A Neuron-Based Pulse Servo for Motion Control

DeWeerth, Steve; Nielsen, Lars; Mead, Carver; Åström, Karl Johan

1990

Document Version:
Publisher's PDF, also known as Version of record

Link to publication

Citation for published version (APA):

General rights
Unless other specific re-use rights are stated the following general rights apply:
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.
• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal

Read more about Creative commons licenses: https://creativecommons.org/licenses/

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
A Neuron-based Pulse Servo for Motion Control

Steve DeWeerth
Lars Nielsen
Carver Mead
Karl Åström

Department of Automatic Control
Lund Institute of Technology
March 1990
**Title and subtitle**

A Neuron-based Pulse Servo for Motion Control

**Abstract**


**Key words**

**Classification system and/or index terms (if any)**

**Supplementary bibliographical information**

**ISSN and key title**

**Language**  |  **Number of pages**  |  **Recipient’s notes**
--- | --- | ---
English | 7 |  

The report may be ordered from the Department of Automatic Control or borrowed through the University Library 2, Box 1010, S-221 03 Lund, Sweden, Telex: 33248 lubis lund.
A Neuron-based Pulse Servo for Motion Control

Steve DeWeerth*, Lars Nielsen**, Carver Mead*, Karl Åström **

*Department of Computer Science, California Institute of Technology, Pasadena, Ca 91125, USA
**Department of Automatic Control, Lund Institute of Technology, Box 118, 221 00 Lund, Sweden

Abstract Sensory control based on biological computational paradigms can be implemented using low-power analog very large-scale integrated circuitry. We describe a design frame and a set of circuit elements with which generic motor controllers can be implemented. We then embed a simple proportional-derivative motor controller in the design frame, and describe its performance advantages over a more traditional controller. We also discuss the merits of biologically inspired systems used in applications related to robotics.

1. Introduction

In recent years, the engineering community has demonstrated a great deal of interest in biological systems. One aim is complete artificial systems whose organizing principles are inspired by those found in biology. Such a system must be implemented in a technology that can incorporate neural-like processing. Current modeling of neural systems using analog VLSI (Mead, 1989) offers practical solutions to engineering problems with complex functionalities. Both systems (biological and artificial) use the properties of physical devices as computational primitives, and both technologies are limited by the interconnections rather than by the computations themselves. The result can be a vast amount of computation in a few square millimeters, and a few milliwatts.

Robotics is one field in which this biological inspiration could obviously be exploited. Biological motor systems can perform very gentle and precise motions even though the individual components, neurons and muscles, are imprecise. This is accomplished by using highly redundant systems that greatly improve accuracy and fault tolerance. Available robotic manipulators, on the other hand, are usually built from high precision components, but can only mimic the dexterity of a human hand in restricted cases.

In this paper, we describe a first achievement in this direction in which a new design methodology is used for the control of a friction limited servo. The results suggest that the size and reliability of the motion controller should be applicable to tasks such as the control of a robot hand built from low precision components like actuators designed with high friction load screws.

2. The Control Problem

Human motion control is a computational process, and so is conventional motion control in robotics. We will now discuss some aspects of these computational metaphors that have inspired the current effort.

Computation and control

Traditionally, technology has strongly influenced how engineers think about control systems. Implementations using relays, operational amplifiers, and microprocessors all have individual requirements that affect their uses. The microcomputer revolution, for example, has had a major impact on control system implementations. Because of shrinking cost and size combined with increases in performance, even single loop controllers are now implemented with microprocessors. As a consequence, control theory has developed in new directions, and perhaps more important, more advanced concepts like adaptive control have become popular (Åström and Wittenmark, 1984, 1989).

Biological systems

As with other implementation media, neurobiology influences the computation used to solve a given control problem. The inherent imprecision in neurons and neural communication pathways requires system-level organization
that is quite different than traditional engineering implementations. Vertebrate nervous systems take advantage of parallelism, aggregation, and redundancy to make up for low-level imprecision. By using this type of high-level organization neural systems attain excellent system precision.

Biological organization characteristics put requirements on any physical medium capable of neural-like computations. First, the physical size of individual computational devices must be small to facilitate highly parallel systems. Also, the power generated by each device must be sufficiently small, allowing the system to dissipate the power generated by densely packed devices. Finally, since neurons aggregate large numbers of signals, there must be some mechanism to keep the resulting signals within operational range. Another important aspect of biological systems is the fact that information is transferred via nerve pulses. It is generally accepted that the pulse representation of the information improves reliability in the transfer of information. However, nature also makes a virtue out of using pulses when controlling muscles. Muscles are damped systems with a fair amount of internal friction (Mansson, 1989). We will later point out the advantages of pulses when driving friction limited systems.

A novel approach
If we are to use neural computational paradigms to implement control systems, we must first have a suitable technology in which to implement these systems. Mead (1989) has introduced a new highly-parallel low-power analog computing paradigm. We will in the following see that, by using this paradigm, it is possible to make regulators with a small number of circuit elements. These systems require a silicon surface which is many orders of magnitude smaller than a conventional system with A/D, CPU, Memory, and D/A. The system can also be made very reliable e.g. by duplicating the circuit elements, so that a component failure will only lead to a slight degradation in the system performance. Beside these advantages related to cost and function, we believe that the view on computational metaphors for control will be influenced by the possibilities inherent in the technology.

3. System Implementation
In the previous section, we described three criteria necessary for any technology capable of implementing highly redundant systems: small physical size, low power, and large signal range (highly compressive functionality). Mead (1989) has presented a design paradigm using analog integrated circuits which operate primarily in the subthreshold (weak inversion) region of transistor operation. This paradigm meets our three criteria quite well: First, systems can be designed with subthreshold MOS devices which use orders of magnitude less power than there counterparts using either standard (superthreshold) analog or digital design techniques. Also, these systems can be designed using a very small amount of silicon area with large, highly redundant systems fitting on a fraction of a standard CMOS die. Finally, due to their exponential nature, subthreshold MOS transistors operate with signals over many orders of magnitude, allowing for the result of the aggregation of large numbers of signals to remain in the operating range of the devices.

Another advantage of this design methodology is the fact that both currents and voltages are used as signal types, giving more varied computational metaphors than standard analog systems which primarily use voltages. These facts make it especially easy to aggregate signals through current summing. In fact our approach is inspired by biological neurons, in which there is significant analog processing and aggregation leading to the possible pulse firing from the cell body. Using neurally inspired ideas and a small number of basic circuit elements, we have developed a design methodology for the design of fault-tolerant analog control systems. This section presents this "design frame" along with the necessary circuits for its implementation.

The Design Frame
There are two restrictions on the design frame into which we embed our systems. First, it must be flexible enough to allow us to implement a number of different types of regulators (i.e., multiple input, cascaded, etc.). It must also allow for the implementation of large-scale redundancy. These characteristics can be accomplished using a model for aggregation which is implemented as a weighted linear combination of differential inputs. This aggregation can be described by

$$ y = \sum_{n} k_n (r_n - z_n) $$

where $k_n$ is the proportionality constant and $r_n - z_n$ is the differential input for the nth stage. Since the sections of a standard regulator usually consist of a functional element multiplied by a constant, this system allows implementation of any elementary control element as long as its signal can be turned into a differential input. (In the next section, we will demonstrate how this can be accomplished for both the trivial case of a proportional element and the slightly more complex case of a derivative element.) Our system also allows for simple replication of the individual functional elements.

For the implementation described in this paper, the input signals are differential voltages. Also, for the purpose of aggregation, an intermediate current representation is used. Finally, as described earlier the output is encoded as pulses to be used to drive the actuators. The following subsections will describe the circuit elements necessary to implement this system, and then describe the actual circuit implementation of the design frame.
Transconductance amplifiers

The transconductance amplifier (Vittoz, 1985), shown in Figure 1, is a differential stage that produces an output current proportional to its differential input voltage $V_1 - V_2$ multiplied by its bias current $i_b$. This circuit consists of a bias transistor that sets the maximum output current of the amplifier, a differential pair of transistors, and a current mirror. The voltage $V_b$ sets the bias current $i_b$. This current is divided into $i_1$ and $i_2$ by the differential pair transistors according to input voltages $V_1$ and $V_2$. The current $i_2$ is directly subtracted from the output node. The current mirror replicates $i_1$ and adds it to the output node. Thus, the output current is the difference between the currents in the two branches of the differential pair: $i_{out} = i_1 - i_2$. The transconductance amplifier is generally used with very small bias currents, and operates well with all transistors in the sub-threshold regime. The output current for this circuit (shown in Figure 1) when the bias current is sub-threshold is

$$i_{out} = i_b \tanh \left( \frac{V_1 - V_2}{2} \right)$$

with voltages given in units of $V_1/\kappa$ where $V_1$ is the thermal voltage and $\kappa$ is the gate voltage efficiency. Saturation behavior is clearly exhibited for large differential input voltages ($|V_1 - V_2| > 2V_1/\kappa$), at which point nearly all of the bias current is flowing through one of the transistors in the differential pair. For small differential voltages, the transfer equation can be approximated as

$$i_{out} \approx g(V_1 - V_2)$$

with $g = i_b \frac{2V_1}{\kappa}$ is the amplifier transconductance. We consider the linear approximation valid when $|V_1 - V_2| < 2kT/q\kappa \approx 100\text{mV}$.

The functionality of the transconductance amplifier can be extended by mirroring both differential pair currents ($i_1$ and $i_2$) to form a wide-range transconductance amplifier. The traditional reason for this extension is the isolation the output node from both differential pair transistors allowing a wider range of output voltages. However, this extension also facilitates multiple output currents. A wide-range amplifier with two outputs is shown in Figure 2. The output currents are complements of each other with $i_1 \approx g(V_1 - V_2)$ and $i_2 \approx g(V_2 - V_1)$. This variation on the transconductance amplifier is used in the implementation of the design frame.

The neuron circuit

Figure 3 shows a circuit that converts an input current into a series of pulses whose frequency is a monotonic function of the input current (Mead 1989). This circuit is named the neuron after the cells that generate nerve pulses in biological systems. The circuit consists of an input current mirror, a non-inverting high gain stage (two standard CMOS inverters), a pair of reset transistors, and two capacitors.

Figure 4 shows the response of the input and output voltages during pulse generation. The current mirror sources the current $i_n$ onto the input node. When the input capacitance integrates enough charge to reach $V_{in}$ (the switching threshold of the input inverter), the inverters switch and the output goes high. This transient is fed back to the input and capacitively divided be-
This rectified and converted a pair of output complementary amplifiers. Each amplifier has the equation:

\[ G_2 \left( \frac{V_{in} - V}{V_{out}} \right) = C \frac{dV}{dt} \]

By rearranging this equation, we can derive the following equation for \( V \):

\[ V = \frac{1}{\tau_s + 1} V_{in} \]

where \( \tau \equiv C/G_1 \) and \( s \) is the Laplace transform of \( \frac{d}{dt} \). This circuit is known as a follower-integrator (Mead 1989) and has the same transfer function as an R-C integrator.

4. The P-D Regulator

In this section we will describe an embedding of a P-D (proportional–derivative) regulator into our design frame. We will first describe the differentiator circuit used in the derivative section of the regulator. Then we will describe the implementation of the complete regulator.

The differentiator circuit

Figure 6 shows the schematic of a differentiator circuit consisting of two transconductance amplifiers and a capacitor. Kirchoff's Current Law at node \( V \) requires that the charging current supplied by the amplifier with transconductance \( G_1 \) must equal the current flowing into the capacitor, thus

\[ G_1 (V_{in} - V) = C \frac{dV}{dt} \]

overlapping pulses. The sign of the aggregated currents determines which output (\( V_{out1} \) or \( V_{out2} \)) is firing. By using the pulse outputs from the chip as inputs to the two terminals of a D.C. motor, we can bidirectionally control the motor.

The completed system

The design frame implementation is shown in Figure 5. The systems consist of \( N \) input devices implemented as complementary-output amplifiers. Each amplifier has a differential input voltage \( V_1 - V_2 \) and a gain set by \( V_0 \). The output currents from these amplifiers are summed onto a pair of wires. These aggregated currents are half-wave rectified and converted into pulses by two neuron circuits. This system converts an arbitrary number of weighted differential voltages into a pair of pulse trains. The dual-rail complementary nature of the system guarantees non-
The derivative is computed by the second transconductance amplifier which takes the difference between the output and the input voltages of the follower-integrator. Using the transfer functions for the follower-integrator and the transconductance amplifier in its linear regime gives the following transfer function for this circuit:

\[ \frac{i_{out}}{V_{in}} = \frac{G_2 \tau s}{\tau s + 1} \]

For \( \tau s \ll 1 \) the right side of this equation can be approximated by \( G_2 \tau s \) which is the transfer function of a perfect differentiator.

The completed regulator

Figure 7 shows the completed P-D regulator. The proportional section is implemented by using a complementary-output amplifier. The derivative section is implemented using the differentiator circuit with the output amplifier replaced by a complementary-output amplifier. The inputs to the system are a the operating point represented as a voltage \( y \), and a set-point reference voltage \( y_{set} \). The constant \( \tau \) sets the time constant of the differentiator. The constants \( k_P, k_D, k_R \) are used to set the gains for the proportional and derivative sections of the regulator. Given that \( \tau s \ll 1 \), the outputs of the regulator are

\[ u_+ = k_P (y - y) - \frac{k_D \tau}{k_R} \frac{dy}{dt} \]

\[ u_- = k_P (y - y) + \frac{k_D \tau}{k_R} \frac{dy}{dt} \]

5. Experimental results

In this section, we describe an experiment in which we compare the operation of our pulsed-output controller with a standard analog D.C. controller. Before going into the details it is relevant to mention something about robot control principles. Recently, nonlinear controllers utilizing the physics of robots has been proposed in general terms (Slotine, 1988), where classical concepts in mechanics like e.g. energy conservation, can be used both in stability analysis and in the design of trajectory controllers. One specific factor of physical nature is friction, which has become a popular phenomenon of study, since its handling in control is important and nontrivial (Canudas et al., 1987). In the following, this latter aspect is significant.

We use a simple robot model as our physical plant. A robot can, from a motion control point of view, be regarded as a physical setup consisting of a electric motor, a drive system, and an inertial load. In our case (Figure 8) we have a 18 V DC-motor, a gear-box with ratio 1:8, a coupling with some backlash, and an inertial load. The position and speed is measured and available on the front panel. Our experimental setup is shown in Figure 9. The neuron chip was installed in the box to the left of the D.C. servo. The voltmeter is used for coarse speed measurements.

At medium and high speeds both controllers work well. However, at very slow speeds, the pulse controller continues to drive the system while the analog controller fails. The analog controller needs a fairly high reference signal to cause the motor to break free of the static friction inherent in the system. When the reference velocity is decreased, the system has a tendency to "stick". With the pulse servo, however, each pulse contains enough energy for the motor to overcome the static friction. Thus, instead of stopping when the reference velocity is small, the motor continues to move. At high speeds the time-averaged output of the pulse controller operates as the control signal, and the duty cycle controls the speed as expected. At low velocities, however, each pulse generates
a discrete movement of the motor, similar to the operation of a stepper motor.

The behavior has not been fully explored in experiments, but a simple measurement of the speed indicates the following. The conventional controller can hardly function in speeds below 2 rad/s. At these speeds a small disturbance can stop the motion. The neuron servo on the other hand easily can work reliably at below 0.2 rad/s.

6. Conclusions

We have presented a biologically inspired controller applicable to robotics applications. We have described the basic circuitry and design frame for such an implementation and showed how they can be applied to a particular example, namely a P-D regulator. We have also shown the advantages of using a neural-like pulsed output representation to improve slow-speed operation of a friction limited system. Finally, we discussed the utility of exploiting parallelism, aggregation, and redundancy to improve system-level performance given imprecise low-level components.

One advantage of the analog VLSI technology used in this implementation is the ease of combining sensing, intermediate processing, and control (both conventional and adaptive). Sensory systems (visual and auditory) that can be used as the input sensor for feedback systems have been designed (Mead, 1989). Since the control circuitry is implemented using VLSI it can easily be integrated on the same chip as these sensory systems to form integrated sensory systems. We have designed a system combining a visual system that computes the centroid of the 2-D visual field (DeWeerth and Mead, 1988) with two P-D regulators to be used for 2-D visual tracking. We are also working on the extension of the neuromally inspired designs to the modeling of high-level biological sensory–motor systems.

We are concentrating much of this effort on oculomotor systems and have completed a simplified adaptive model of the vestibulo-ocular reflex (DeWeerth and Mead, 1990).

Speed and accuracy are major issues in the design and control of robots. However, the robot characteristics when it comes to flexibility, dexterity, and intelligence could be even more important. Therefore it is important to consider the function of and interfaces to higher levels of the design when designing the basic motion control. On this higher level adaptivity is desirable both in its classical form and more recent versions of neural networks (Åström and Wittenmark, 1989). Also here, adaptivity can be integrated on the same chip using the proposed design frame.

We see this technology as having applications in many areas. A very promising set of applications comes from biology itself. The control of artificial motor systems to mimic the behaviors of animals should be much more attainable if the low-level computational structures are also biologically related.

7. References


