Design an Implementation of Fuzzy Approximation PI Controller for Automatic Cruise Control System

Pallab Maji, Sarat Kumar Patra, Kamalakanta Mahapatra
National Institute of Technology, Rourkela
Department of Electronics and Communication Engineering
Rourkela, Odisha - 769008
India
pallab.vsp@gmail.com, skpatra@gmail.com, kkm@nitrkl.ac.in

Abstract: Fuzzy logic systems have been widely used for controlling non-linear and complex dynamic systems by programming heuristic knowledge. But these systems are computationally complex and resource intensive. This paper presents a technique of development and porting of a Fuzzy Logic Approximation PID Controller (FLAC) in an automatic cruise control (ACC) system. ACC is a highly nonlinear process and its control is trivial due to large change in parameters. Therefore, a suitable controller based on heuristic knowledge will be easy to develop and provide an effective solution. But major problem with employing fuzzy logic controller (FLC) is its complexity. Moreover, designing of rulebase requires efficient heuristic knowledge about the system which is rarely found. Therefore, in this paper a novel rule extraction process is used to derive a FLAC. This controller is then ported on a C6748 DSP hardware with timing and memory optimization. Later it is seamlessly connected to a network to support remote reconfigurability. A performance analysis is drawn based on processor-in loop test with Simulink model of a cruise control system for vehicle.

Key–Words: Digital Signal Processor, Intelligent System, Genetic Algorithm, Real-time System, Fuzzy System.

Nomenclature

T Engine Torque Constant.

β Torque Co-efficient.

ω Present Torque Rate.

ω_m Peak Torque Rate.

C_r Co-efficient of Rolling Friction.

C_d Drag Co-efficient.

ρ Density of Air.

A Area of the Vehicle.

g Gravitational Constant.

1 Introduction

Fuzzy Logic Controllers (FLC) have gained high appreciation in nonlinear control [1, 2] although Proportional-Integral-Derivative (PID) controllers and Model Predictive Controllers (MPC) are widely used in industrial processes [3, 4, 5]. This is mainly due to its aptitude which can counter nonlinear control problems by programming heuristic knowledge. These heuristics are derived from expert human knowledge and the input and output relationship of the process. However, it is vital that the heuristic knowledge is derived from plant experts. But it has been observed that skilled and expert operator for a process plant is a scarce resource. It is also a proven fact that human knowledge is often prone to error. Therefore it becomes critical that correct system heuristics are used to derive the fuzzy control parameters. Many designs in the literature have used evolutionary techniques for fuzzy parameter (FCP) extraction[6, 7, 8]. However, these methods eliminate the human knowledge from these controllers; which is so imperative. This paper introduces a generic fuzzy logic controller system (G-FLCS) which extracts FCP from an input output relationship of a plant using genetic algorithm and provides a web based user interface (WebUI) for users to tweak these parameters for fine tunning of the fuzzy control system. This system provides a remote accessibility and control over systems. Internet-of-things (IOT) have already been a revolution in the automation industry. Fuzzy logic has been extensively used in ACC and collision avoidance system over the years[9, 10, 11]. However, this paper presents a system that controls an automatic cruise control (ACC) unit of a vehicle using fuzzy logic and connects it seamlessly to the Internet which will allow technical support engineers to remotely configure if it malfunctions. ACC has been used in this paper as an example subsystem of a vehicle but this feature of remote reconfigurability can be extended to other modules of vehicle automation and advance driver assistance systems (ADAS). Fig. 1 graphically describes the control strategy employed for the ACC.

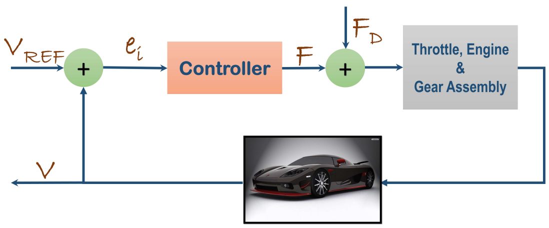


Figure 1: Block diagram of Automatic Cruise Control System

This paper is organized as follows. In section 2 mathematical modeling of ACC is described and is simulated in section 3 for data collection. Fuzzy logic approximation control is explained in section 4 and the proposed hardware G-FLCS is implemented in section 5. Performance analysis of this controller with reference to PID controller and Model Predictive Controller is drawn in section 6. The paper is finally concluded in section 7 and followed by the references used during this research work.

2 Mathematical Modeling Automatic Cruise Control System [12]

Consider a vehicle of mass *m* moving at a velocity *v*. A force *F* is generated from the engine while a disturbance force F_d is resisting motion of the vehicle. Therefore, the equation of motion of the vehicle is given by

$$m\frac{dv}{dt} = F - F_D \tag{1}$$

The vehicle engine generates force *F* which is proportional to the rate of injected fuel in the engine. This phenomenon in turn controls the throttle of the vehicle. Torque produced T at engine speed ω can be mathematically represented as

$$T(\omega) = T_m\left(1 - \beta\left(\frac{\omega}{\omega_m} - 1\right)^2\right) \tag{2}$$

where T_m is maximum torque generated by the engine at full throttle to attain ω_m, the maximum engine speed with torque coefficient β. For a gear ratio *n* and wheel radius *r*, current velocity can be related to the engine speed by (3)

$$\omega = \frac{n}{r}v = \alpha_n v \tag{3}$$

Therefore the driving force can be computed as

$$F = \frac{nu}{r}T(\omega) = \alpha_n u T(\alpha_n v) \tag{4}$$

Basically there are three major disturbance forces working on the vehicle namely, gravitational force (F_G), rolling friction of the road and vehicle tires (F_R), and aerodynamic drag due to the body of the vehicle (F_A).

$$F_D = F_G + F_R + F_A \tag{5}$$

The gravitational force acting on the vehicle can be modeled based on the slope of the roads.

$$F_G = mg\sin\theta \tag{6}$$

$$F_R = mgC_r sgn(v) \tag{7}$$

$$F_A = \frac{1}{2}\rho C_d A v^2 \tag{8}$$

Combining (5) and (6), (1) becomes

$$m\frac{dv}{dt} = \alpha_n u T(\alpha_n v) - mg\sin\theta - \tag{9}$$
$$mgC_r \mathrm{sgn}(v) - \frac{1}{2}\rho C_d A v^2$$

3 Simulation of Automatic Cruise Control System

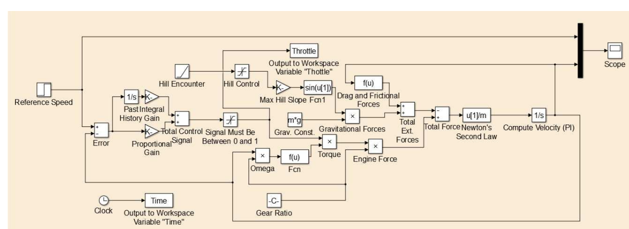


Figure 2: Simulink Model of Automatic Cruise Control

The mathematical model of an ACC presented in section 2 is simulated in Matlab/Simulink with a PID controller. The Simulink model as presented in Fig. 2, is simulated and the system response is observe and recorded. This recorded dataset is further used to extract fuzzy control parameter (FCP) which drives the G-FLCS. This method is carried out using PID control algorithm and therefore, the G-FLCS eventually approximates the system response of the PID controller. However, since the control algorithm uses fuzzy logic in its core, a fast and easy tracking is observed.

3.1 Control Algorithms

There are various control algorithms in literature that have been effective in automatic cruise control systems [9, 8, 13, 14]. However, these control algorithms do not provide a system which can be remotely controlled. The system responses cannot be fine tuned on-field. Moreover, control algorithms like model predictive control (MPC) and PID control are generally sluggish in their approach. In this paper, the system response with proposed G-FLCS control algorithm is compared to system responses with MPC and PID control as presented in Fig. 3.

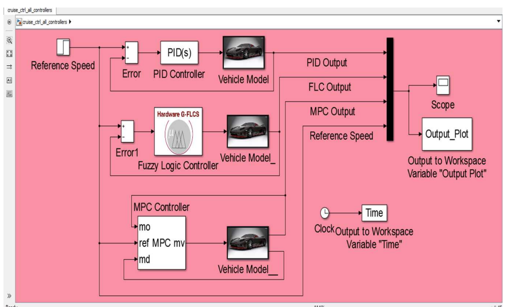


Figure 3: Simulink Model of Automatic Cruise Control System with MPC, PID and proposed GFLCS on C6748 DSP hardware

3.2 Data Collection for Proposed controller design

A system response with maximum of four inputs and one output is recorded. Once the dataset is derived, it is passed on to the system for FCP extraction. The proposed method for FCP extraction is based on Genetic algorithm. In this algorithm, there are 40 system parameters that are varied according to genetic algorithm to achieve two objectives, namely

- Quick settling, and
- Tracking reference.

The simulink model with initial FCP is used as an objective function which returns the transient and settling time. The co-ordinates of the membership functions are nonlinear inequality constraints. Thus varying these FCP will provide a absolute tracking of reference[1].

Throughout this implementation process, data synchronization and communication between user and the hardware G-FLCS is critical as it can be observed in Fig. 5. This figure depicts how data communication can be achieved with the help of various control signals between, client-server and server-hardware G-FLCS. A web application provides a systematic user interface which collects data from authenticated users. The application waits for a new connection request on startup. On successful login, a user is presented with a web page containing four different tab-windows, each for basic parameters, information about inputs, outputs and Rulebase. These windows will be preloaded with extracted parameters as shown in Fig. 4 and can provide a way for fine tuning the control parameters by an operator. In Fig. 5, operations like connections, login, parameter collection and communication with H-FLC through a server program are performed by the web application and are shaded in *Light Grey*. When a user provides the fuzzy control parameters (FCP) data, the web application validates the entered data based on following protocols.

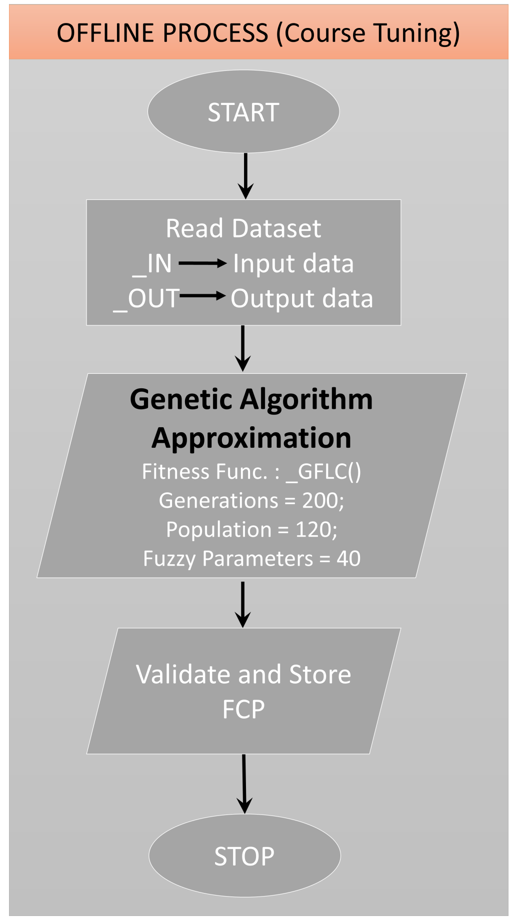


Figure 4: Fuzzy Control Parameter (FCP) extraction using Genetic Algorithm

- All MFs have properly defined co-ordinates within specified range of operation.
- Number of Input(s) and Outputs are correctly de-

[1]The codes can be downloaded from `https://goo.gl/Zb17fZ`

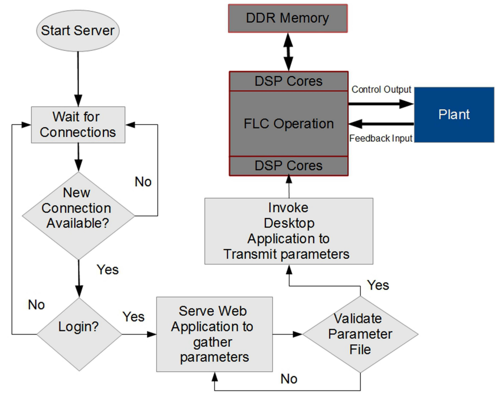


Figure 5: Dataflow of the proposed G-FLCS system

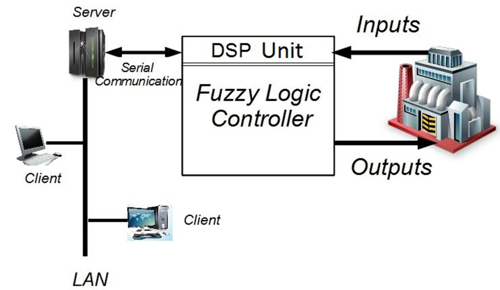
fined.

- Rules are validated according to Mamdani model.

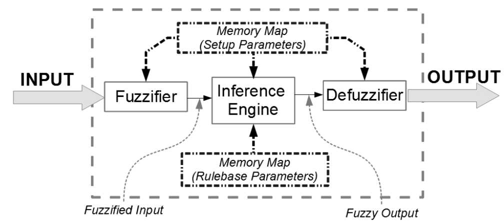
After validation of new parameters, the server application connects serially to the hardware G-FLCS. Serial communication protocol is used in this paper for the ease of implementation. This data communication can be easily extended to industry standard controller area network (CAN) protocol. G-FLCS completes current execution and generates control signal to the system. Thereafter, it acknowledges any incoming serial communication request and starts receiving and storing fuzzy parameters in the data memory. Sys/BIOS is widely used real time operating system for TI DSPs and has been used in this work to take care of the multitasking of fuzzy processes.

4 Fuzzy Logic Approximation Control(FLAC)

The proposed system architecture involves hardware-software co-design to present a complete reconfigurable G-FLC as shown in Fig. 6a and Fig. 6b. The WebUI in client server model represents the software and the driver layer to interface the H-FLC through serial port. The DSP hardware receives FCP data serially and stores them. These parameters are segregated in two categories namely *Setup* and *Rulebase* data. The driver layer in the hardware G-FLCS receives and acknowledges the data transmission.



(a) System Architecture



(b) Hardware Design

Figure 6: Proposed M-FRHC based H-FLC Design Architecture

5 Hardware Realization on TI C6748 DSP

5.1 Code Optimization

Code written in assembly (ASM) is processor-specific, C code can readily be ported from one platform to another. However, optimized ASM code runs faster than C and requires less memory space. Before optimizing, make sure that the code is functional and yields correct results. After optimizing, the code can be so reorganized and resequenced that the optimization process makes it difficult to follow. One needs to realize that if a C-coded algorithm is functional and its execution speed is satisfactory, there is no need to optimize further. But, this motivates us to stretch the optimization such that we can extract best possible efficiency from the system [15].

Step 1 G-FLC is programmed in C Language without using any compiler optimization levels.

Step 2 Intrinsic functions are used along with various optimization levels. Functions like minimum (min), maximum (max), product (prod), division (div) are written as intrinsic functions and called as necessary by the modules and submodules of G-FLCs.

Step 3 Thereafter, the profiler is used to determine and identify the functions and submodules that may need further optimization. Later these functions are converted to linear ASM.

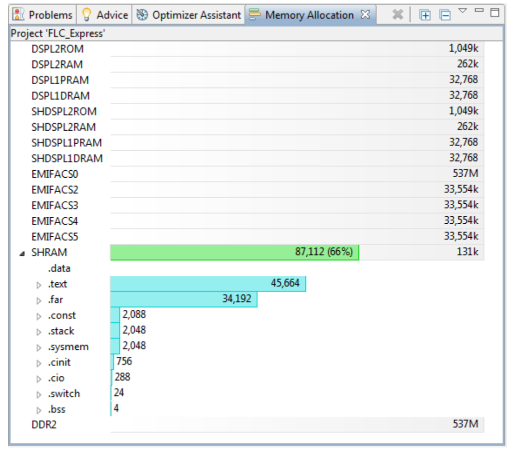
The compiler can insert calls to special functions in the run-time support library (RTS) to support operations that are not natively supported by the ISA. For example, the compiler calls $\_\_c6xabi\_divi()$ ($\_divi()$ in COFF) function to perform 32-bit integer divide operation. Such functions are called compiler helper functions, and result in a function call within the loop body. For example in G-FLC, the compiler accomplishes the division operation by calling the compiler helper function "$\_divi$" in fuzzification and defuzzification modules [15, 16, 17].

5.2 Code Implementation

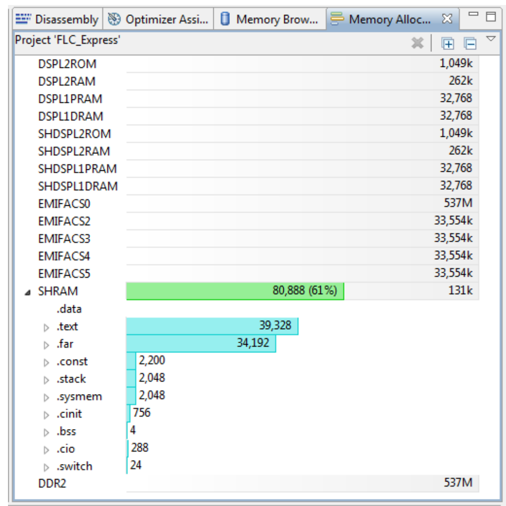
Code Composer Studio (CCS) is a proprietary integrated development environment developed by TI for programming DSP and ARM processors. The design is programmed in C language and optimized as described in section 5. Thereafter it is cross compiled using CCS v5.5 compiler and implemented on TMS320C6748 as target DSP processor. This system represents the hardware G-FLCS as discussed previously. G-FLCS is connected to a server PC using onboard UART and provides a platform which is capable of accepting FCP file to operate as a standalone tunable G-FLCS. Detailed memory utilization of the built code is shown in Fig. 7. It can be observed that the code size without optimization acquires 66% of the SHRAM and has a code size of 88K in comparison to 61% of the SHRAM and code size of 80K with code optimization technique described in 5.1.

6 Performance Analysis

A performance analysis between MPC, PID and proposed hardware G-FLCS is drawn after a hardware-in-loop simulation of Simulink model presented in Fig. 3 is completed. The ACC system response under influence of individual controllers is plotted in Fig. 8a It is evident from comparing responses of these controllers that the Fuzzy approximated PI controller has a very less settling time and a fast transient response. However, the fuzzy controller structure provides with an option to fine tune the control parameters on runtime. Moreover, mathematical model of the ACC plant was also not in the vicinity while formulation of fuzzy parameters were undertaken and Fig. 4 abundantly explains this. Hereby reducing the computational complexity which exponentially increases with increase in input variables. The control output from the proposed hardware G-FLCS closely matches with the control output from the Fuzzy Toolbox of Matlab. An mean



(a) Without Optimization



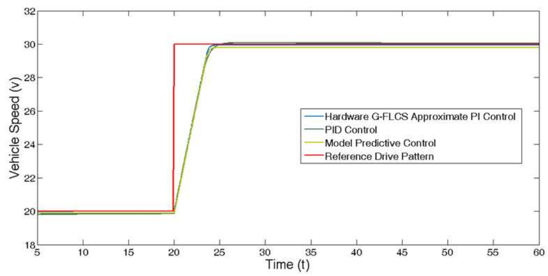
(b) Optimization

Figure 7: Memory Utilization of Proposed System Realized in TI C6748 DSP

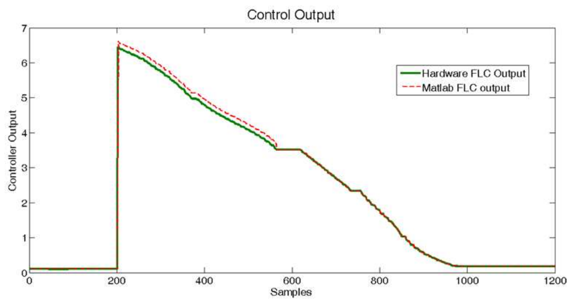
square error of 0.13 was recorded. Fig. 8b shows the control output of the two fuzzy logic controllers when operated on the ACC plant model.

7 Conclusion

It is evident from responses of the three controllers presented in Fig. 8a, proposed FLAC has a very less settling time and a fast transient response compared to others. Mathematical model of the plant was not in the vicinity while formulation of fuzzy parameters were undertaken. A parameter extraction technique inspired with genetic algorithm is implemented on an ACC process to provide the applicability of the proposed idea. The resultant hardware FLAC is found to have a low computational complexity and high code density and connects seamlessly to the local area network (LAN).



(a) System Response



(b) Control Output

Figure 8: (a) Response of ACC under influence of various controllers, (b) Control output of hardware G-FLCS and Matlab Fuzzy Toolbox with same FCP

References:

[1] T. Ross, *Fuzzy logic with engineering applications*, 3rd ed. John Wiley & Sons, Ltd, 2010.

[2] J. Mendel and G. Mouzouris, "Designing fuzzy logic systems," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 44, no. 11, pp. 885–895, 1997.

[3] J. a. Laghari, H. Mokhlis, A. H. Abu Bakar, and H. Mohamad, "A fuzzy based load frequency control for distribution network connected with mini hydro power plant," *Journal of Intelligent and Fuzzy Systems*, vol. 26, no. 3, pp. 1301–1310, 2014. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84893916663\&partnerID=tZOtx3y1

[4] M. J. Nalley and M. B. Trabia, "Control of overhead cranes using a fuzzy logic controller," *Journal of Intelligent and Fuzzy Systems*, vol. 8, no. 1, pp. 1–18, 2000. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-0034393150\&partnerID=tZOtx3y1

[5] E. Ramadan, M. El-bardini, and M. Fkirin, "Design and FPGA-implementation of an improved adaptive fuzzy logic controller for DC motor speed control," *Ain Shams Engineering Journal*, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2090447914000446

[6] A. H. Zavala and O. C. Nieto, "Fuzzy hardware: A retrospective and analysis," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 4, pp. 623–635, Aug. 2012. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84864652589\&partnerID=tZOtx3y1

[7] G. Bosque, I. Del Campo, and J. Echanobe, "Fuzzy systems, neural networks and neuro-fuzzy systems: A vision on their hardware implementation and platforms over two decades," *Engineering Applications of Artificial Intelligence*, vol. 32, pp. 283–331, Jun. 2014. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84900502119\&partnerID=tZOtx3y1

[8] S. Paul Sathiyan, S. Suresh Kumar, and A. Immanuel Selvakumar, "Particle Swarm Optimization technique for rule base optimization of FLC for low speed ACC vehicle," *ARPN Journal of Engineering and Applied Sciences*, vol. 9, no. 6, pp. 981–987, 2014. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84902997634\&partnerID=tZOtx3y1

[9] A. Hassan and G. Collier, "Adaptive cruise control for a robotic vehicle using fuzzy logic," in *Mechatronics 2013: Recent Technological and Scientific Advances*. springer berlin, 2014, pp. 535–542. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84903771470\&partnerID=tZOtx3y1

[10] A. Merah and K. Hartani, "A robust longitudinal control strategy for safer and comfortable automotive driving," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 7, no. 23, pp. 5026–5033, 2014. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84906076376\&partnerID=tZOtx3y1

[11] A. Vahidi and A. Eskandarian, "Research advances in intelligent collision avoidance and adaptive cruise control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, no. 3, pp. 132–153, Sep. 2003. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-10744230637\&partnerID=tZOtx3y1

[12] K. J. Aström and R. M. Murray, *Feedback Systems: An Introduction for Scientists and Engineers*, 2nd ed. Princeton University Press, 2009.

[13] T. Schwickart, H. Voos, J.-R. Hadji-Minaglou, M. Darouach, and A. Rosich, "Design and simulation of a real-time implementable energy-efficient model-predictive cruise controller for electric vehicles," *Journal of the Franklin Institute*, vol. 352, no. 2, pp. 603–625, Feb. 2015. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84922838045\&partnerID=tZOtx3y1

[14] M. C. Martinez-Rodriguez, P. Brox, and I. Baturone, "Digital VLSI Implementation of Piecewise-Affine Controllers Based on Lattice Approach," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 3, pp. 842–854, May 2015. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-84928391148\&partnerID=tZOtx3y1

[15] Texas Instruments, "TMS320C6000 Optimizing Compiler v7.6," Texas Instruments, Tech. Rep., 2014. [Online]. Available: http://www.ti.com/lit/ug/spru187v/spru187v.pdf

[16] R. Chassaing, *Digital Signal Processing and Applications with the C6713 and C6416 DSK*, 1st ed. John Wiley & Sons, Ltd, 2005.

[17] D. Batten, S. Jinturkar, J. Glossner, M. Schulte, and P. D'Arcy, "New approach to DSP intrinsic functions," in *Proceedings of the Hawaii International Conference on System Sciences*. IEEE, 2000, p. 214. [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-0033871853\&partnerID=tZOtx3y1