

**PATHWAYS FOR OPTIMIZATION-BASED
DRUG DELIVERY SYSTEMS AND DEVICES**

Leonidas Bleris* Panagiotis Vouzis***
Mark V. Arnold*** Mayuresh V. Kothare**.¹

* *Department of Electrical and Computer Engineering
Lehigh University, Bethlehem, PA, 18015*
** *Department of Chemical Engineering
Lehigh University, Bethlehem, PA, 18015*
*** *Department of Computer Engineering
Lehigh University, Bethlehem, PA, 18015*

Abstract: Drug synthesis and discovery represents today one of the most rapidly evolving scientific areas. This is primarily due to the interdisciplinary collaboration between chemists, pharmacologists, molecular biologists, and biochemists. A direct implication of the developments in drug discovery is the need for novel drug delivery systems and devices. Considering the advances in engineering disciplines and micro/nano technology the potential for producing new drug delivery devices is substantial. Notably, so is the necessity of these devices in creating solid commercial value propositions for the medical markets. In this work we present research results related to embedding optimization-based control on-a-chip for drug delivery applications.

Keywords: Embedded Control Systems, Real-Time Model Predictive Control, Drug Delivery Devices

1. INTRODUCTION

Today, drugs can be delivered in many different ways including: orally (pills or suspensions), through the vein (intravenously), through the artery (arterially), topically through the skin (transdermally), through the rectum (suppository), through the eye (ocular), through the lungs (inhaled), by injection into the skin (subcutaneously), by injection into the muscle (intramuscularly), and under the tongue (sublingually). By means of improving the drug delivery devices, companies and researchers aim to meet particular goals (Brunner, 2004), most of them related to

the state of the patients. These goals include: improved efficacy, reduced side effects, continuous dosing, reduced pain from administration, increased ease of use, increased use compliance, improved mobility, and decreased involvement of healthcare workers.

Controlled drug delivery (Langer, 2004) currently involves control of the time course or the location of drug delivery. In this work we examine alternative pathways for the implementation of optimal drug delivery control (of the time course) on an embedded target, for glucose regulation by means of insulin delivery. We provide research results of this implementation on an Application Specific Integrated Circuit (ASIC), on a general purpose processor, and finally on a Field Programmable Gate Array (FPGA).

¹ Partial financial support for this research from the US National Science Foundation grant CTS-0134102 and the Pittsburgh Digital Greenhouse is gratefully acknowledged.

Optimization-based control schemes can be used in order to effectively control nonlinear and multivariable models, and to impose constraints on both the control action and the states. The constraints are usually set on the bounds and the rate of change, and can be incorporated in the form of equalities or inequalities. These control algorithms typically involve the solution of an optimization problem that explicitly incorporates knowledge of a dynamic model of a process, with the addition of design and process objectives. This concept has seen widespread use, particularly through applications of predictive control. The advantages of Model Predictive Control (MPC), such as the ability to handle constraints, the applicability to nonlinear processes and to multivariable problems, constitute this control method an ideal choice for satisfying most of the design objectives of a drug delivery device.

Currently, companies that want to sell insulin delivery devices must illustrate to the Food and Drug Administration (FDA) that their devices are “substantially equivalent” to drug delivery devices already for sale. MPC controllers (device) and the drug (insulin) are approved and are on the market individually. The use of insulin is the standard method for regulating glucose. Thus the toxicology of the pharmaceutical is known. Nevertheless, implementations of MPC have been on the market mainly for chemical plants, and on workstation implementations. In 2004, the FDA announced that the world's first implantable Radio Frequency Identification (RFID) microchip for human has been cleared for medical uses in the United States. About the size of a grain of rice, “VeriChip” is a subdermal radio frequency microchip. The device has no power supply, and it is activated when a scanning device runs across the skin above it. A tiny transmitter on the chip then releases patient-specific information. Although not capable of carrying out arithmetic operations such technology is available, and one of the biggest regulatory thresholds was surpassed. Thus, it is a matter of time before new generations of similar devices seek FDA approval. There are however other issues to be examined: the performance robustness of the chip, the biocompatibility of the device, and that the stability and bioavailability of the drug have not been compromised by the drug-device combination.

Although the solution to glucose regulation problems may appear in the form of an implantable capsule (Leoni and Desai, 2001) that continuously produces insulin and releases it to the bloodstream, there are numerous potential applications for medical controllers on-a-chip including: control of physiological processes, muscle control, respiration control, drug infusion control (for instance during anesthesia), cardiac pacemakers and de-

fibrillators, heart rate control, blood flow and pressure control, HIV control, and neurological implants. For example, in the Human Immunodeficiency Virus (HIV) control case, where the drug cocktail provided is currently expensive, the regulation (with the solution of an optimization problem) of the dosage to the absolute minimum is highly desirable. Another example is a device that detects irregular heart pulses and uses a control algorithm to regulate doses of a blood thinning drug (for example aspirin). This device maintains the risks of thrombosis at low levels while minimizing the chances of internal bleeding.

2. BLOOD GLUCOSE CONTROL IN DIABETIC PATIENTS

Diabetes mellitus is a chronic health condition where the human body is unable to produce insulin and properly breakdown sugar (glucose) in the blood. The insufficient insulin production or lack of responsiveness to insulin, results to hyperglycemia (blood glucose levels over 120mg/dL). There are two primary types of diabetes mellitus, type I (insulin-dependent or juvenile-onset), which may be caused by an autoimmune response, and type II (non-insulin-dependent or adult-onset). Symptoms include hunger, thirst, excessive urination, dehydration and weight loss. Complications can include heart disease, stroke, neuropathy, poor circulation leading to loss of limbs, hearing impairment, vision problems and death.

The treatment of diabetes requires regular insulin injections, proper nutrition and exercise in order to maintain normoglycemia, defined as blood glucose 70–100mg/dL. Insulin and glucagon are the hormones responsible for glucose regulation. Both insulin and glucagon are secreted from the pancreas, and thus are referred to as pancreatic endocrine hormones. Most of the long-term complications associated with diabetes result from sustained hyperglycemia, but hypoglycemia can result in very acute symptoms such as coma and death.

A significant effort has been devoted to the development of closed-loop controllers for blood glucose control. For approaches to applying control to diabetic subjects the reader is referred to (Parker *et al.*, 1999; Doyle-III *et al.*, 2000; Rubb and Parker, 2003; Parker and Doyle-III, 2001).

2.1 Minimal Glucose Model

Minimal models of glucose and insulin plasma levels have been developed (Bergman *et al.*, 1979; Pacini and Bergman, 1986) for humans using

Frequently-Sampled Intravenous Glucose Tolerance (FSIGT) tests. During a FSIGT test, a single intravenous injection of glucose is given to a fasting subject and blood samples are collected at regular timed intervals. The blood samples are then analyzed for glucose and insulin concentration. Figure 1 shows a typical response from a normal subject.

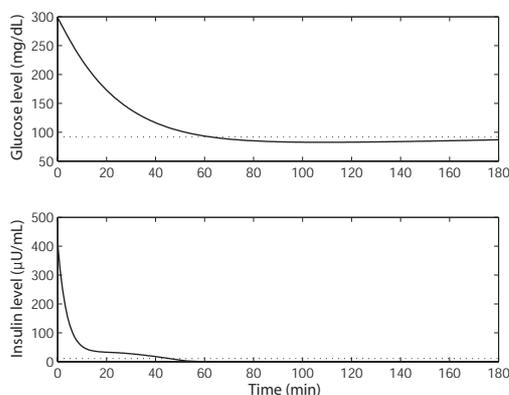


Fig. 1. Typical glucose and insulin response from a normal subject.

As illustrated in Figure 1, the glucose level in plasma is at a peak at the time of the injection, drops to a minimum which is below the basal glucose level, and then gradually returns to the basal level (dashed line). The insulin level in plasma rapidly rises to a peak immediately after the injection, drops to a lower level which is still above the basal insulin level, rises again to a lesser peak, and then gradually drops to its basal level. Depending on the state of the subject, there can be wide variations from this response that can determine the condition of the patient (G. M. Steil and Bergman, 1993).

The glucose minimal model involves two physiologic compartments: an interstitial tissue compartment and a plasma compartment. The differential equations corresponding to the two compartments are:

$$\frac{dG(t)}{dt} = k_1(G_b - G(t)) - X(t)G(t) \quad (1)$$

$$\frac{dX(t)}{dt} = k_2(I(t) - I_b) - k_3X(t) \quad (2)$$

where t is time, $G(t)$ is the plasma glucose concentration at time t , $I(t)$ is the plasma insulin concentration at time t , and $X(t)$ is the interstitial insulin at time t , with $G(t_0)=G_0$ and $X(t_0)=0$. G_b is the basal plasma glucose concentration and I_b is the basal plasma insulin concentration. The insulin sensitivity is defined as $S_I = k_2/k_3$ and the glucose effectiveness is defined as $S_G = k_1$. Basal plasma concentrations of glucose and insulin are typically measured either before or 180 minutes after the administration of glucose.

There are four unknown parameters in this model: k_1, k_2, k_3 , and G_0 that depend on the particular subject and can be estimated experimentally. We use the parameters adopted from (Pacini and Bergman, 1986; Riel, 2004).

3. MODEL PREDICTIVE CONTROL

Model Predictive Control originated in the chemical process industries. The main advantages of MPC are the ability to handle constraints and its applicability to multivariable nonlinear processes. Because of the computational requirements of the optimizations associated with MPC, it has primarily been applied to plants in the process industry, with slow dynamics. Furthermore, existing implementations of MPC typically perform numerical calculations using workstations in 64-bit Floating Point (FP) arithmetic, which is too expensive, power demanding and large in size. Therefore the implementation of real-time embedded model predictive control, for systems with fast dynamics, where the size and the application precludes the use of a dedicated workstation, presents new technological challenges.

Controllers belonging to the MPC family are generally characterized by the following steps: initially the future outputs are calculated at each sample interval over a predetermined horizon N , the prediction horizon, using a process model. These outputs $y(t+k|t)$ for $k=1\dots N$ depend up to the time t on the past inputs and on the future signals $u(t+k|t)$, $k=0\dots N-1$ which are those to be sent to the system. The next step is to calculate the set of future control moves by optimizing a determined criterion, in order to keep the process as close as possible to a predefined reference trajectory. This criterion is usually a quadratic function of the difference between the predicted output signal and the reference trajectory. In some cases, in order to minimize the control effort the control moves $u(t+k|t)$ are included in the objective function:

$$J_P(k) = \sum_{k=0}^P \{ [y(t+k|t) - y_{ref}]^2 + Ru(t+k|t)^2 \} \quad (3)$$

$$|u(t+k|t)| \leq b \quad , \quad k \geq 0 \quad (4)$$

where $y(t+k|t)$ are the predicted outputs, y_{ref} is the desired set reference output, $u(t+k|t)$ the control sequence and R is a the weighting on the control moves, a design parameter. This system is subject to input constraints given by the vector b . Finally, the first control move $u(t|t)$ is sent to the system while the rest are rejected. At the next sampling instant the output $y(t+1)$ of the system is used in the optimization using feedback and the procedure is repeated so that we get an updated control sequence.

4. EMBEDDED MODEL PREDICTIVE CONTROL

Some initial results have been reported to the direction of embedding Model Predictive Control. Further analysis and references can be found at (Bleris *et al.*, 2006a). With the following subsections we report our recent research results, and we focus on the FPGA implementation.

4.1 General purpose processor

We have examined a general purpose processor implementation (Bleris and Kothare, 2005b; Bleris and Kothare, 2005a). We used a single board computer phyCORE-MPC555 that packs the power of Motorola's embedded 32-bit MPC555 microcontroller within a miniature footprint. The MPC555 is a high-speed 32-bit Central Processing Unit (CPU) that contains a 64-bit floating point unit designed to accelerate advanced algorithms, running at 40MHz. To implement the optimization algorithm of MPC, we used a combination of software tools: CodeWarrior Integrated Development Environment (IDE), MATLAB, Real-Time Workshop, and SIMULINK.

In order to test the performance of a multi-model MPC formulation in real time we used Processor in the Loop (PIL) co-simulations; the MPC chip in closed loop with the monitored patient (the minimal model on the host workstation). To examine the influence of the control horizon on the computational costs of MPC running on the Motorola processor, we set the number of optimizations at a fixed number. In Figure 2 we provide profiling results for different cases of prediction horizon and optimization steps. Additionally, in Figure 3 we provide the performance of MPC for different number of optimizations, keeping fixed the prediction and control horizons. As expected the computational time for this case grows linearly with the optimizations. Both these profiling results illustrate that we remain in all examined cases at under one second for the computation of the optimal insulin dosage using MPC. From interpolation of the results of Figure 3 we obtain the required time for one optimization loop, (a) 53msec for prediction horizon of 21 and control horizon of 7, (b) 95msec for prediction horizon of 31 and control horizon of 7.

4.2 Application-Specific Instruction Processor

For the Application Specific Integrated Circuit (ASIC) implementation (Bleris *et al.*, 2006a; Bleris *et al.*, 2005) we proposed the following design framework. By emulating the microcontroller arithmetic operations, we reduce the precision of

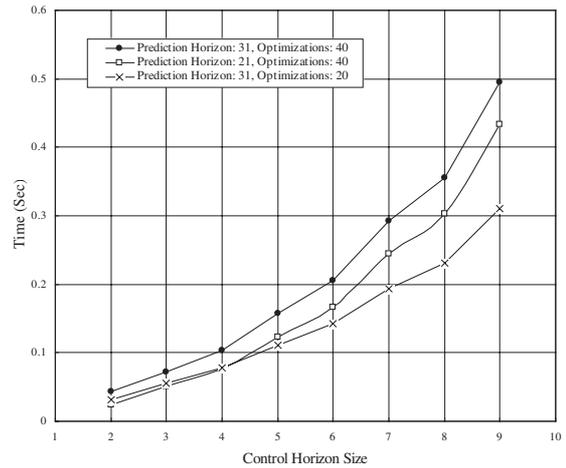


Fig. 2. Profiling results using as a variable the control horizon for different number of optimizations and prediction horizons.

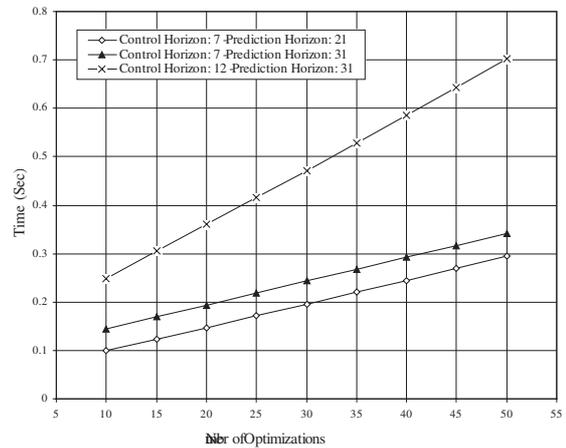


Fig. 3. Profiling results for fixed prediction and control horizon sizes and variable number of optimizations.

the microprocessor to the minimum, while maintaining stable control performance for a particular control application. This reduction is accomplished by series of parametric tests using different word sizes and utilizing computational tools to simulate the controlled model. Taking advantage of the low precision, a Logarithmic Number System (LNS) based micro-processor architecture was proposed (Garcia *et al.*, 2004) (Figure 4) that provides energy, computational cost, and price savings (Figure 5). This reduced-precision ASIC can achieve sampling speeds as low as 32msec for relatively large problems. Additionally, to quantify the advantage of reducing the precision, estimations for both 64-bit FP and 16-bit LNS circuits showed that for an arithmetic unit that computes addition, subtraction, multiplication and division, the size required is about 17 times larger for 64-bit FP.

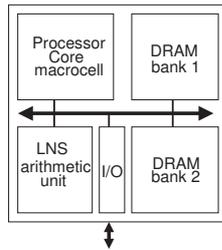


Fig. 4. Architecture block.

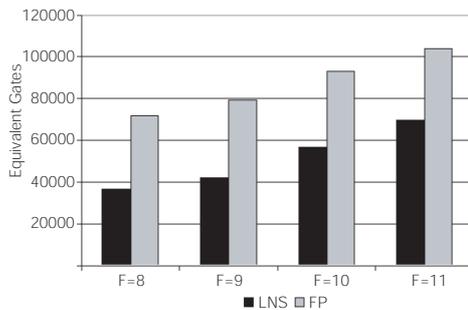


Fig. 5. Arithmetic Logic Unit (ALU) areas for LNS and FP at different precisions.

4.3 Mixed software-hardware embedded controller

For the mixed software-hardware embedded controller we examine a codesign (Vouzis *et al.*, 2005; Bleris *et al.*, 2006b) step before the actual implementation that decomposes the algorithm into two parts. One that fits into the host processor and one that fits into the custom made unit that performs all the (repetitive and computationally demanding) arithmetic operations. The selected microprocessor acting as a host for our design is the 16-bit Extensible Instruction Set Computer (EISC) from ADCUS, Inc. For prototyping we use the Field Programmable Gate Array (FPGA) Virtex-4 XC4VLS25 device of Xilinx interfacing with Matlab, running on a PC workstation, in order to implement Processor-In-the-Loop (PIL) techniques that help to test and debug the embedded system. Both the ADCUS microprocessor and the matrix co-processor are described in Verilog and the whole design is synthesized with the ISE 7.1 design environment of Xilinx.

A field-programmable gate array is a large-scale integrated circuit that can be programmed after it is manufactured rather than being limited to a predetermined, unchangeable hardware function. FPGAs come in a wide variety of sizes and with many different combinations of internal and external features. What they have in common is that they are composed of small blocks of programmable logic. These basic blocks may be replicated many thousands of times to create a large programmable hardware fabric. In more complex FPGAs these general-purpose logic blocks are combined with higher level arithmetic and control

structures, such as multipliers and counters, in support of common types of applications such as signal processing.

Defining the behavior of an FPGA (the hardware that it contains) has traditionally been done either using a Hardware Description Language (HDL) such as VHDL or Verilog or by arranging blocks of pre-existing functions, whether gate-level logic elements or higher-level macros, using a schematic- or block diagram-oriented design tool. Hardware applications implemented in FPGAs are generally slower and consume more power than the same applications implemented in custom ASICs. Nonetheless, the dramatically lowered risk and cost of development for FPGAs have made them excellent alternatives to custom Integrated Circuits (ICs). The reduced development times associated with FPGAs often makes them compelling platforms for ASIC prototyping as well.

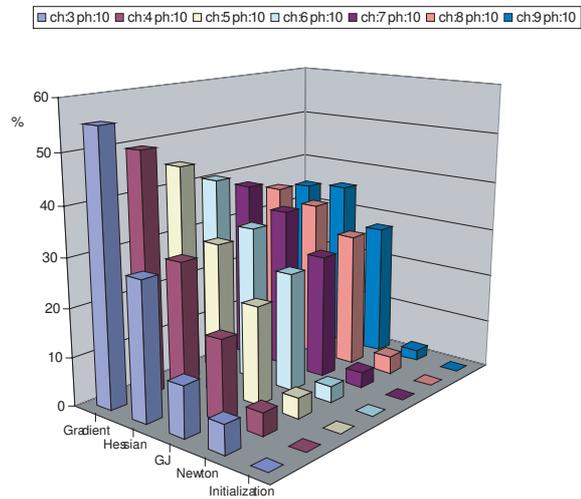


Fig. 6. Profiling results for prediction horizon 10 and variable control horizon.

We use Newtons algorithm to solve the optimization problem by combining the constraints into the cost function using barrier functions. This optimization algorithm consists of five functions which can be considered as the five basic operational blocks. These functions are: the initializations prior to each iteration loop of the optimization, the calculation of the Gradient vector and the Hessian matrix, the Gauss-Jordan matrix inversion, and finally the optimal move calculation using Newtons iteration.

In order to partition the algorithm to hardware and software we examine the behavior of these operational blocks using a profiler. In Figure 6 we present the profiling results of the five operational blocks, for a prediction horizon of 10 and variable control horizons. A direct observation is that the computation of the Gradient and the Hessian requires approximately 70 – 80% of the

total optimization time. The next most expensive function is the matrix inversion which can take up to 30% of the total time, for large control horizons. Furthermore, we observe that for small control horizons the Gradient function uses almost half of the total optimization time and double the time of the Hessian function. Finally, we observe that by increasing the control horizon size, the matrix inversion becomes more expensive, and the computational time required by the Gradient and Hessian functions converge. This higher level analysis of the MPC optimization code reveals that the repetitive matrix operations of the Gradient and Hessian, comprise the major part of the processing. Therefore these specific matrix operations are the main part that has to be implemented efficiently, while the rest of the operations can be performed by a general purpose microprocessor.

5. CONCLUDING REMARKS

A selection of research results on embedding MPC were presented in this paper. These include profiling results of a general purpose processor, synthesis estimates of an ASIC chip, and codesign considerations for a co-processor FPGA implementation. During the presentation of the paper we intend to provide the results of the MPC running on the FPGA.

REFERENCES

- Bergman, R. N., Y. Z. Ider, C. R. Bowden and C. Cobelli (1979). Quantitative estimation of insulin sensitivity. *American Journal of Physiology* **236**, E667–E677.
- Bleris, L. G. and M. V. Kothare (2005a). Implementation of Model Predictive Control for Glucose Regulation using a General Purpose Microprocessor. In: *44th IEEE Conference on Decision and Control and European Control Conference*. Seville, Spain.
- Bleris, L. G. and M. V. Kothare (2005b). Real-time implementation of model predictive control. In: *2005 American Control Conference*. Portland, OR. pp. 4166–4171.
- Bleris, L. G., J. G. Garcia and M. V. Kothare (2005). Model predictive hydrodynamic regulation of microflows. In: *2005 American Control Conference*. Portland, OR. pp. 1752–1757.
- Bleris, L. G., M. V. Kothare, J. G. Garcia and M. G. Arnold (2006a). Towards embedded model predictive control for system-on-a-chip applications. *Journal of Process Control* **16**, 255–264.
- Bleris, L. G., P. Vouzis, M. G. Arnold and M. V. Kothare (2006b). Submitted: A Co-Processor FPGA Platform for the Implementation of Real-Time Model Predictive Control. In: *2006 American Control Conference*. Minneapolis, MI.
- Brunner, C. S. (2004). Challenges and opportunities in emerging drug delivery technologies. Product Genesis Inc.
- Doyle-III, F. J., R. S. Parker and E. P. Gatzke (2000). Advanced model predictive control for type I diabetic glucose control. In: *Proceedings of the 2000 American Control Conference*. Chicago, IL.
- G. M. Steil, A. Volund, S. E. Kahn and R. N. Bergman (1993). Reduced sample number for calculation of insulin sensitivity and glucose effectiveness from the minimal model. *Diabetes* **42**, 250–256.
- Garcia, J. G., M. G. Arnold, L. G. Bleris and M. V. Kothare (2004). LNS architectures for embedded model predictive control processors. In: *2004 International Conference on Compilers, Architectures and Synthesis for Embedded Systems*. Washington, D.C.. pp. 79 – 84.
- Langer, R. (2004). Transdermal drug delivery: past progress, current status, and future prospects. *Advanced Drug Delivery Reviews* **56**, 557–558.
- Leoni, L. and T. A. Desai (2001). Nanoporous biocapsules for the encapsulation of insulinoma cells: Biotransport and biocompatibility considerations. *IEEE Transactions on Biomedical Engineering* **48**, 1335–1341.
- Pacini, G. and R. N. Bergman (1986). A computer program to calculate insulin sensitivity and pancreatic responsiveness from the frequently sampled intravenous glucose tolerance test. *Computer Methods and Programs in Biomedicine* **23**, 113–122.
- Parker, R. S. and F.J. Doyle-III (2001). Control-relevant modeling in drug delivery. *Advances in Drug Delivery Reviews* **48(2)**, 211–248.
- Parker, R. S., F. J. Doyle and N. A. Peppas (1999). A model-based algorithm for blood glucose control in type I diabetic patients. *IEEE Transactions on Biomedical Engineering* **46(2)**, 148157.
- Riel, N. V. (2004). Minimal models for glucose and insulin kinetics. In: *Technique Report*. Eindhoven University of Technology.
- Rubb, J. D. and R. S. Parker (2003). Glucose control in type I diabetic patients: A volterra model-based approach. In: *Proceedings of the 2003 ADCHEM*. Hong Kong, China.
- Vouzis, P., L. G. Bleris, M. V. Kothare and M. G. Arnold (2005). Towards a Co-design Implementation of a System for Model Predictive Control. In: *2005 AIChE Annual Meeting*. Cincinnati, OH.