Journal of Scientific & Industrial Research
Vol. 69, May 2010, pp. 342-349

Feedrate optimization by polynomial interpolation for CNC machines based on a reconfigurable FPGA controller

J J de Santiago-Perez[1], R A Osornio-Rios[1]*, R J Romero-Troncoso[1], E Cabal-Yepez[1] and R G Guevara-Gonzalez[2]

[1]HSPdigital - CA Mecatronica, Facultad de Ingenieria Campus San Juan del Rio, Universidad Autonoma de Queretaro,
Rio Moctezuma 249, San Cayetano, C.P. 76807 San Juan del Rio, Qro., Mexico
[2]Laboratorio de biosistemas, Division de Estudios de Posgrado, Facultad de Ingenieria, Universidad Autonoma de Queretaro,
Cerro de las Campanas s/n, C.P. 76010, Queretaro, Qro., Mexico

Received 30 November 2009; revised 29 January 2010; accepted 01 February 2010

This paper presents an optimization feed rate technique for smooth trajectories generation based on polynomial interpolation. Proposed methodology implements constant or variable feed rate based on a reconfigurable FPGA. Experimental results verified proposed methodology with high efficiency for both approaches.

Keywords: CNC machines, Feedrate optimization, FPGA controller

Introduction

Manufacturing industry has been quickly evolving due to technological advances in high-speed and high-precision machining[1]. Machine-tool automation and implementation of CNC (computerized numerical control) controllers are major thrust for manufacturing companies in order to produce high quality products[2] with reduced production time and increased profits[3]. Feedrate (constant and variable) is most important parameter related to time and quality in machining process. Constant feedrate increases machining speed[4] and quality[5], if toolpath does not present sharp corners or high curvature values. CNC selects constant feedrate for a given operation to produce acceptable performance in operation time and contouring accuracy[6]. On the one hand, some studies[7,8] determined optimal machining parameters [feedrate, cutting speed, and material removal rate (MMR)] in order to reduce cutting forces and vibrations when constant feedrate was used for segmented trajectory without sharp corners or high curvature values. Also, several studies[9-11] analyzed that feedrate optimization is necessary to improve machining efficiency and reduce production time and cost. Methodologies have been proposed to work out both constant feedrate[5,6,12] and variable feedrate[11,13,14-17] approaches. However, implementation of these techniques require evaluation of complex interpolation algorithms, which demand high computational load[18]. Yau & Wang[18] realized a need to develop a real-time interpolation algorithm that can still maintain advantage of smooth interpolation with low computational load; a system capable of implementing both federate approaches in real-time is derisable [19-21].

This paper presents an optimization feedrate technique for smooth trajectories generation through constant and variable feedrate according to toolpath properties. Also, an FPGA (field programmable gate array) implementation was carried out with consequent advantages in real-time application (reconfigurability, low cost and parallel processing).

Experimental Section

Toolpath Generator

For generation of smooth and continuous toolpaths for both constant and variable feedrate approach implementations, methodology considers a set of nodes in XZ plane that determines machining toolpath to determine a sequence of polynomial functions, which describe smooth toolpath by means of spline interpolation. Polynomial functions allow a good interpolation fitting and provide smooth traces.

---

*Author for correspondence
E-mail: raosornio@hspdigital.org

First step is to select a set of nodes $\{u_1=(x_1,z_1),u_2=(x_2,z_2),...,u_N=(x_N,z_N)\}$ from standard CAD/CAM software. Nodes determine initial desired toolpath. In this work, coordinates provided by standard G code were used. Next step aims at performance of toolpath segmentation based on tangential curve behavior. Tangential angles are calculated for each pair of two consecutively nodes u_i, u_{i+1}, for interval $[-\pi,\pi]$, whereas first and fourth quadrants comprehend inclination angles over interval $\left[-\frac{\pi}{2},\frac{\pi}{2}\right]$. However, rotation coordinates were designed in such a way that inclination angles belong to interval $[-0.49\pi, 0.49\pi]$. With this limitation, huge values of inclination angles are avoided and consequently discontinuities in inclination of each segment. Rotation is based on Eq. (1) where α_i, θ_i represent tangential and rotational angle respectively for i-th segment as

$$\theta_i=\begin{cases}\pi & if \quad \alpha_i\in[-\pi,-0.51\pi]\\ -\pi & if \quad \alpha_i\in(-0.51\pi,-0.02\pi]\\ 0 & if \quad \alpha_i\in(-0.02\pi,0.47\pi]\\ \pi/2 & if \quad \alpha_i\in(0.47\pi,0.96\pi]\\ \pi & if \quad \alpha_i\in(0.96\pi,\pi]\end{cases} \qquad \ldots(1)$$

Methodology consists on scanning behavior of rotational angles θ_i. Then, segmentation stage determines k groups of r_k coordinate points $u_{k,1}=(x_{k,1},z_{k,1}), u_{k,2}=(x_{k,2},z_{k,2}),\ldots,u_{k,r_k}=(x_{k,r_k},z_{k,r_k})$ that keep same rotational angle $\theta_{k,1}=\theta_{k,2}=\ldots=\theta_{k,r_k}=\theta$ for k-th segmentation. Next stage of procedure is to calculate rotated and translatedpoints $u^*_{k,1}=(x^*_{k,1},z^*_{k,1}), u^*_{k,2}=(x^*_{k,2},z^*_{k,2}),\ldots,u^*_{k,r_k}=(x^*_{k,r_k},z^*_{k,r_k})$ as

$$\begin{cases}x^*_{k,i}=(x_{k,i}-x_{k,1})\cos\theta+(z_{k,i}-z_{k,1})\sin\theta\\ z^*_{k,i}=-(x_{k,i}-x_{k,1})\sin\theta+(z_{k,i}-z_{k,1})\cos\theta\end{cases} \qquad \ldots(2)$$

Last step is polynomial fitting where polynomial functions were generated for describing smooth toolpath. Generation begins with a group of rotated and translated points $\{u_{k,1},u_{k,2},\ldots,u_{k,r_k}\}$, and then a polynomial function $p_k(x)$ is fitted. Next, interpolation error is measured in order to decide if polynomial has an acceptable fit. Finally, if interpolation error is less than specified maximum permissible error, polynomial is accepted and continues with following segmentation. Otherwise last node is dropped from current segmentation and a new polynomial is generated to interpolate reduced group of nodes. Dropped nodes are located in a new segmentation to generate a new polynomial.

To ensure a C^2 continuous toolpath, a modified least-square method was developed to satisfy $p_k(0)=0$, $p_k(x^*_{k,r})=z^*_{k,r}$, $p'_k(0)=\rho_{k,1}$, $p''_k(0)=\rho_{k,2}$, $\rho_{k,1}=p'_{k-1}(x^*_{k-1,r_{k-1}})$ and $\rho_{k,2}=p''_{k-1}(x^*_{k-1,r_{k-1}})$. A quintic polynomial function $p_k(x)=\sum_{i=0}^{5}a_i x^i$ on range $[0,x^*_{k,r}]$ was proposed for interpolation, since it allows control of first and second derivatives, and tracks segmentation points. Then, toolpath generator provides a sequence of polynomials functions $p_1(x), p_2(x),\ldots,p_k(x)$ with corresponding rotational angles $\theta_1,\theta_2,...,\theta_k$.

Feedrate Scheduling

To schedule feedrate from toolpath characteristics and to generate dynamic profile, optimum feedrate values along entire trajectory that maintains constrained peak accelerations are calculated. Starting with a given trajectory, featuring C^2 continuity, peak dynamics is evaluated along entire trajectory. Peak-acceleration positions R_j occur when Eq. (3) holds.

$$\left[1+p'_k(x)^2\right]p'''_k(x)-3p''_k(x)^2\,p'_k(x)=0 \qquad \ldots(3)$$

From Eq. (9), solutions R_j are numerically calculated, which corresponds to location of peak accelerations all along $p_k(x)$. Then, tool acceleration A_θ is calculated

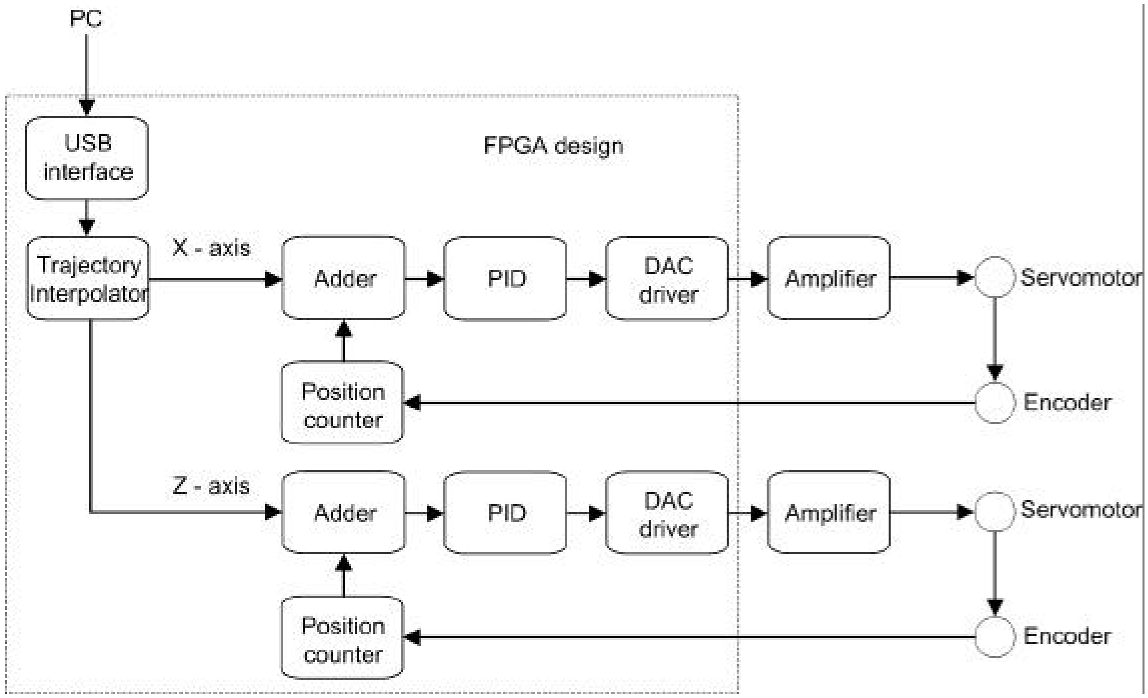

Fig. 1— Block diagram of axes control

for each solution R_j based on Eq. (4) where θ is curve angle determined by Eq. (5) and V_T is desired feedrate.

$$A_\theta = V_T \left| \frac{p_k''(R_j)}{\left[1 + p_k'(R_j)^2\right]^{3/2}} \right| \qquad \ldots(4)$$

$$\theta = \tan^{-1}\left(p_k'(R_j)\right) \qquad \ldots(5)$$

Considering a maximum feedrate V_m, if peak accelerations reach machine acceleration constraint A_m, then feedrate in those nodes V_j can be recalculated through Eq. (6).

$$V_j = \sqrt{\left| \frac{A_m\left[1 + p_k'(R_j)^2\right]^{3/2}}{p_k''(R_j)} \right|} \qquad \ldots(6)$$

In this way, constant feedrate is maintained till peak acceleration is reached, $A_\theta \le V_m$. Otherwise, if curve presents high curvature values then feedrate is varied according to Eq. (7). Then, scheduling feedrate considers both approaches (constant and variable federate) according to curvature behavior with same methodology. Finally, for a given toolpath divided into several polynomials $p_k(x)$, arc length s_k is calculated as

$$s_k = \int_0^{x_{k,r}^*} \sqrt{1 + \left[p_k'(x)\right]^2} \qquad \ldots(7)$$

Once allowable feedrate values along toolpath are established, dynamic profile is generated for each polynomial $p_k(x)$ using reported method[22]. Profile is built considering arc length, starting and ending feedrate, machine constraints, and acceleration and jerk limits. Resulting profile describes tool dynamics along a given toolpath.

Proposed Architecture

Proposed methodology was implemented into an FPGA, where trajectory interpolator, a CNC controller for two axes XZ, and additional structures (interface, adder, PID, DAC driver, and position counter) were embedded (Fig. 1). Additional structures were embedded into same FPGA along with trajectory interpolator showing advantages of FPGA over other implementation

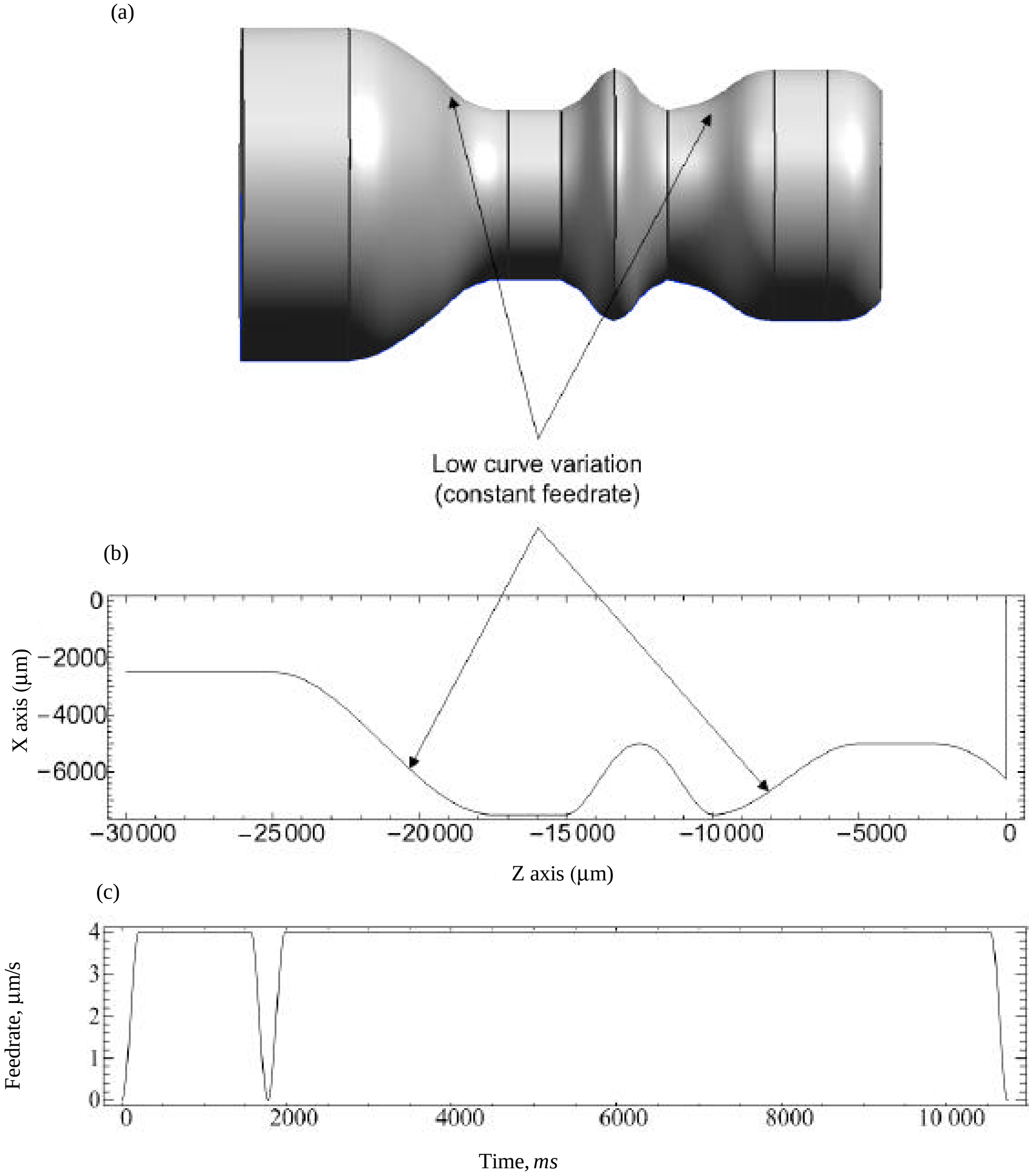

Fig. 2— Constant feedrate: a) CAD model; b) toolpath; and c) feedrate

techniques where several devices (DSP or PC) are needed. This work refers to trajectory interpolator, which is combination of toolpath generator and profile generator. Initially, trajectory interpolator gets information of polynomial functions, rotational angles, sample rate, and generator profile from computer via USB interface. Horner method is implemented into polynomial evaluation block and toolpath is calculated. Rotation module implements Eq. (2) for rotation functions. Profile generator performs parametric technique[22]. Finally, an integrator module is necessary to generate position reference of trajectory for X and Z axes because profile generator and polynomial evaluation work with first derivative of functions.

Simulation

Two case studies were conducted to test versatility and efficiency of feedrate optimization and toolpath generation for constant and variable federate approaches.

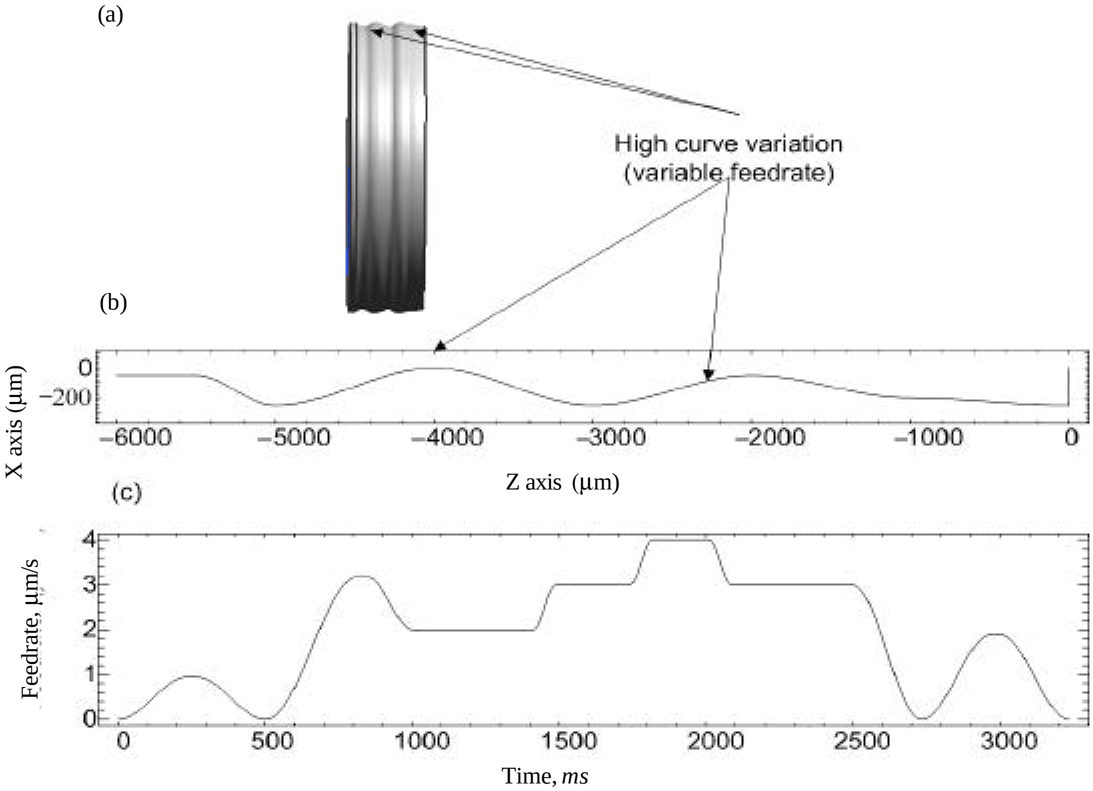

Fig. 3— Variable feedrate: a) CAD model; b) toolpath; and c) feedrate

Case Study 1: Constant Feedrate

A smooth trajectory was generated where constant feedrate approach was optimal due to low variation on curvature [CAD model (Fig. 2a), polynomial toolpath (Fig. 2b)]. For this case, small changes on X axis require larger changes on Z axis. Then, toolpath is optimized with a constant feedrate behavior of tool (Fig. 2c).

Case Study 2: Variable Feedrate

A high-curvature trajectory was generated where approach of variable feedrate was better suited [CAD model (Fig. 3a), polynomial toolpath (Fig. 3b)]. In this case, changes on curve are close, and then it produced high variation on derivative and curvature. Feedrate presents a non-constant behavior due to geometric characteristics of toolpath (Fig. 3c).

Experimentation

Proposed methodology was applied for generation of polynomial interpolation and scheduling feedrate for machining workpiece in a retrofitted to CNC lathe where both case studies were implemented. Machining was performed in a retrofitted to CNC lathe using EZG705-0-101 servomotors, a 6000 *counts/rev* incremental encoder, and Copley-403 servoamplifiers for axes

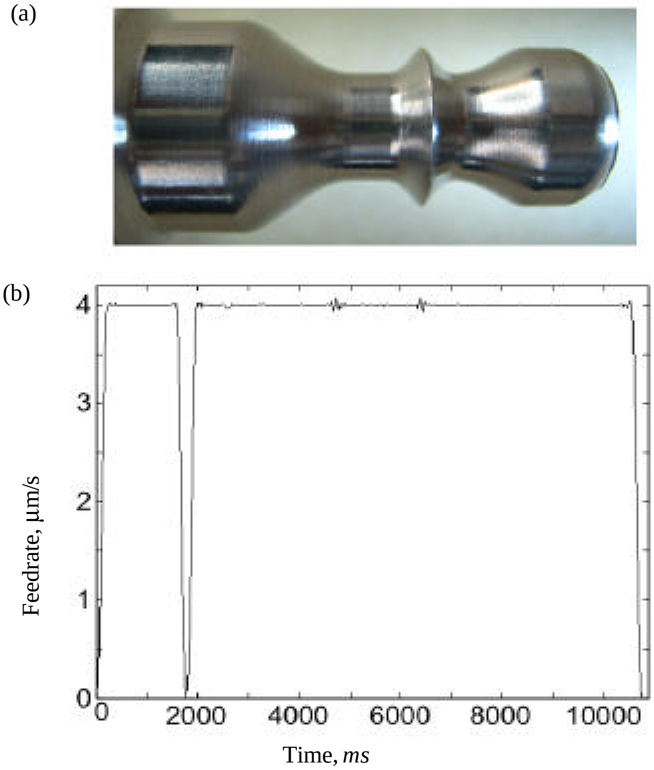

Fig. 4—Experimental results on constant feedrate: a) Machined piece; and b) Measured feedrate

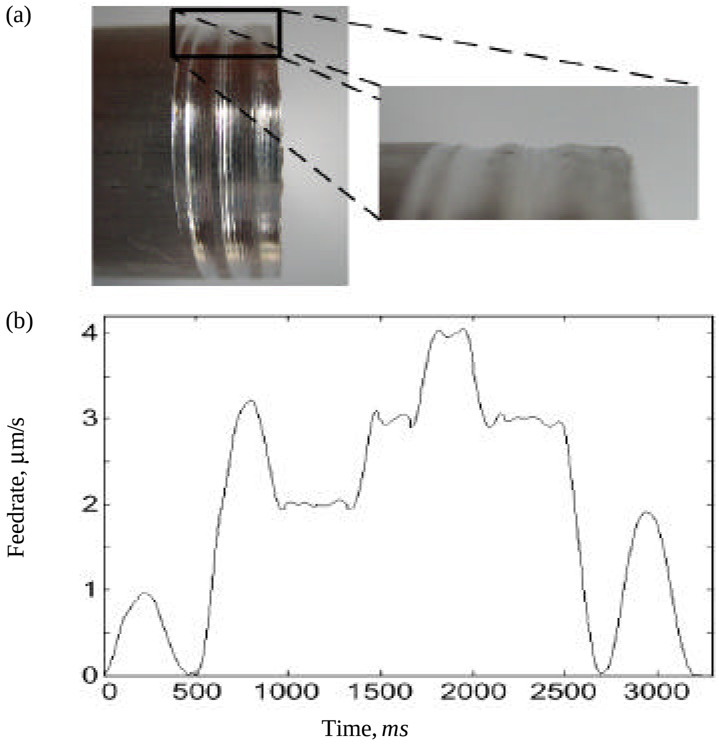

Fig. 5— Experimental results on variable feedrate: a) Machined piece; and b) Measured feedrate

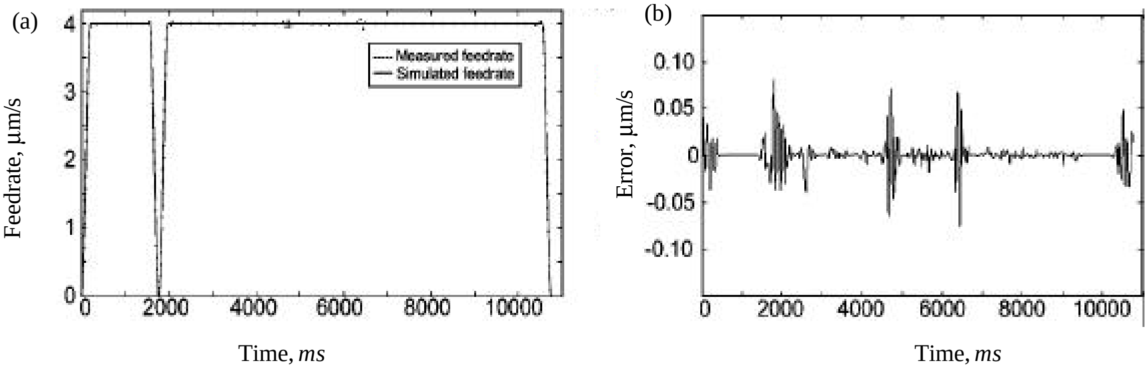

Fig. 6— Analysis of constant federate: a) Comparison between measured and simulated data; and b) Feedrate error

motion. Experimentation with polynomial trajectory is made with a proprietary open-architecture CNC controller[3] into a Spartan-3 FPGA from Xilinx. Experimental setup consists of embedded position controller, DAC driver and USB interface.

Results and Discussion

Experimental result for first case study of constant feedrate presents machined piece of first model (Fig. 4a) and shows feedrate behavior during machining process (Fig. 4b). Feedrate was measured by reported[23] technique starting from encoder information. Experimental result for second case study of constant feedrate presents machined piece of first model (Fig.5a) and shows feedrate behavior during machining process (Fig. 5b).

A comparison between theoretical model and measured data was performed. Analysis of first case study with constant federate shows both measured and simulated data (Fig. 6a) and error between them(Fig. 6b). In this case, mean error is 0.0001 µ*m/s*, with a standard deviation of σ=0.0122 µ*m/s*, and an absolute peak error

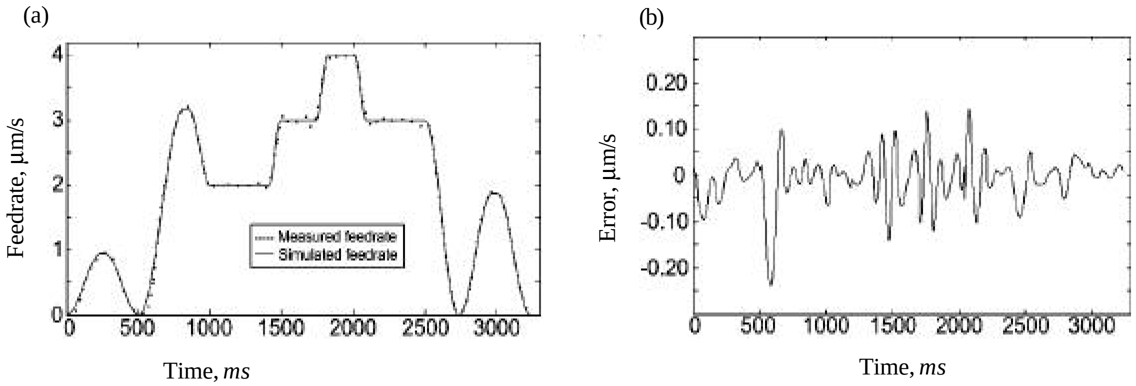

Fig. 7— Analysis of variable federate: a) Comparison between measured and simulated data; and b) Feedrate error

of 0.0811 µ*m/s*, Analysis of second case study with variable feedrate shows both measured and simulated data (Fig. 7a) and error between them (Fig. 7b). For second case study, mean error is -0.0087 µ*m/s*, with a standard deviation of σ=0.0122µ*m/s* and an absolute peak error of 0.2385 µ*m/s*, Also, an analysis by coefficient of determination R-square[24] was performed to measure level prediction between measured and simulated data. Coefficients of determination for first and second case studies were 0.9998 and 0.9990 respectively. Both values were close to one indicating a good level of adjusting.

FPGA advantages (parallel processing, low cost and reconfigurability) allowed implementation of proposed interpolation method along with other complementary modules. This fact provided an advantage of proposed methodology against other available methods[12,13,18] showing excessive computational load of their algorithms and implementation required several devices while in this work all modules were embedded into a single FPGA.

Conclusions

A novel methodology for feedrate optimization based on polynomial interpolation was developed where several parameters (curvature, acceleration and dynamic constrains) were considered for feedrate scheduling. Proposed methodology can generate smooth trajectories, which can carry through both feedrate (constant or variable) according to toolpath properties. Experimental results verified efficiency of proposed methodology for constant and variable feedrates. FPGA advantages (parallel processing, low cost and reconfigurability) allowed implementation of proposed interpolation method along with other complementary modules.

Acknowledgements

This project was partially supported by CONACyT scholarship 199846, PROMEP 103.5/08/5382, FOMIX-QRO-2008-CO2-102123, SEP-CONACyT 84723 and PROINNOVA CONACyT 112051 projects.

References

1 Shin S-J, Suh S-H & Stroud I, Reincarnation of G-code based part programs into STEP-NC for turning applications, *Comput Aided Design*, **39** (2007) 1-16.

2 Correa M, Bielza C & Pamies-Teixeira J, Comparison of Bayesian networks and artificial neural networks for quality detection in a machining process, *Expert Syst Appl*, **36** (2009) 7270-7279.

3 Osornio-Rios R A, Romero-Troncoso R J, Herrera-Ruiz G & Castañeda-Miranda R, FPGA implementation of higher degree polynomial acceleration profiles for peak jerk reduction in servomotors, *Robot Cim-Int Manuf*, **25** (2009) 379-392.

4 Tsai M-S, Nien H-W & Yau H-T, Development of an integrated look-ahead dynamics-based NURBS interpolator for high precision machinery, *Comput Aided Design*, **40** (2008) 554-566.

5 Yeh S-S & Hsu P-L, Adaptive-feedrate interpolation for parametric curves with a confined chord error, *Comput Aided Design*, **34** (2002) 229-237.

6 Dong J, Ferreira P M & Stori J A, Feed-rate optimization with jerk constraints for generating minimum-time trajectories, *Int J Mach Tool Manuf*, **47** (2007) 1941-1955.

7 Lanjewar R W, Saha R, Datta U, Banerjee A J, Jain S & Sen S, Evaluation of machining parameters for turning of AISI 304 austenitic stainless steel on auto sharpening machine, *J Sci Ind Res*, **67** (2008) 282-287.

8 Parameswara C V S & Sarcar M M M, Evaluation of optimal parameters for machining brass with wire cut EDM, *J Sci Ind Res*, **68** (2009) 32-35.

9 Erdim H, Lazoglu I & Ozturk B, Feedrate scheduling strategies for free-form surfaces, *Int J Mach Tool Manuf*, **46** (2006) 747-757.

10 Tae J K, Kim H S & Park S H, Machineability in NURBS interpolator considering constant material removal rate, *Int J Mach Tool Manuf*, **45** (2005) 665-671.

11 Sencer B, Altintas Y & Croft E, Feed optimization for five axis CNC machine tools with drive constraints, *Int J Mach Tool Manuf*, **48** (2008) 733-745.

12 Erkorkmaz K & Altintas Y, High speed CNC system design. Part I: jerk limited trajectory generation and quintic spline interpolation, *Int J Mach Tool Manuf*, **41** (2001) 1323-1345.

13 Erkorkmaz K & Heng M, A heuristic feedrate optimization strategy for NURBS toolpaths, *CIRP Ann-Manuf Tech*, **57** (2008) 407-410.

14 Marchenko Tikhon, Tae Jo Ko, Seung Hyun Lee & Hee Sool Kim, NURBS interpolator for constant material removal rate in open CNC machine tools, *Int J Mach Tool Manuf*, **44** (2004) 237-245.

15 Chen J-S, Huang Y-K & Chen M-S, Feedrate optimization and tool profile modifications for high-efficiency ball-end milling process, *Int J Mach Tool Manuf*, **45** (2005) 1070-1076.

16 Doruk Merdol S & Altintas Y, Virtual cutting and optimization of three-axis milling processes, *Int J Mach Tool Manuf*, **48** (2008) 1063-1071.

17 Altintas Y & Erkorkmaz K, Feedrate optimization for spline interpolation in high speed machine tools, *CIRP Ann-Manuf Tech*, **52** (2003) 297-302.

18 Yau H-T & Wang J-B, Fast Bezier interpolator with real-time look ahead function for high-accuracy machining, *Int J Mach Tool Manuf*, **47** (2007) 1518-1529.

19 Khare K, Singh R P & Khare N, Comparison of pipelined IEEE-754 standard floating point adder with unpipelined adder, *J Sci Ind Res*, **64** (2005) 354-357.

20 Khare K, Singh R P & Khare N, Comparison of pipelined IEEE-754 standard floating point multiplier with unpipelined multiplier, *J Sci Ind Res*, **65** (2006) 900-904.

21 Arun C & Rajamani V, Design and VLSI architecture of non-polynomial based low probability of error (Pb) Viterbi decoder, *J Sci Ind Res*, **68** (2009) 97-106.

22 Osornio-Rios R A, Romero-Troncoso R J, Herrera-Ruiz G & Castañeda-Miranda R, Computationally efficient parametric analysis of discrete-time polynomial based acceleration-deceleration profile generation for industrial robotics and CNC machinery, *Mechatronics*, **17** (2007) 511-523.

23 de Santiago-Perez J J, Osornio-Rios R A, Romero-Troncoso R J, Herrera-Ruiz G & Delgado-Rosas M, DSP algorithm for the extraction of dynamics parameters in CNC machine tool servomechanisms from an optical incremental encoder, *Int J Mach Tool Manuf*, **48** (2008) 1318-1334.

24 Liu Honghu, Zheng Yan & Shen Jie, Goodness-of-fit measures of R^2 for repeated measures mixed effect models, *J Appl Stat*, **35** (2008) 1081-1092.