

一种优化的最大逼近角的 CORDIC 算法改进

任全会¹, 吕春峰²

(1 郑州铁路职业技术学院 电子工程系, 河南 郑州 450052;

2 郑州铁路职业技术学院 软件学院, 河南 郑州 450052)

摘 要: 本文提出了一种优化的最大逼近角的 CORDIC 算法改进, 低位通过最高非零位最大逼近角的计算方法实现, 高位通过高速进位加法器的结构实现. 采用基本角度线性编码方式, 不但使流水线方式容易实现, 而且旋转方向也容易确定, 同时减少了迭代次数和增加迭代速度, 精度越高优势越明显. 算法实现不需要 ROM, 能大大节省硬件资源, 可以应用在高精度高速度的运算领域.

关键词: 最大逼近角; CORDIC 算法; 基本角度线性编码; 迭代次数

中图分类号: TN47

文献标识码: A

文章编号: 1000-7180(2013)09-0050-03

An Optimized Maximum Approximation Angle CORDIC Algorithm Improvement

REN Quan-hui¹, LV Chun-feng²

(1 Department of Electronic Engineering, Zhengzhou Railway Vocational and Technical College, Zhengzhou 450052, China;

2 College of Software, Zhengzhou Railway Vocational and Technical College, Zhengzhou 450052, China)

Abstract: The paper proposes an optimized maximum approximation angle CORDIC algorithm improvement, low through maximum approximation angle of highest non zero binary calculation method, structure to achieve high though high-speed binary adder. Using micro-rotation angle recoding for low binary of the angle, pipelined manner not only makes easy to implement, but also to determine the direction of rotation, while reducing the number of iterations and increase the iteration speed, the higher the accuracy the more obvious advantages. The algorithm does not require ROM, significant saving in hardware resources, can be applied in the field of precision high-speed arithmetic.

Key words: maximum approximation angle; CORDIC algorithm; micro-rotation angle recoding; number of iterations

1 引言

CORDIC 算法是一种循环的迭代算法, 广泛的应用在图像处理领域. 传统的 CORDIC 算法主要有高基 CORDIC 算法和角度重编码 CORDIC 算法等, 这些算法在计算过程中存在迭代次数多、运算速度慢、占用硬件资源大的缺点^[1]. 本文提出了一种优化的最大逼近角的 CORDIC 算法, 优化了角度重编码的旋转角度的序列, 这样旋转方向就很容易确定, 使

迭代次数减少, 迭代速度加快. 算法中不需要硬件乘法器, 采用循环迭代节省了硬件资源.

2 CORDIC 的基本原理

CORDIC 算法的基本思想是从起点向量 (x_0, y_0) 进行矢量旋转得到向量 (x, y) , 角度旋转 θ 后得到的向量为^[2]:

$$\begin{cases} x = \cos \theta (x_0 - y_0 \tan \theta) \\ y = \cos \theta (y_0 + x_0 \tan \theta) \end{cases} \quad (1)$$

收稿日期: 2012-11-20; **修回日期:** 2012-12-22

基金项目: 河南省青年骨干教师资助项目(2012GGJS-251)

在硬件实现上令 $\tan\theta = 2^{-i}$, $k = \cos\theta = \frac{1}{\sqrt{1+2^{-2i}}}$, 式(1)变为:

$$\begin{cases} x = k(x_0 - y_0 2^{-i}) \\ y = k(y_0 - x_0 2^{-i}) \end{cases} \quad (2)$$

如果旋转任意角度为 $\theta_i = \arctan 2^{-i}$, 则总的旋转角度为:

$$\theta = \sum_{i=0}^n d_i \theta_i \quad (3)$$

式中, $d_i = \pm 1$, 取决于正时针旋转还是逆时针旋转^[3]. 最终的迭代方程为:

$$\begin{cases} x_{i+1} = k_i(x_i - y_i d_i 2^{-i}) \\ y_{i+1} = k_i(y_i - x_i d_i 2^{-i}) \end{cases} \quad (4)$$

式中, $k_i = \cos(\arctan(2^{-i}))$. k_i 的乘积为比例因子 k 可表示为:

$$k = \prod_{i=0}^{n-1} k_i = \prod_{i=0}^{n-1} \frac{1}{\sqrt{1+2^{-2i}}} \quad (5)$$

式中, 值由迭代次数 n 决定, 如果 $n = \infty$, 则 $k = 0.60725$.

3 优化的最大逼近角的CORDIC算法

从CORDIC算法原理可以看出旋转角度的序列是定值, 在每次的迭代计算得出结果后才能确定旋转方向. 使用BBR(Binary to Bipolar Recoding)和MAR(Micro-rotation Angle Recoding)的方法不但能很容易的计算出旋转方向, 而且能加快迭代次数^[4].

设旋转角度 θ 的区间为 $-\frac{\pi}{4} \leq \theta \leq \frac{\pi}{4}$, 用二进制表示 $\theta = \sum_{i=1}^M B_i 2^{-i}$, 其中 M 是位数, 补码可表示为:

$\theta = -B_0 + \sum_{i=1}^M B_i 2^{-i}$, 其中 M 是位数, 补码可表示为:

$\theta = -B_0 + \sum_{i=1}^M B_i 2^{-i}$, θ 由两部分构成:

$$\begin{aligned} \theta &= (-B_0 + \sum_{i=1}^{N-1} B_i 2^{-i}) + \sum_{i=N}^M B_i 2^{-i} \\ &= \theta_h + \theta_l \end{aligned} \quad (6)$$

其中 $N = \frac{M-1}{3}$, 可以通过 $2^{-N} - \arctan 2^{-N}$

$< 2^{-M}$ 得到. 所以 θ_h 可表示为:

$$\theta_h = -B_0 + 2^{-1} + \sum_{i=2}^N (2B_i - 1) 2^{-i} - 2^{-N} \quad (7)$$

用 $\arctan 2^{-i}$ 的形式表示 θ_h , 因为 $\arctan 2^{-i} \neq 2^{-i} (i \leq N-1)$, 所以两者的关系可以表示为:

$$2^{-i} \arctan 2^{-i} + \sum_{j=2}^{n(i)} \arctan 2^{-Q_j^i} \quad (8)$$

$$= \sum_{j=1}^{n(i)} \arctan 2^{-Q_j^i} + w_i$$

式中, $Q_j^i = i, j = 1, 2, \dots, N-1$; w_i 是 θ_h 每一位的误差; $n(i)$ 是迭代次数. 通过式(8) 可以把 θ_h 表示为:

$$\theta_h = \sum_{i=1}^N \delta_i \left[\sum_{j=1}^{n(i)} \arctan 2^{-Q_j^i} + w_i \right] - 2^{-N} \quad (9)$$

式中, $\delta_i = 2B_i - 1, i = 2, 3, \dots, N$. θ_h 的逼近误差 $\sum_{i=1}^{N-1} w_i < 2^{-N}$, θ 补码的值决定旋转方向. 精度固定和迭代次数固定便于流水线的实现. 在精度等级比较高的时候, 迭代次数的优势会因为误差约束条件的影响而变小^[5]. 通过表1给出 $M = 24$ 使用MAR方法得到的 θ_h 结果可证明这个结论.

表1 $M=24$ 时的 θ_h 结果

2^i	$n(i)$	Q_j^i	$ w_i $
2^{-1}	3	1, 5, 8	$< 2^{-9}$
2^{-2}	2	2, 8	$< 2^{-9}$
$2^{-i}, i=3, \dots, 7$	1	i	$< 2^{-10}$
$\sum n(i) = 10$			$\sum_{i=1}^7 w_i < 2^{-8}$

通过表1可以看出, 当 i 的值在 $[-M, -1]$ 的区间内, θ 的二进制的每个非零位满足 $\arctan 2^i < 2^i < \arctan 2^{i+1}$, 所以 θ 的二进制的每个非零位最大逼近值为 $\arctan 2^{-i}$. 设 θ_i 是经过 i 次旋转后的剩余旋转角度, k 是最高非零位, $\arctan 2^{-k}$ 是经过 $i+1$ 次旋转后的角度, 则 $|\theta| < 2^{-k} + 2^{-k}$, 能够实现对最高非零位进行连续两次逼近. 原来的角度值的最高非零位会因为得到的最大逼近角为零.

根据式(8)中 $n(i)$ 的取值不同可以把 θ_h 分为两部分: $\begin{cases} \theta_{hh}: n(i)=1, i \geq t, t \text{ 为常数} \\ \theta_{hl}: n(i) > 1, i < t, t \text{ 为常数} \end{cases}$, $n(i)$ 的值随着精度的提高而增大, 这样就能减少迭代次数. 对 θ_{hl} 部分进行连续逼近可以加快迭代速度. θ_{hh} 部分采用MAR进行转换:

$$\theta_{hh} = \sum_{i=1}^N (1 - 2B_i) \arctan 2^{-i} + \sum_{i=0}^{t-1} w_i - 2^{-N} \quad (10)$$

把逼近误差和 -2^{-N} 加到 θ_l 上能得到:

$$\hat{\theta}_l = \theta_l + \sum_{i=1}^{N-1} w_i - 2^{-N}$$

$$\begin{aligned}
 &= -\hat{B}_N 2^{-N+1} + \sum_{i=N}^M \hat{B}_i 2^{-i} \\
 &= (1 - 2\hat{B}_N) 2^{-N+1} + \sum_{i=N+1}^{M+1} (2\hat{B}_i - 1) 2^{-i} - 2^{1(M+1)}
 \end{aligned}$$

$$\hat{B}_i \in \{0, 1\} \quad (11)$$

因为 $|\hat{\theta}_i| < 2^{-N+1}$, 精度固定时 $\arctan 2^{-i} = 2^{-i}$, $i \geq N$, 把 θ_h 迭代后算法收敛, 由式(11)可以确定 $\hat{\theta}_i$ 部分的迭代方向:

$$\begin{cases} \hat{\delta}_N = (1 - 2\hat{B}_{N-1}) \\ \hat{\delta}_i = 2(2\hat{B}_i - 1) \end{cases} \quad (12)$$

θ_h 经过优化后的迭代次数为 $2(t-1) + (N-t+1) = t + N - 1$, 同时 θ_i 的迭代次数不变. 优化前和优化后的迭代次数的对比如表 2 所示.

表 2 CORDIC 算法优化前后迭代次数对比

精度 M	N	t	$n(i)$		迭代次数	
			优化前	优化后	优化前	优化后
34	12	4	20	15	42	37
48	16	6	35	21	67	53
60	20	8	46	26	89	67
68	24	9	56	29	102	77

通过表 2 可以看出, 经过 CORDIC 算法优化的迭代次数明显减少, 并且精度越高经过 CORDIC 算法优化的迭代次数比优化前减少的越多, 而且优化算法减少了大量的计算, 节省了硬件资源^[6].

4 优化后的 CORDIC 算法结构

以精度为 60 对优化后的 CORDIC 算法结构进行设定. 当 $M=60, N=20, t=8$ 时计算比例系数:

$$\begin{aligned}
 k &= \prod_{i=1}^7 (1 + 2^{-2i})^{\frac{1}{2}} (1 + 2^{-14})^{-1} \times \\
 &\quad \prod_{i=8}^{19} (1 + 2^{-2i})^{-\frac{1}{2}} \times \prod_{i=20}^{60} (1 + 2^{-2i})^{-\frac{1}{2}} \quad (13)
 \end{aligned}$$

算法结构由 θ_{hl} 、 θ_{hh} 和 θ_i 三部分构成. 其中 θ_{hl} 部分进行两次的迭代逼近, 具体算法结构如图 1 所示. θ_{hl} 部分每一位都进行两次逼近, 坐标计算是 Z 模块, 角度计算是 J 模块. Z 模块的输入是 x_0 和 y_0 , 输出是 x 和 y . 为了保证旋转角度确定, 输入的角度要在 $[0, \pi/4]$ 范围内. J 模块在计算角度时, 使用原码的形式进行计算, 最高非零位的最大逼近角可以用内部减法器的减数表示出来^[7].

5 算法性能分析

图 1 优化后的 CORDIC 算法结构中的每个 Z 模块都由一级寄存器和三级组合逻辑组成. 每个 Z

模块的延时为 T_z , 所有 Z 模块总的延时为 $14T_z$. 每个全加器的延时为 T_1 , 则每个 Z' 模块的延时为 $2T_1$, 所有 Z' 模块总延时为 $102T_1 + T$ (T 是输出寄存器延时). 用 Verilog HDL 语言在 Altera 公司推出的 Stratix IV 型 FPGA 上实现一个 32 位 CORDIC 优化算法, 仿真结果如图 2 所示. 与传统算法相比速度提高了 32%, 节省硬件资源 20.4%.

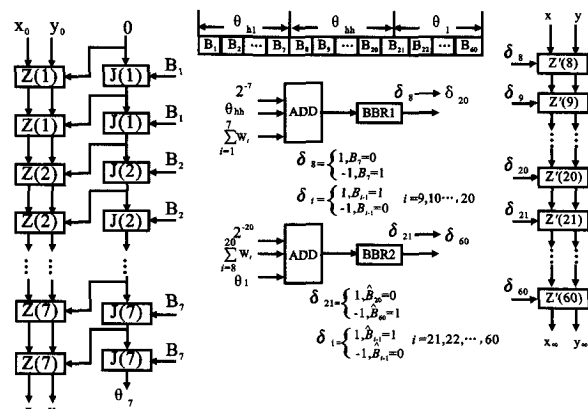


图 1 优化后的 CORDIC 算法结构

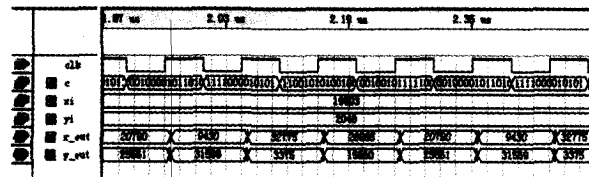


图 2 仿真波形图

6 结束语

本文给出实现 60 位精度的算法结构, 证明了最大逼近角 CORDIC 优化算法的可行性. 该算法不但缩减了繁琐的逼近项的计算, 而且能减少迭代次数, 精度越高, 优化后迭代次数减少越多. 算法还能使流水线更容易实现, 同时不需要 ROM 存储器, 使用 CPLD 或者 FPGA 实现时能减少大量的硬件资源, 使运算速度更快. 优化后的算法可以应用在高速数字图像处理和高速数字通信等领域.

参考文献:

- [1] 张伟, 张安堂, 肖宇. 基于改进的 CORDIC 算法 FFT 的复乘及其 FPGA 实现[J]. 电子技术引用, 2011, 37(4): 51-54.
- [2] 肖振华, 林水生. 基于 CORDIC 算法的 FPGA 实现[J]. 实验科学与技术, 2011, 9(5): 19-22.
- [3] 张建斌, 梁芳, 刘乃安. 一种改进型 CORDIC 算法的 FPGA 实现[J]. 微电子学与计算机, 2010, 27(11): 181-184.

(下转第 57 页)

合影响下寻找出每个节点最优的洪泛范围,进而从两方面最大限度的保证网络的稳定性与传输业务的可靠性.实验证明了该算法能够改善网络的性能.

参考文献:

- [1] Moy J. OSPF version 2[EB/OL]. (2012-10-19). <http://www.ietf.org/rfc/rfc2328.txt>.
- [2] Alaellinoglu C, Jacobson C, Yu H. Towards millisecond IGP convergence[EB/OL]. (2000-11-06). <http://www.nanog.org/meetings/nanog20/abstracts.php?pt=MTA3MiZuYW5vZzlw&nm=nanog20,2000>.
- [3] Xu Mingwei, Yang Yuan, Li Qi. Selecting shorter alternate paths for tunnel-based IP fast reroute[J]. Computer networks, 2012, 56(2): 845-857.
- [4] 武林平, 罗红兵, 艾志玮, 等. 大规模计算系统的主动故障管理方法[J]. 华中科技大学学报, 2010, 38(增刊1): 20-24.
- [5] 张亚普, 孟相如, 张立, 等. 基于 SVM 和模拟逻辑的告警相关性分析[J]. 计算机应用研究, 2011, 28(2): 685-

688.

- [6] 于涛, 陈山枝, 秦臻, 等. 大规模灾害条件下基于连通支配集的重路由方案[J]. 高技术通讯, 2008, 18(1): 11-15.
- [7] 张宇. 人工免疫算法中阴性选择算法的研究[D]. 浙江: 浙江大学硕士学位论文, 2007.
- [8] Betker A, Gerlach C, Hulsermann R, et al. Reference transport network scenarios[R]. MultiTeraNet Project, 2004.

作者简介:

陈铎龙 男, (1988—), 硕士研究生. 研究方向为网络可生存性增强技术.

孟相如 男, (1963—), 教授, 博士生导师. 研究方向为宽带通信网络技术.

徐有 男, (1977—), 硕士. 研究方向为宽带通信网络技术.

袁荣坤 男, (1986—), 硕士. 研究方向为网络可生存性.

(上接第 52 页)

- [4] 毕卓, 戴益君. 全定制 CORDIC 运算器设计[J]. 计算机工程与科学, 2011, 33(10): 64-69.
- [5] 王敏, 薛忠杰. 基于改进 CORDIC 算法的 NCO 设计[J]. 电视技术, 2011, 35(1): 30-33.
- [6] 李杰明, 郑学仁. 基于 CORDIC 算法的数字图像旋转

实现[J]. 电子技术应用, 2009(6): 72-74.

- [7] 段文伟, 于龙洋, 李署坚. 一种改进的 CORDIC 算法及其 FPGA 实现[J]. 微电子学与计算机, 2012, 29(2): 95-98.

作者简介:



任全会 男, (1978—), 硕士, 讲师. 研究方向为 ARM 和 EDA 技术.



吕春峰 男, (1971—), 硕士, 讲师. 研究方向为计算机网络和多媒体.