CA, April 2001.
 [9] Chen G L, Wang X F, Zhuang Z Q, et al. Genetic algorithms and its applications[M]. Beijing: People's Post and Telecommunications Publishing House, 1996.

(上接55 页)

每个半蝶型单元包含 4 次实数乘法、4 次实数加法。用 i 表示当前位于的级数, 当 $LF+1 \leq i \leq M$ 时, 第 i 级包含的半蝶型单元个数为 2^{L+M-i} 。所有半蝶型单元的实数乘法次数为:

$$4 \sum_{i=LF+1}^M 2^{L+M-i} = 2^{L+M} (2^{M-LF} - 1) = 2^{M+2} \cdot 2^{L+2}$$

另外还有 $L+LF-M$ 级为全蝶型单元, 每级包含 2^{M-1} 个蝶型单元, 每个全蝶型单元需 6 次实数加法, 4 次实数乘法。因此全蝶型单元的实数乘法次数为:

$$(L+LF-M) \cdot 2^{M-1}$$

总实数乘法次数为:

$$(L+LF-M) \cdot 2^{M-1} + 2^{M+2} \cdot 2^{L+2}$$

(2) $L+LF < M$ 时:

前 $M-L$ 级可应用 DIT-FFT pruning; 后 L 级可应用 DIT-FFT pruning, 图 5 给出 $M=4, L=2, LF=1$ 时的信号流图。

与前一种情况类似, 前 $M-L$ 级不需要乘法运算。后 L 级为半蝶型单元, 当 $LF+1 \leq i \leq M$ 时, 第 i 级包含的半蝶型单元个数为 2^{L+M-i} 。则所有半蝶型单元的实数乘法次数为:

$$4 \sum_{i=L+1}^M 2^{L+M-i} = 2^{L+M} (2^{M-L} - 1)$$

此时不再存在全蝶型单元, 故上式为总的实数乘法次数。

4 结论

本文提出的方法是对传统 FFT pruning 方法的改进。为达到窄带高分辨率的要求, 对输入/输出端同时采用 DIT-FFT pruning, 引入对输出点分级方法, 结合 FFT pruning 与频移变换, 简化了信号流图, 降低了算法的复杂度, 实现了对任意窄频带上非 2 的整数幂次点数的输出。(收稿日期: 2007 年 1 月)

参考文献:

[1] Markel J D. FFT Pruning[J]. IEEE Transactions on Audio and Electroacoustics, 1971, AU- 19(4): 305- 311.

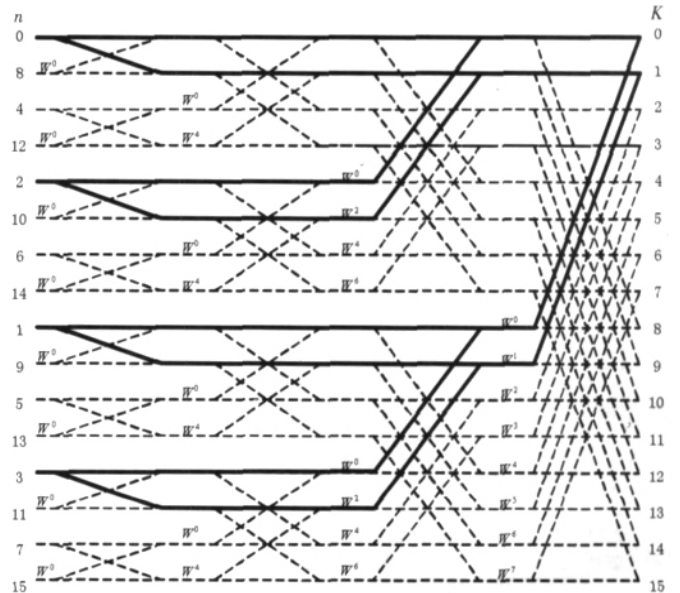


图 5 $L+LF < M$ 时的 DIT-FFT pruning 信号流图

[2] Skinner D P, Pruning the decimation in-time FFT algorithm[J]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1976(4): 193- 194.
 [3] Sreenivas T V, Rao P V. SFFT algorithm for both input and output pruning [J]. IEEE Trans Acoust, Speech, and Signal Processing, 1979, ASSP- 27(3): 291- 292.
 [4] Sreenivas T V, Rao P V. S High-resolution narrow-band spectra by FFT Pruning[J]. IEEE Trans Acoust, Speech, and Signal Processing, 1980, ASSP- 28(2): 254- 257.
 [5] Nagai K, Pruning the decimation in-time FFT algorithm with frequency shift[J]. IEEE Trans Acoust, Speech, and Signal Processing, 1986, ASSP- 34(4): 1008- 1010.
 [6] 胡广书. 数字信号处理——理论、算法与实现[M]. 北京: 清华大学出版社, 2004.