Meso: A Virtual Musculature for Humanoid Motor Control

by

Bryan Adams

B.S., Electrical Engineering and Computer Science (1999)

Massachusetts Institute of Technology

Submitted to the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the Degree of
Master of Engineering In Electrical Engineering and Computer Science

at the

Massachusetts Institute of Technology

September 2000

Signature of Author ...
Department of Electrical Engineering and Computer Science
September 8, 2000

Certified by ...
Rodney A. Brooks
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by ..
Arthur C. Smith
Chairman, Department Committee on Graduate Theses
Department of Electrical Engineering and Computer Science

Meso: A Virtual Musculature for Humanoid Motor Control

by
Bryan Adams

Submitted to the Department of Electrical Engineering and Computer Science

September 8, 2000

in Partial Fulfillment of the Requirements for the Degree of
Master of Engineering In Electrical Engineering and Computer Science

ABSTRACT

Humanoid behavior requires a system with access to humanoid variables. Our humanoid robot, Cog, has two arms that are structurally similar to those of humans; however, the sensory system only provides a sense of strain and position. This thesis describes a model of the human energy metabolism that is linked to the robot s behavior. As the robot uses its arms, the model incorporates the behavior to create a sense of tiredness, fatigue, soreness, or excitement in the robot, both locally at the joints and globally as a part of the overall system. The model also can limit the robot s exertion when appropriate according to the biological system.

Thesis Supervisor: Rodney A. Brooks
Title: Professor, Electrical Engineering and Computer Science

Contents

Acknowledgements

I did this work with the help of many, many important people.

Professor Rodney Brooks, my supervisor, has provided me with the doorway to the world of Artificial Intelligence and humanoid robotics. Being a part of his group has made my MIT career what it is today. I also wish to thank the members of the Humanoid Robotics Group at MIT for their friendship and support.

In particular, Matt Marjanovic, my one-time boss and current mentor, provided me with the inspiration for this work. I hope that, as he finishes his degree, he finds what I ve done to be helpful and maybe even inspirational. I look forward to working with him over the coming year(s).

My family (Mom, Dad, Emily and Laura) has always been my main source of support and guidance. Many phone conversations and trips home have provided me with the perspective and enthusiasm that is critical to my success. I love you all.

My sister, Emily A. Adams, was particularly helpful not only in providing emotional support, but also tirelessly explaining the basics of biochemistry to me, answering pointed questions, and generally taking me from a bio-birdbrain to the point where I am today.

1 Motivation

One of the central challenges in building humanoid robotics is how to best make the electro-mechanical systems in robots emulate the biological systems in humans. Cameras are used in place of human eyes, microphones in place of eardrums, hinges and bearings in place of cartilage and bones, and electric motors or hydraulics in place of a muscle system. Of course, the systems are designed to be very similar to their human analogs, but due to the difference in material and configuration, the robotic and human systems often have radically different abilities and limitations. While overcoming the limitations of the electro-mechanical systems are part of the engineering task, some of the abilities of the robotic system may also create problems because they lead to abilities that are patently *unhuman* in their magnitude or scope.

The example dealt with in this thesis is that of the motors that replace the muscles in the arms. In humans, the duration of a muscle exertion is regulated by a number of different systems delivering energy to the muscle. However, the robot s arms, because they run on electricity and are plugged into the national power grid, have limits in duration of exertion that greatly exceed those of humans. In this case, the robot has super-human ability unless artificial limits are implemented to make the system conform to the same behavior as the human system. The *meso* system proposed by this thesis does exactly that: by simulating some of the human metabolic system, the robot will be restrained by limits that do not stem from its physical system.

One might ask what the value is in putting human-like limits on the mechanical system. Just because humans are limited in the amount of force they can exert, should the robot be subject to such limitations? Another suggestion that deserves consideration is to give the robot a mechanical metabolism that is related to the limitations of the robot s system. The answer to these questions lies in our research focus on human-robot interaction.

1.1 Humanoid robots must operate in the world through human channels in order to develop like humans

The first reason to place human-inspired limitations on the robot s behavior is to aid in its human-like development. Our lab focuses on building a robot that undergoes a development pattern inspired by human development (Brooks 1997), which implies that the control system that dictates the arm s behavior should develop along the same lines as the human system. This type of developmental control structure relies on humanoid channels to sense the world.

When working on real robots, though, this type of developmental control structure for arms requires an additional component because of the lack of a physical analog. A contrasting example will help illustrate the point. An active vision system that models the human visual system (such as the one designed by Scassellati 1998) must model the sensory channel provided by the eyes. The cameras that serve as the robot s eyes, while not perfect, provide a reasonably similar channel of information. The engineering effort necessary to rectify the differences between the biological system and the mechanical system are relatively minor: a pair of cameras has proven to be sufficient to model the different resolution on different parts of the retina. With this simple modification, the vision system receives information that is plausibly similar to the information that influences the human visual system. This is not to say that the visual system itself is simple or provides limited information about the world. In fact, the opposite is true: the cameras provide much more data than the sensors in the arms. However, the data provided by the eyes and the data provided by the cameras is sufficiently similar to create human-like, visually controlled systems.

Modeling the arm s sensory system, however, requires a great deal more engineering effort. Whereas the cameras act as a reasonable biological analog to eyes, the motors in the arms provide none of the sensory information associated with muscles. Some effort has been made to create a physical system that emulates the human musculature; specially designed joints provide strain and position information (see Chapter 3 for more details). However, the metabolic system, which both provides the energy for movement as well as perceptual feedback, has no physical analog. The robot is able to draw power continuously, without any sort of sensory change, for as long as the physical structure will support the behavior.

Yet the metabolic system in humans has a direct impact on behavior in arm movement: the ability to exert a force is constrained by the metabolic system s ability to deliver energy to power the exertion. This limitation to the arm s ability to move will have a particularly significant effect on a control system that uses the arms real behavior to learn new behaviors. Additionally, the metabolic system provides sensory feedback about the state of the energy delivery system (i.e., feeling tired after a lengthy or intense exertion). By using this data, a learning control system can anticipate the limits of the energy supply system and alter behavior before these limits are reached. By supplying both of these feedback sources (a limit on exertion and feedback about energy usage), the *meso* system will allow for the development of a more humanoid control system.

1.2 Empathy with the robot is enhanced if the robot experiences human-like limitation and sensation

Of course, controlling a humanoid robot to behave in humanoid ways is a goal in its own right. However, the *meso* system contributes to a deeper, and perhaps more important aspect of this goal. If the purpose of humanoid robotics research is to build a robot with the highest possible level of function, then sacrifices on behalf of biological conformity probably do not make sense. However, over-emphasis on the functionality of a robot can quickly strain the term humanoid . To use the system in question as an example, consider the case of a robot with unlimited motor power at its disposal. As its control system explores the abilities of its arms, the robot has no reason to ever modulate behavior to conform to human speeds or strengths, use postures that maximize efficient energy usage, or take breaks. A human interacting with this robot will see these behaviors and constantly be reminded that his counterpart is a robot.

But the robot has lost more than the aesthetic value of resemblance to a human. The interactions between it and humans will take on a completely different dynamic from human-to-human interaction. At a fundamental level, a human and a robot cannot easily share experiences unless the robot s experiences, which are linked its capabilities, are similar to those of a human.

The *meso* system, like all the systems that are implemented on Cog, reflects this value of shared experiences for two reasons. First, for humans to attribute human qualities to the robot, it must have a truly humanoid form. This means that the robot *should not be able* to perform actions that are outside the range of human ability. While such actions may provide a functional benefit (in the traditional sense of the word), they ultimately create barriers between the robot and the person, and are therefore in conflict with our research goals. As the human sees the robot perform a reasonably human arm gesture, he is able to identify what it feels like to make such a gesture, and this can provide valuable insight into the robot s motivation. If, however, the gesture exceeds human ability in magnitude or duration, then the human is unable to identify with the behavior and the connection is lost.

The second reason for valuing shared experience over function is to allow the robot to experience the world through humanoid channels. If the robot has senses that humans do not or lacks crucial senses that humans have, there will be two negative outcomes. First, the robot s reaction to the environment will be wrong. If the robot has alien sensors, it will react to things that humans will not react to, and this inhuman reaction will create barriers to social interaction. If it is missing crucial sensors, it will fail to react to typical human stimuli, and inhuman behavior will result. Second, the robot s internal state will be governed in large part by the data it receives from its sensory channels. If the robot s experience includes channels unknown to humans, then the hidden internal state will be largely indecipherable. And, because interaction relies on the na ve observer correctly inferring this internal state, missing or alien sensory channels will break down or prevent human interaction.

Of course, the mechanics of these electro-magnetic sensorimotor experiences are still fundamentally different from their biological counterparts. And, without knowing exactly what parts (if any) of our sensorimotor systems are important, we cannot say for certain that a robot will or will not have an experience sufficiently similar to the human one to allow for meaningful interaction with humans. However, even if this is the case, by simulating human-like systems on robots (perhaps at the cost of traditional function), we will learn something about human-machine interaction, and perhaps even the human systems themselves.

2 The Platform

2.1 Typical position control results in stiff, non-compliant behavior with low sensory value

Robotic arms are typically controlled via commanded joint angles. Consider a simple example of a robot arm made up of two segments connected at a rotational joint. An angle between the segments is typically commanded to the joint and a sensor provides the actual angle of the arm. A control strategy (usually based around a proportional-derivative controller) then attenuates the error in the output. Commanding a series of angles with small differences at a set rate generates simple movement. This basic control strategy can be augmented in several ways: a more complex controller can attenuate the error more quickly; inverse kinematics can create smoother trajectories; dynamic modeling can compensate for error caused by the physics of motion. But these refinements do not change the nature of the arm. It moves precisely from position to position, making accurate task repetition easy. With the right motors and mechanical setup, the arm can also be very powerful.

Applications of this type of arm take advantage of these qualities. Robot arms are often involved in spray painting automobile equipment because they can do a consistently efficient job, lowering costs for paint and raising the quality of the paint job. Other robots are employed for microchip fabrication, manipulating small objects in small spaces where human hands would be too big and clumsy. Other arms perform tasks too dangerous for humans such as ultra-high temperature welding. These types of position-controlled arms are useful, in fact, for the very reason that they are *not* like human arms. In strength, accuracy, and precision, these robotic arms vastly outperform the human arm.

But superiority in these areas comes at a price: the robotic arms are very poor at tasks that humans consider trivial. For example, obstacles to a planned trajectory represent a serious challenge: the result is often damage to the arm or the obstacle or both. Changes in the dynamics of the arm (say, picking up a mass at the end of the arm) can cause a radical change in behavior or even instability. Additionally, the feedback from a robotic arm provides very little information about its state. A typical robotic arm accurately senses position and quickly computes the derivatives (velocity, acceleration, jerk). By contrast, humans can control their arms not only using a basic sense of position and velocity, but also other senses about the extremes of motion: pain when the arm is extended too far, fatigue when the arm has been moving for too long, soreness after a particularly strenuous exertion has taken place. Just as the robotic arm s control is based around its sensory information, the human arm s control uses these variables to modify behavior.

2.2 Spring-like behavior forms the basis for a simple but effective muscular system

Our goal in putting arms on Cog is to provide the robot with another humanoid channel to its environment. For that channel to truly be humanoid, the arms must not only behave in a humanoid way, the robot should control them as humans do as well. This is an important distinction to make: it is possible to give the appearance of humanoid behavior without humanoid control. But a humanoid behavior, if it is generated by an un-human control structure, will be insufficient for our work. Because the robotic arm produces movement using an electric motor instead of muscles, humanoid movement must be system must come from the control structure commanding the motors. This also fits with the development model being used to control the rest of the rest of the robot. The arm behaviors should start out at the level of an infant and grow over time. If the control structure generating the behaviors is not based around a human model, there is no reason to expect this development to take place, and the robot s arms will not be able to be integrated with the rest of the robot. Additionally, humanoid behavior includes humanoid reaction to disturbances from the environment. If the behavior of the robot s arms is generated in a non-human way, then the reaction to these disturbances will be governed by the rules of that command law, and the result will not be humanoid behavior.

The first issue to be addressed by the control structure concerns the nature of the movement. While most robot arms are controlled by regulating the position of each joint, the physiology of human arms makes this kind of control highly unlikely if not impossible. First, human arms follow very smooth trajectories, and the command rate that this would require has not been found among the control structures in the arms (Massaquoi 1996). Second, because muscles can only exert force in one direction (the direction of contraction), human arms are moved by concerted effort on the part of antagonistic muscle pairs. Often, one joint (such as the shoulder) can be under the influence of several of these pairs. For successful position control, they would have to work in close concert, and, again, the biological evidence does not support this structure. Third, studies of human physiology (such as Stein 1982) have suggested that there is a direct relationship between the force that a muscle exerts and the length of that muscle at the time of exertion (which may or may not relate to joint position, depending on the insertion points of the muscle). This relationship between length and force by antagonistic muscles has led several researchers to conclude that muscle pairs create "springlike" motion (Hogan, 1990).

These two conclusions led to two design decisions for the control of Cog s arms. First, for the many reasons listed above, position control was discarded in favor of force control. This means that the commands issued to the motors regulate the force output without regard to position. This allows the motors to function more like muscles with minimal computational cost. From a technical standpoint, force control requires an accurate measurement of the force exerted by the motors, however the novel series-elastic actuators (see Pratt & Williamson 1995) and a simple PID controller make this step easy and precise.

The biology of human arms also inspired the first control law for Cog s arms. Emulating the action of antagonistic muscles with a correlation between length and force, each joint s control law consisted of a simple proportional-derivative spring law. The force for each joint depended on the displacement from a prescribed set point (and the rate of movement) to determine the commanded force:

$$\tau = k\left(\theta_2 - \theta_{setpoint}\right) + B(\dot{\theta}_2)$$

where tau is the commanded force at a joint, θ_2 is the actual joint angle, and $\theta_{setpoint}$ is the equilibrium point of the virtual spring. K and B represent a stiffness and damping coefficient, respectively. This spring law is effective in creating many humanoid arm behaviors. By manipulating the set point of the virtual spring at each of the joints, the arms move smoothly and deflect without harm off obstacles to the trajectory. This simple control law successfully served as the foundation for another body of work involving neural oscillators (Williamson 1999) that allowed to robot to saw wood, play with a slinky toy, and throw a ball. This simple system of springs has also allowed for multi-modal work, such as reaching for a visual target (Marjanovic, Scassellati, Williamson, 1996).

It is important to note that this simple spring law is not a panacea. It omits several subtleties about the musculature system. First, the effect of where the muscles attach to the skeleton (the complexity of insertion points) is ignored entirely. Second, the fact that muscles sometimes attach to other muscles (thereby co-activating each other in some cases) is also ignored. Third, the musculature of the human arm does not consist of one pair of muscles at each joint. Most of the muscles used for common movement span multiple joints (polyarticulation). The effect of this structure, particularly salient, given that the various joints can affect each other s states, is also ignored. Yet, given the large range of humanoid behaviors created with this simple law, it represented a good foundation for future work.

The *meso* system hopes to address another of the spring law s shortcomings. Human muscles are "springlike," but do not blindly conform to a simple spring law. The chemical reactions that create movement consume energy and create byproducts that often affect performance. These effects are not modeled in the spring law at all; the arms currently behave as if they were attached by mechanical springs. Additionally, the energy exerted by the muscles in the arms creates additional proprioceptive feedback that can have an effect on behavior, action selection, and emotive state. In order to move forward in creating generalized arm behaviors, these shortcomings must be addressed.

3 The Biological System

While the virtual spring offers a great deal in terms of creating a foundation for humanoid behavior, it falls short in a number of key areas. Specifically, the spring law does not include any limits due to energy consumption or muscle capability. In order to have the arms perform fully humanoid behaviors, they must not only be limited in the same ways, but the limiting system should provide the robot with human-like feedback about itself.

In humans, this limiting system is a part of the set of chemical processes collectively known as metabolism. Metabolism is a general term that describes, at the molecular level, the interactions that sustain life. While this thesis is primarily concerned with the biochemical reactions that govern energy synthesis, storage, and consumption, different reactions (also called pathways) drive other critical processes such as digestion and waste removal, growth and healing, and reproduction.

The many different metabolic pathways interact in a number of ways. Often times, two pathways will share a starting substance. Other processes may create by-products somewhere in the middle of their reactions that are used by other pathways (often in entirely different organs of the body). They also all share a need for energy, making this function of the metabolic system central to the body s survival (McMurray 1977). These same energy-producing pathways provide the energy for limb movement. For these two reasons, this aspect of the metabolic system will be modeled in this thesis.

This chapter will provide a brief description of how the biological system works from a control perspective.

3.1 ATP, the basic muscle energy unit, is drawn from several sources

All of the body s functions require energy. The neural impulses in the brain, the churning of stomach acid to break down food, and the beating of the heart to pump blood all require some form of chemical energy. Muscle contraction, the reaction of a group of muscle cells to a specific stimulus from the nervous system, uses chemical energy in the same way to create the limb movement relevant to this thesis. At a molecular level, the answer is quite straightforward: energy is harvested by breaking the high-energy phosphate bond in adenosine tri-phosphate (ATP), leaving adenosine di-phosphate (ADP). At the metabolic system level, though, this begs the following question: where does the ATP come from?

The answer is different for each tissue in the body, but in the case of muscles, there are four potential sources for energy. First, the muscle has a small supply of creatine-phosphate (CP) stored locally in the muscle (Okunieff 1978). The CP readily donates its phosphorus bond to ADP to form ATP. However, this local store is not enough for extended movement, and it mainly allows other energy systems time to swing into action.

As the CP is exhausted, the muscle begins to break down its next local supply of energy, glycogen. Glycogen breakdown has two prime effects: the first is to create energy for the muscle in the form of ATP, and the second is to signal the body to provide additional energy for extended movement. Under ideal conditions, glycogen produces 37 units of ATP from each glycogen unit (Okunieff 1978), and the local supply in each of the tissues in the body can provide energy for roughly 24 hours before other sources must be brought into use. In addition, the glycogen burning also releases lactic acid into the bloodstream, which starts a chemical reaction known as the **Cori cycle** (Gilbert 1992). The liver uses the lactic acid to produce glucose for use by the muscles. Glucose is a mobilized form of glycogen: the muscle can use it for energy during exertion, and it is harvested and stored as glycogen during times of rest. Interestingly, because muscles lack a key enzyme (glucose 6-phosphatase), they can only use this local store of glycogen for their own use. This is in contrast to the liver and kidneys, which can (and do) convert glycogen to glucose to be circulated in the bloodstream.

As activity continues, the muscle moves on to its third source of energy, fat (note, this requires the proper oxygen conditions, see section 3.3). Stored as adipose tissue, the fat supply in a well-nourished human provides a (virtually) limitless energy supply. It is worth noting that fat is not another storage form of glucose, but instead is broken down into fatty acids, which are released into the bloodstream. These fatty acids are then further broken down into ATP via the tri-carboxylic-acid cycle (TCA) cycle (Champe 1994). Fat breakdown, also called **beta-oxidation**, is triggered by low energy levels in the muscles. If the fat supply runs out or fat cannot be broken down because of the condition of the fuel consumption in the muscle, then the final source of energy is the breakdown of protein in the body. Cells to form structural and functional components using protein, however, and using it as a fuel source can cause damage to the overall structure of the body. Protein breakdown only occurs under situations of extreme malnourishment.

3.2 The bloodstream serves to carry both messages and fuel

The purpose of the energy metabolic pathways is to provide the body with both a constant supply of fuel for general operation as well as providing for the sudden spikes of need that accompany physical exertion. This is accomplished through the alternate storage and mobilization of the different fuel sources in different locations. Because the system is distributed, though, there must be communication and transport to allow each of these anatomically separate organs interact in efficient ways. The bloodstream carries both the messages that coordinate fuel usage as well as the actual fuel itself (Fell 1997).

The pancreas is a critical organ in coordinating the organs involved in the energy pathways. It monitors the content of the bloodstream and sends out chemical messengers that activate the storage and release mechanisms. **Insulin** is secreted when the glucose level in the bloodstream is high. This is a signal to the muscles, liver, and kidneys that there is sugar available in the bloodstream (which was provided by nutrients from the gastro-intestinal system) and it should be stored for later use. This causes the adipose tissue to create fat, the liver and kidneys to store glucose in the form of glycogen to later release into the bloodstream, and the muscle to store glucose as glycogen for later use during spikes of activity. People afflicted with diabetes have a pancreas that does not secrete enough insulin, and, consequently, can suffer the effects of malnourishment because their organs do not store enough energy during feeding times.

Glucagon creates the opposite behaviors from insulin. Instead of signaling that the bloodstream is full of glucose, glucagon is a signal that reserve fuels should be mobilized and deposited in the bloodstream for use. The liver and kidneys respond by increasing their breakdown of glycogen into glucose, and adipose tissue increases the deposit of fatty acids into the bloodstream. Muscles often begin the process of protein phosphorylation, and this can cause damage. Glucagon generally signals that the energy pathways are low in precursors, and the body is in desperate need of energy.

The final signal carrier is **epinephrine**. Created in the adrenal medulla and the sympathetic nerve endings, it signals a sudden spike in the need for energy. It works very similarly to glucagon in that it stimulates the body to release stored energy for use. Primarily activated in dangerous or frightening situations, though, it works faster and provides a more immediate source of energy.

3.3 The heart rate has an impact on the quality and type of fuel consumption

In addition to carrying the fuel and messaging systems, the blood also transports other chemicals from tissue to tissue for use in other chemical reactions. While most substances are generally kept within homeostatic boundaries, one that has a wider range of values during times of extreme exertion is oxygen. While the amount of oxygen that the blood can carry is relatively static, the amount that is delivered to the organs is dictated by the heart rate. As exertion goes up, signals through the nervous system bring the heart rate up as well to increase the flow of blood through the system.

The variation in heart rate, though, also impacts the way the fuel is burnt. The bloodstream delivers oxygen, a critical component in each of the chemical reactions that create ATP. However, the rate at which the blood is pumping either enhances or prohibits the amount of oxygen that can be delivered for these reactions. If the blood is moving too slowly, an insufficient volume of oxygen is brought to the muscle. If the blood is moving too quickly, though, the reactions are unable to use the oxygen that is present. This means that the blood must be moving at a mid-range speed to maximize oxygen utilization. Fitness experts term this speed as a target heart rate (Sizer 2000).

The heart rate varies greatly over the course of normal activity, though. If the heart rate is in the target range, there are positive effects on both the body and on behavior. With an abundance of oxygen, the muscle can use energy-rich fat as a fuel source. One fully-combusted unit of fat is the best energy source available in the body, and it is virtually inexhaustible. Those trying to lose weight are generally trying to burn fat, therefore, exercise specialists often recommend exercising at a level of exertion that keeps the heart pumping the target heart rate to maximize the fat that is burnt.

When oxygen is not available, though, glycogen or glucose must be burnt to sustain activity. Burning this fuel anaerobically, though, has three negative side effects. First, the amount of ATP that is produces per unit of glycogen/glucose is much less. The complete aerobic combustion glycogen can produce ten times the amount of energy that anaerobic combustion yields. Consequently, more fuel is used for the same amount of work. Additionally, the incomplete combustion of glycogen/glucose leaves lactic acid. While lactose is a normal by-product of glycogen/glucose use, lactic acid is not as easily taken away by the bloodstream, and can build up in the muscle, causing pain upon contraction. This feeling of soreness can be an impediment to movement. In fact, according to some research (Chase 1988), high quantities of lactic acid can cause an unbalance in the ph of the muscle, which can depress the muscle s ability to exert any force at all.

4 *Meso* Implementation

4.1 The implementation must operate on a real robotic system

Any implementation of the metabolic system must deal with the issue of complexity. While the metabolic system is well understood at the chemical-reaction level, the interplay between each of the reactions, if modeled explicitly, would create a system that would require massive amounts of computation from even the most powerful of computers. For our application, the requirements are even more stringent: the model must operate on a robot in real-time, hence the computation available must allow the model to create the proper feedback on the proper timescale.

However, creating a model with a reasonable level of complexity is reasonable for a humanoid robot because the model (at this point) only needs to recreate two aspects of the metabolism. First, it should provide the robot with both humanoid behavioral limits. This model should place humanoid restrictions on the nearly unlimited power that the robot is capable of exerting. Second, the model should create the accompanying humanoid experience that goes along with testing these limits. Without entering into the debate of whether a robot actually has experiences , it is enough to say that this system should provide an interface to metabolic variables to regulate other behaviors. *Meso* accomplishes these goals by creating a modeling some of the biochemical systems that provide energy to humans. The model does not work at the molecular level, but instead treats the metabolism as a simple control mechanism. While the values of the variables in the model do not reflect actual values found in humans (these vary too greatly for specific values to be useful), they do change in proportion with the human reactions. Because the model is fundamentally tied to the behavior of a real robot (specifically, the humanoid robot described in section 2), the result of the biochemical processes are then applied to the electro-mechanical system to achieve the second goal.

Because this implementation is specific to this robot and this environment, it is also important to consider the time scale at which the robot currently operates. In the current environment, interactions with Cog are short (i.e., less than one hour). While a small set of individuals do interact with the robot over a period of weeks, months, and years, the robot has thus far been designed to exhibit the same properties from day to day and week to week. As such, this implementation of *meso* focuses on modeling metabolic effects that happen over the course of an interaction lasting less than one hour. Because developing these longer-term effects could prove beneficial in the future, some consideration is paid to allowing for future development in this direction. Many of the long-term metabolic effects can be represented in this model by dynamically (but slowly) changing the coefficients of the reactions established in *meso*. Other long-term metabolic changes could include the results of a trauma: long periods without nutrients or with an unbalanced diet. Because the overall fitness level rarely (if ever) changes in perceptible ways over these short-term interactions, though, neglecting this part of the model will not change the nature of the short-term behaviors.

Aside from managing the inherent complexity of recreating the metabolism at the chemical reaction level, any model of the metabolic system must recognize that each person s metabolic system is entirely different. While the chemical reactions are the same in all people, the higher level relationships (i.e., the amount that the heart rate goes up for a given amount of work) vary greatly not only from person to person, but vary for a given person over the course of his life. Factors such as genetic makeup, environmental quality, and general fitness level change the relationships greatly, in some cases by an order of magnitude. As such, there is no single right behavior, but instead a range of values that the system should be able to emulate. The goal should be to have a system where the right set of parameters and associations are established, and the specific relations can be manipulated. With that basic framework, future work can model the influence of these other more distant factors.

Finally, the *meso* system creates a virtual metabolic state for the robot. But, like the metabolic state in humans, the sensing of that state is vague and poorly understood. Unlike the senses used to understand the external environment, proprioceptive sensing does not result in concrete thoughts, but instead creates a feeling that may or may not be acted upon by a higher level of control. While the *meso* system provides parameters that correspond to nebulous feelings such as tiredness , the correct use of this variable to create humanoid behavior is left to the designer of the next system.

4.2 The implementation uses the sok *architecture*

The metabolism of the robot must integrate seamlessly with the overall control structure. On Cog, this is accomplished by running all control via the *sok* architecture, created by Matthew Marjanovic (Marjanovic 2000). It allows for processes to be dynamically started, stopped, restarted, and connected with other processes. This system is currently used to implement the spring law that creates the virtual muscles. It will also serve as the basis for the metabolic system.

The lowest level of control on the robot is the servo loop providing force control at each of the joints. This is done via a DSP on the motor control card. Running at 1.6kHz, this computation is unaffected by the load on other parts of the control system, and thus provides a stable base. Commands are sent to the DSP via a library of simple calls; in the case of Cog, they are called from the body of a *sok* process. In the pre-*meso* implementation, one *sok* process collected all the sensory input, computed the spring law for each joint, and sent the commands to the DSP. This was done successfully at roughly 1kHz.

Because the *meso* system creates additional computational load, the control is split into two parts. The first part is very similar to the old implementation: it collects sensory data and generates commands for the spring law. However, instead of sending these commands directly to the DSP, they are now sent to the second part, which monitors and modulates the commands according to the metabolic model, and then sends them to the motors. Given enough computational power, this structure allows the *meso* system to have its own scheduled processing time, allowing the performance of the spring law to continue at 1kHz. Because the *meso* system is also a *sok* process, the proprioceptive outputs are available to other processes via *sok* processes.

4.3 The model consists of models of the different organs passing signals through shared memory

The implementation of the metabolism occurs in the body of the sok process. Running at roughly 1kHz, the model operates by updating a set of variables according to a list of commands. These commands, broken down by organ function, alter a set of variables corresponding to the chemicals necessary for energy metabolism. These variables represent the bloodstream through which each of the organs not only communicate (via chemical messengers) but deliver fuel and waste products. Each organ only reads the variables that the biological organ has access to, and the modification is done entirely through this shared memory.

The following sections include small stubs of code from this loop and an explanation of what the model is recreating, a short summary of how it works and how it integrates with the rest of the system, and any significant differences from the biological system. This implementation incorporates exertion from the overall robot, but only provides proprioceptive data for a single joint. The final implementation will provide this data for each joint in both of the arms.

```
/* BLOODSTREAM:     */

gstrain = 0;
for(i=4; i<6; i++) {
  // currently only computing the right arm values
  sstrain[i] = strain_short_sum_r[i]/MAX_SHORT_HIST;
  lstrain[i] = strain_long_sum_r[i]/MAX_LONG_HIST;

  gstrain += strain_long_sum_r[i];
}

gstrain = gstrain/(2*MAX_LONG_HIST);

if(gstrain < Min_Target_Rate)
  percent_anaerobic =
        fabs((Min_Target_Rate - gstrain) / Target_Dist);
if((gstrain > Min_Target_Rate) && (gstrain < Max_Target_Rate))
  percent_anaerobic = 0;
if(gstrain > Max_Target_Rate)
  percent_anaerobic =
        fabs((gstrain - Max_Target_Rate) / Target_Dist);
```

The rate of the blood flow is an important factor to the performance of the metabolism, as it transports the various chemicals from organ to organ. This rate is stored in the gstrain variable, which is the average of the average strains of each of the muscles. Essentially, it is a measure of the overall activity level of the robot. In this implementation, only two joints are activated, thus, the total strain is only divided by a factor of 2.

The percent_anaerobic variable contains the measure of how well the muscles can harvest oxygen from the blood. As described in chapter 3, if this rate is too slow or too fast, the muscles cannot get the requisite oxygen for energy-producing chemical breakdowns, and the consumption levels are significantly changed. Because this oxygen delivery relation arises out of aspects of the chemical reactions that are not modeled in this implementation, it is modeled here directly as a function of the heart rate. The Min_Target _Rate , Max_Target_Rate , and Target_Dist variables create a range of exertions that allow for completely aerobic combustion. These variables are health-related in humans and can be changed to vary the fitness level of the robot.

```
/* PANCREAS and ADRENAL GLANDS:      */

if(gi_glucose > 0)
  modify_blood_messengers(INSULIN, (gi_glucose * .00005));
if(gi_glucose < 0.2)
  modify_blood_messengers(GLUCAGON, 0.00002);

if(excitement > 0)
  modify_blood_messengers(EPINEPHRINE, (excitement * .1));
if(sstrain[JOINT] > THRESH)
  modify_blood_messengers(EPINEPHRINE,
                          (sstrain[JOINT] * .00000001));
```

The pancreas and adrenal glands provide other inputs to the metabolic system. The pancreas monitors the gastro-intestinal input to the system (provided by an input *sok* port). When there is gastro-intestinal fuel available (gi_glucose), the pancreas increments the insulin shared variable (in proportion to the amount of gi_glucose) to signal that the blood glucose will be rising, and the other organs should prepare to store that energy. When, instead, there is low gastro-intestinal input, the pancreas increments the glucagon variable, which has the opposite effect: the other organs mobilize their stored energy. In this way, the pancreas keeps the glucose shared variable at a homeostatic level. Without this regulation, gastro-intestinal input would cause a large spike in the blood glucose and long periods without gastro-intestinal input would allow that variable to become too low.

The adrenal glands produce epinephrine, which creates a very similar effect as glucagons, but has two significant differences. First, epinephrine can be introduced into the bloodstream quickly, which allows more energy to be mobilized faster in times of extreme need. Second, the adrenal glands are triggered not by gastro-intestinal levels, but instead by stress. Consequently, they can be triggered neurologically (in other words, by an emotional or behavioral subsystem), and therefore can provide extra fuel for a fight-or-flight response or they can be triggered by activity by the body to allow for extra fuel during times of extreme exertion. Both types of triggers are implemented.

```
    /* ADIPOSE TISSUE:     */

    // -- INSULIN: turn glucose into fat
    modify_blood_fuel(GLUCOSE, (blood_messengers[INSULIN] * -1 *
FAT_MESSGR_CONST));
    modify_blood_messengers(INSULIN, MESSGR_ABSORB);
    ;// (infinite supply of fat ...)

    // -- GLUCAGON: put in fatty acids into blood
    modify_blood_fuel(FATTY_ACIDS, (blood_messengers[GLUCAGON] *
FAT_MESSGR_CONST));
    modify_blood_messengers(GLUCAGON, MESSGR_ABSORB);
    ;// (infinite supply of fat ...)

    // -- EPINEPHRINE: put in fatty acids into blood
```

```
    modify_blood_fuel(FATTY_ACIDS, (blood_messengers[EPINEPHRINE] *
FAT_MESSGR_CONST));
    modify_blood_messengers(EPINEPHRINE, MESSGR_ABSORB);
```

The adipose tissue, more commonly called fat, serves as a storage facility for energy. This virtual organ reads the levels of the various chemical messengers (modified in the previous section) and modifies the relevant blood fuel levels appropriately. When the insulin variable has been signaled, glucose is drawn from the bloodstream. In the biological system, fat is created. However, because the fat source in well-nourished humans can last for days or weeks, it is assumed to be infinite in this model. Consequently, the glucose withdrawal does not actually affect the system. Similarly, when the glucagon variable has been modified, the fatty_acids variable is incremented, but no source is correspondingly decremented. Epinephrine is the final messenger, and its effects are similar to that of glucagon. The only real difference is that the epinephrine variable can reach much higher levels more quickly, and can therefore allow more fuel to be released quicker.

These three reactions highlight another important aspect of the biological system that is modeled here: glucose is turned into fat, but fat is not turned back into glucose for energy. Fatty acids are used directly by the muscles (in conjunction with glucose) to provide fuel under aerobic conditions.

```
    /* LIVER:      */


    // -- turn gastro-intestinal glucose into bloodstream
    tmpfloat = .00015 * gi_glucose;
    gi_glucose -= tmpfloat;
    modify_blood_fuel(GLUCOSE, tmpfloat);

    // -- GLUCOSE: regulate the glucose levels in the lood
```

```
if(blood_fuel[GLUCOSE] < Min_Glucose_Level) { //min=2
  modify_blood_fuel(GLUCOSE, .0002);
  ; // glycogen store for the liver ...
}

// -- LACTATE: remove lactose, put in glucose
if(blood_fuel[LACTATE] > 0)  {
  tmpfloat = (blood_fuel[LACTATE] * .0002);
  modify_blood_fuel(GLUCOSE, tmpfloat);
  modify_blood_fuel(LACTATE, (-1 * tmpfloat));
}

// -- INSULIN: store blood glucose as glycogen
if(blood_messengers[INSULIN] > 0) {
  tmpfloat = .001 * blood_messengers[INSULIN];
  modify_blood_fuel(GLUCOSE, (tmpfloat * -1));
  modify_blood_messengers(INSULIN, MESSGR_ABSORB);
}

// -- GLUCAGON: turn glycogen into glucose
if(blood_messengers[GLUCAGON] > 0) {
  tmpfloat = .001 * blood_messengers[GLUCAGON];
  modify_blood_fuel(GLUCOSE, tmpfloat);
  modify_blood_messengers(GLUCAGON, MESSGR_ABSORB);
}

// -- EPINEPHRINE: make glucose for blood
if(blood_messengers[EPINEPHRINE] > 0) {
  tmpfloat = .001 * blood_messengers[EPINEPHRINE];
  modify_blood_fuel(GLUCOSE, tmpfloat);
  modify_blood_messengers(EPINEPHRINE, MESSGR_ABSORB);
}
```

Biologically, the liver works as a chemical filter for the bloodstream. Each of the chemical messengers causes the liver to take action that maintains the levels of the various blood fuels. In the model, the liver reacts to each of the chemical messengers and blood fuels. When insulin is present, the liver reduces the glucose store and contributes to a local glycogen store. Like in the adipose tissue, this fuel store should last longer than any anticipated interactions, so no record is kept. Conversely, glucagon causes the glucose to be incremented (and a theoretical glycogen store to be depleted). Epinephrine again causes the mobilization of energy, so the glucose variable is incremented.

The liver, though, also reacts to the levels of the blood fuels. If the glucose level gets too low (from over-exertion, malnourishment, or some other metabolic imbalance), the liver notices this and turns additional glycogen into glucose. The liver is also sensitive to lactate, the waste product of aerobic combustion. The liver decrements the lactate variable and increments the glucose variable (modeling the Cori cycle in humans). This has the effect of helping maintain the glucose level even during times of exertion.

```
/* MUSCLES:     */

// -- RESTING
if(sstrain[JOINT] < THRESH) {
  // -- restore the cp if sstrain < thresh
  if(cp[JOINT] < 10)
  cp[JOINT] += .0002; // replenish in 10 sec

}

// -- INITIAL EXERTION
if(sstrain[JOINT] > (lstrain[JOINT]+THRESH)) {

  // -- drain cp, if no cp, drain glycogen
  if(cp[JOINT] > 0)
  cp[JOINT] -= sstrain[JOINT] * .00002; //reduce in 5 sec
  else {
  tmpfloat = percent_anaerobic * sstrain[JOINT] * AEROBIC_CONST;
  muscle_glycogen[JOINT] -= tmpfloat;
  lactic_acid[JOINT] += tmpfloat;

  tmpfloat = (1-percent_anaerobic) *
              sstrain[JOINT] * ANAEROBIC_CONST;
  muscle_glycogen[JOINT] -= tmpfloat;
  modify_blood_fuel(LACTATE, tmpfloat);
  }
}

// -- EXTENDED EXERTION
if((sstrain[JOINT] > THRESH) &&
   (sstrain[JOINT] < (lstrain[JOINT] + THRESH)) &&
   (sstrain[JOINT] > (lstrain[JOINT] - THRESH))) {

  // -- anaerobic combustion
  tmpfloat = percent_anaerobic * sstrain[JOINT] * AEROBIC_CONST;
  modify_blood_fuel(GLUCOSE, (-1 * tmpfloat));
  lactic_acid[JOINT] += .1*tmpfloat;

  // -- aerobic combustion
  tmpfloat = (1-percent_anaerobic) *
```

```
                sstrain[JOINT] * ANAEROBIC_CONST *.1;
    modify_blood_fuel(FATTY_ACIDS, (-1 * tmpfloat));
    modify_blood_fuel(GLUCOSE, (-1 * tmpfloat));
    modify_blood_fuel(LACTATE, tmpfloat);

  }
```

The muscle reaction to the metabolic system is the centerpiece of the *meso* system. This first section, the energy-burning model, has three sections. First, when the muscles are at rest, the initial energy source replenishes itself. The phospho-creatine well (cp[JOINT]), which is the first energy to be expended during exertion, is slowly incremented. In the biological system, this fuel source is created and broken down using chemicals that stay in the muscle. Because this reaction mainly involves chemicals that do not impact the rest of the metabolic system, the cp[JOINT] variable is replenished without drawing energy from any other source.

When the muscle is not at rest, though, there are two stages of energy expenditure. The stages are differentiated by whether the muscle has just begun to exert itself and is mobilizing the various energy sources or if the muscle is in the middle of a lengthy exertion. This is sensed by comparing the short-term average (average over the last second) with the long-term average (average over the last 30 seconds). When the short term average is much higher than the long term average, the model simulates the initial stages of energy consumption. First, the phospho-creatine well is depleted (with no penalty to the rest of the system). However this only takes a matter of seconds, and the muscle must then move on to the local glycogen store.

Burning glycogen (and the similar reaction which burns glucose) requires that the fuel be processed either with or without oxygen (aerobically or anaerobically). Because the oxygen level varies with the heart rage, the reaction generally happens under both conditions, with the available oxygen resulting in aerobic conditions, and the remainder of the energy requirement coming from anaerobic combustion. To reflect this in the model, the fuel consumption is calculated under both conditions, with the percent_anaerobic variable determining the relative quantities. The model reflects the differences between these two types of combustion in two ways. First, the aerobic portion produces lactate, which the liver then recycles into new glucose. The anaerobic portion, however, creates lactic acid, which leads to soreness and pain during muscle movement. Second, the relative efficiencies of the fuel combustion conditions are also reflected in the model. The AEROBIC_CONSTANT is ten times larger than then ANAEROBIC_CONSTANT, modeling the fact that the incomplete breakdown results in a much less energy and a much higher fuel requirement for the same exertion.

The extended exertion portion is much like the initial post-phopho-creatine phase, however it incorporates other fuel sources. The same constants reflecting the relative efficiency levels are used in the aerobic and anaerobic reaction. As exertion moves on, the muscle switches from using the local glycogen store to the glucose in the blood (the glucose variable), and because the lactate from the aerobic glycogen burning has caused the glucose variable to be incremented, its value is higher than usual. As the reaction becomes more aerobic, the muscle begins to use its most efficient fuel source: fatty acids. Released by the adipose tissue, the fatty acids allow the muscle to exert much more force than the glucose or glycogen. This is reflected in the model by the fact that the fatty acids are only consumed at one tenth the rate of the glucose (reflected in the .1 factor). It is important to note, though, that glucose is required in the burning of the fatty acid, so there is still a draw on the glucose source.

```
// -- INSULIN: replenish glycogen
if(blood_messengers[INSULIN] > 0) {
  tmpfloat = .001 * blood_messengers[INSULIN];
  muscle_glycogen[JOINT] += tmpfloat;
  modify_blood_fuel(GLUCOSE, (-1*tmpfloat));
  modify_blood_messengers(INSULIN, MESSGR_ABSORB);
}

// -- drain lactic acid
if(lactic_acid[JOINT] > 0)
  lactic_acid[JOINT] -= .000001;

tmpfloat = 1;
// -- determine if ph-balance should limit muscle output
if(lactic_acid[JOINT] > LACTIC_THRESH) {
  tmpfloat = 1- (2*(lactic_acid[JOINT] - LACTIC_THRESH));
  strain_reduce[JOINT] = tmpfloat * strain_r[JOINT];
}
```

The muscles, like other organs, store energy during times of high glucose availability. Consequently, insulin causes the muscles to draw from the glucose in the blood and add to the local glycogen stores. Because these glycogen stores can reflect a feeling of muscle fatigue, this store is modeled with a variable (muscle_glycogen[JOINT]) for each muscle, each of which is incremented during this energy storage period. Additionally, the bloodstream draws away any lactic acid that might be in the muscles. This reduces the soreness and pain that comes with anaerobic activity.

The final piece of the muscle model is the potential to limit the muscle output due to a ph imbalance. When an over-abundance of lactic acid builds up in the muscles, pain is not the only outcome. The over-acidity can cause the contractile ability of the cells to be greatly diminished. This is modeled by reducing the output strain (strain_reduce[JOINT]) if the lactic acid is over a certain threshold.

```
/* ENERGY DRAIN     */
```

```
modify_blood_fuel(GLUCOSE, -0.00001);
modify_blood_fuel(FATTY_ACIDS, -0.00001);
```

The organs each require energy to perform their various functions as well. Two small fuel drains model this effect. Most organs (muscle, liver, kidneys) use fatty acids as their resting fuel, but two critical organs, the brain and the red blood cells, lack the necessary mitochondria in their cells, and therefore must use glucose as fuel sources. To model this fact, both fuel sources are slowly drawn from over time. Monitoring mechanisms in the liver, though, ensure that these fuel sources maintain reasonable values.

5 Results and Discussion

One of the central challenges to the model is for it to exhibit the proper behavior under many different conditions of exertion. The same system must maintain homeostatic values in each of the different variables when the arm is performing a short high-intensity exertion as well as when it is subjected to a lower-intensity but longer-term exertion. This must also be combined with a proper response to the new inputs that are available to the system, namely, a gastro-intestinal input as well as the potential for a neurological stimulation of the adrenal glands.

This section will outline a series of tests performed on the system and provide the relevant outputs. Each set of results will be followed by analysis of the similarities and differences with the biological system.

5.1 Heart rate and oxygen delivery

The heart rate is regulated according to the global exertion rate of the robot. The heart rate then determines the amount of oxygen that can be delivered to the muscles during fuel combustion. In these examples, the exertion was brought to a level that was above or below the target zone, and the percent_anaerobic variable was observed and recorded.

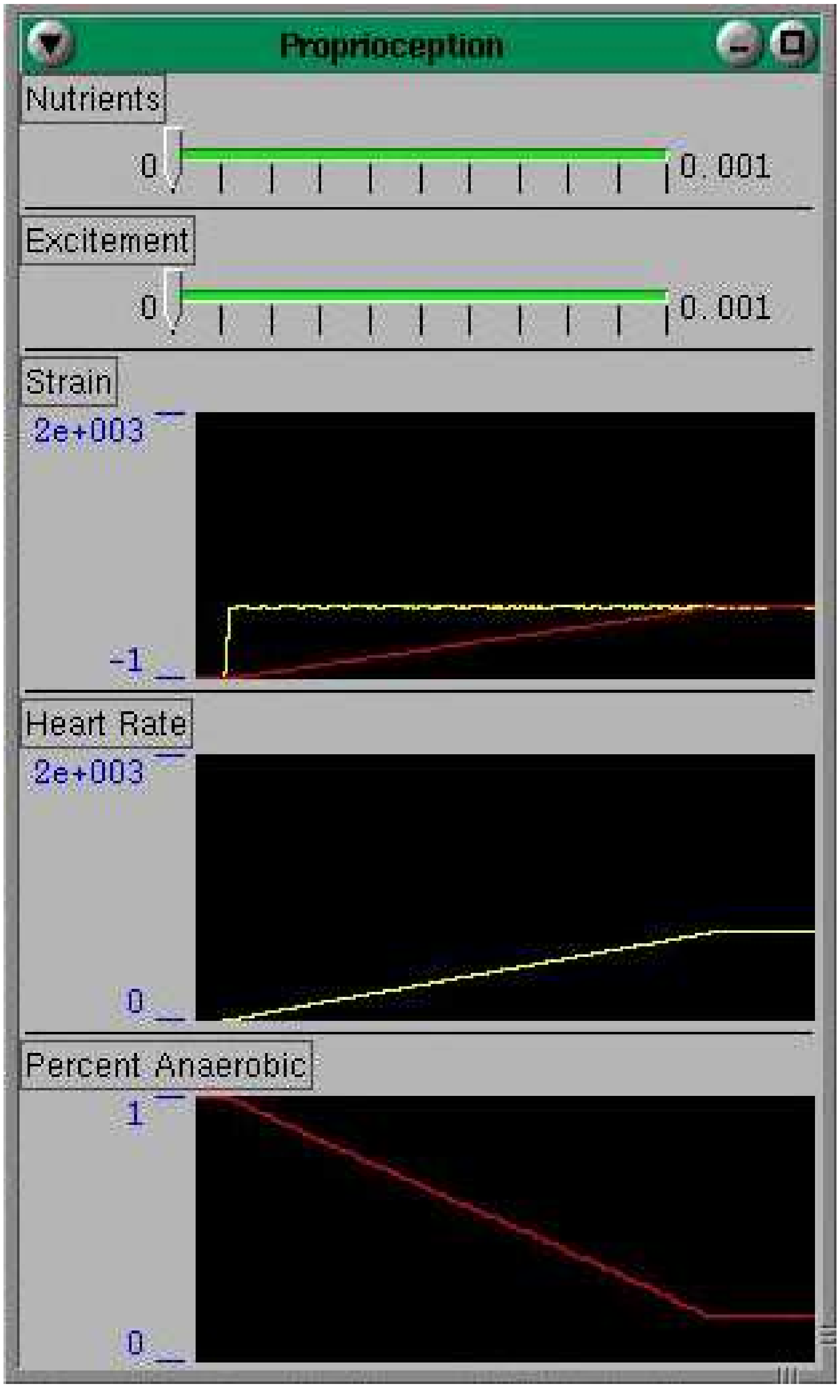

Figure 1 (time displayed: 1m 25s): The heart_rate and percent_anaerobic variables during mostly aerobic exertion. The top panel shows the short-term strain (yellow trace) and the longer-term averaged strain (red trace).

In figure 1, the level of strain is relatively low, and the percent_anaerobic variable, which is the measure of how much fuel combustion can take place in the presence of oxygen, decreases with it.

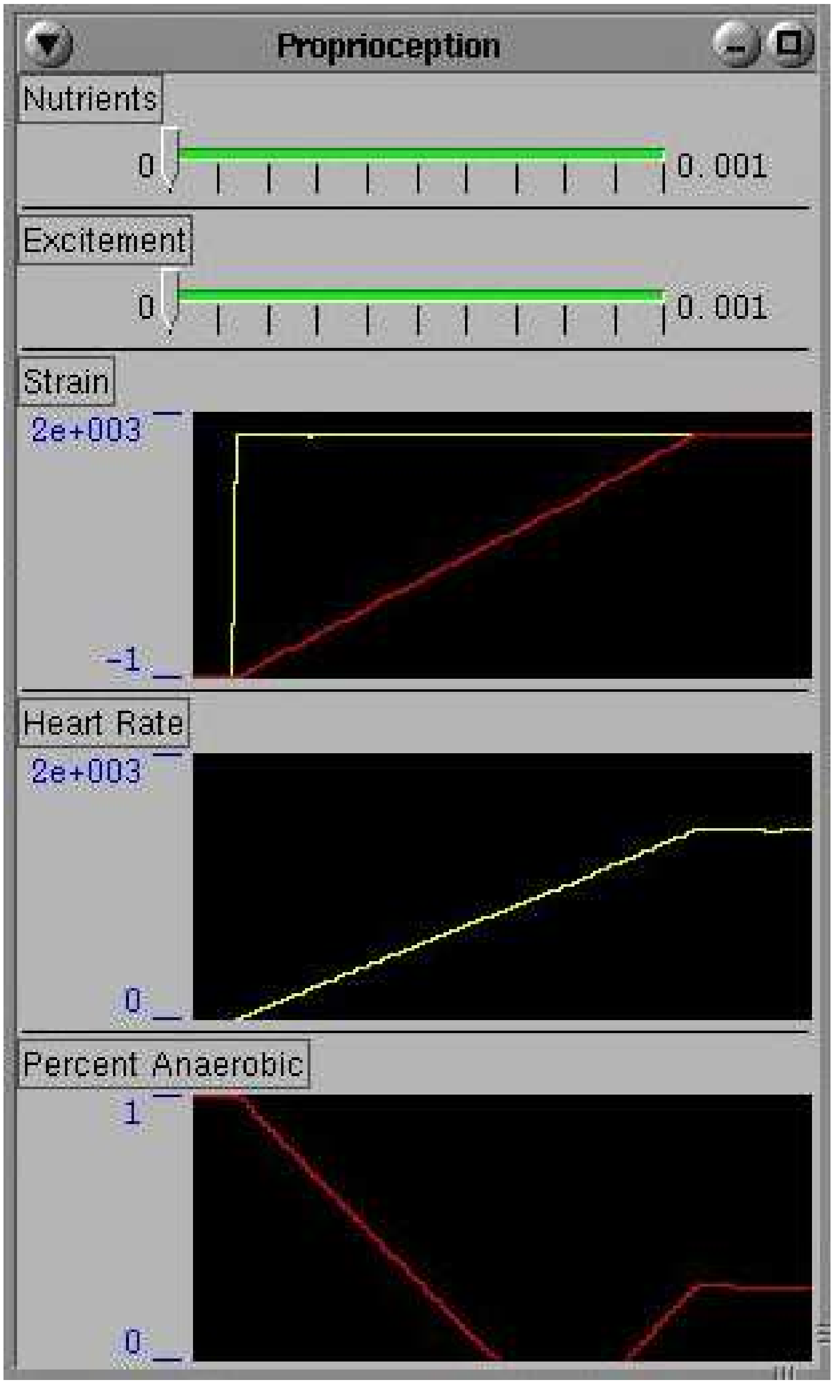

Figure 2 (time displayed: 1m 25s): The heart_rate and percent_anaerobic variables during mostly anaerobic exertion

In figure 2, the exertion level brings the heart rate out of the target zone. In this case, as the heart rate first begins to increase, the percent_anaerobic variable drops to zero and stays there while the heart rate is in the target zone. However, as it continues to rise, the percent_anaerobic goes up again, simulating the inability of the muscles to harvest oxygen due to the speed of the bloodstream.

5.2 Homeostatic fuel levels through varied nutrient inputs

As the robot takes on nutrients, it is important to keep the blood fuel levels relatively static. While they do vary over time (and the body senses those variations), they do not reach zero or their maximum values except in examples of trauma or malfunction. In the following examples, the robot s nutrient input is varied and the resulting blood fuel levels are displayed.

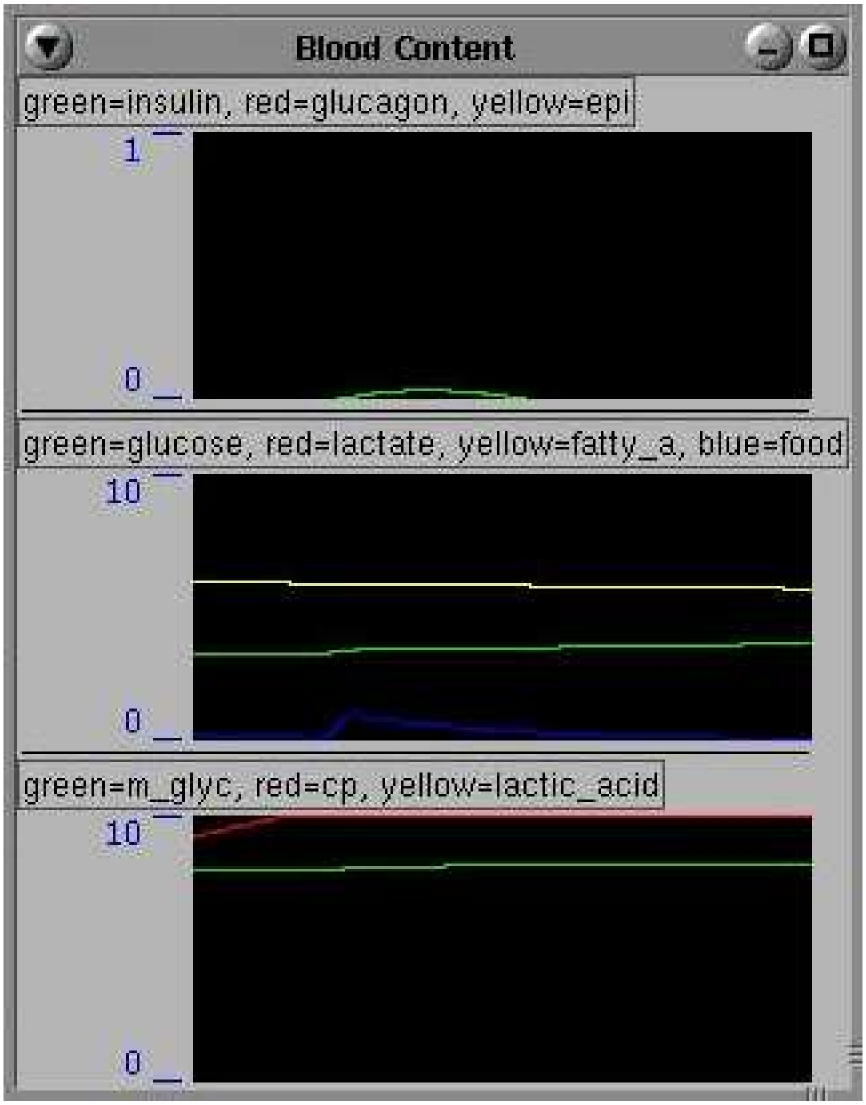

Figure 3 (time displayed: 2m 35s): Nutrients are inserted into the system via the gi_glucose variable (middle window, blue trace). The other blood fuels react accordingly.

In figure 3, the robot was given a blast of nutrients (the blue trace on the middle window). The system reacted and maintained a relatively steady glucose level (green trace, middle window) by adding insulin to the system (green trace, top window). The rise in insulin caused the other organs, such as the liver and adipose tissue, to draw from the glucose and create stores (glycogen and fat, respectively). Because the fat and liver glycogen stores are modeled as infinite stores, their levels are not displayed, however the muscle glycogen (green trace, bottom window) did rise as the insulin was introduced into the system.

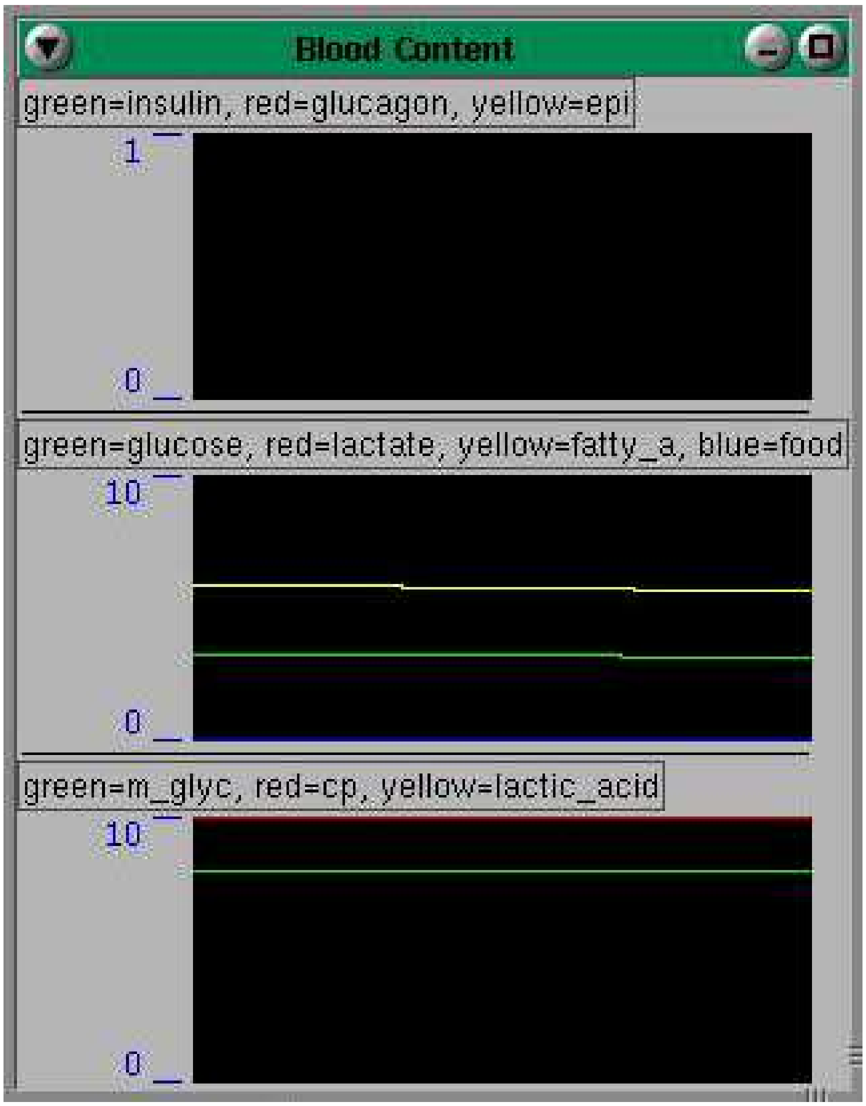

Figure 4 (time displayed: 2m 35s): No nutrients are introduced into the system, yet the glucagon (too small to show) allowed the blood fuel levels to remain relatively constant.

Figure 4 shows the robot at a state of zero input nutrients. The fuel levels do change (the body s tissues draw on the fuel for survival), but the fuel is depleted much more slowly than the rate it is being drawn. In this case, the glucagon variable is incremented just enough to encourage the liver and adipose tissue to deposit fuel into the blood to prevent the exertion from completely draining the system.

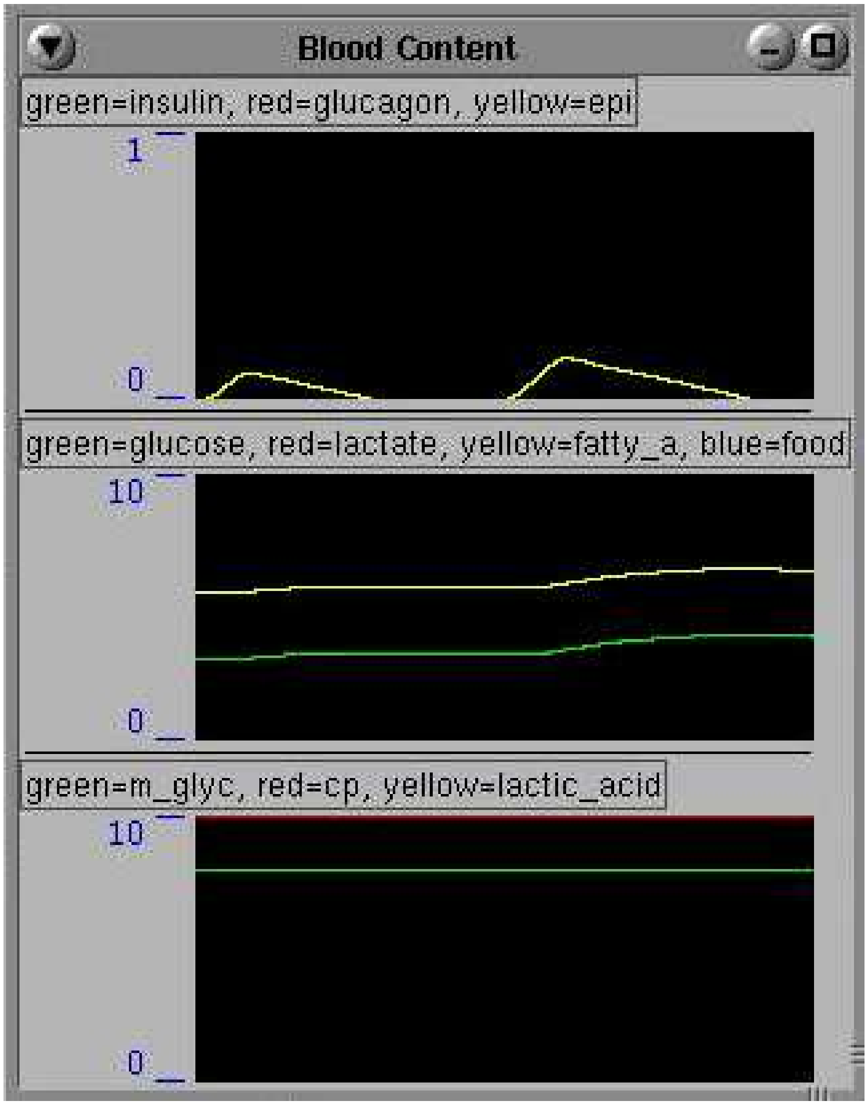

Figure 5 (time displayed: 1m 25s): Two short bursts of epinephrine cause the liver and adipose tissue to release glucose and fatty acids, respectively, into the bloodstream for immediate use.

In addition to nutrients, stress-induced epinephrine can also cause a variation in the blood fuel levels as well. In this example, the robot has just digested a small amount of nutrients, so the glucagon levels have not yet begun to rise. A small rise in epinephrine is administered to the system; similar to the way a cognitive stressor might excite the adrenal glands. The fuel levels rise, providing the body with extra fuel to prepare for action.

5.3 Blood fuel levels during aerobic and anaerobic activity

The model must not only maintain the blood fuel levels during varied nutrient levels, but also during varied levels of exertion. In the following examples, the robot will be driven at different levels of exertion and the blood fuel levels will be demonstrated.

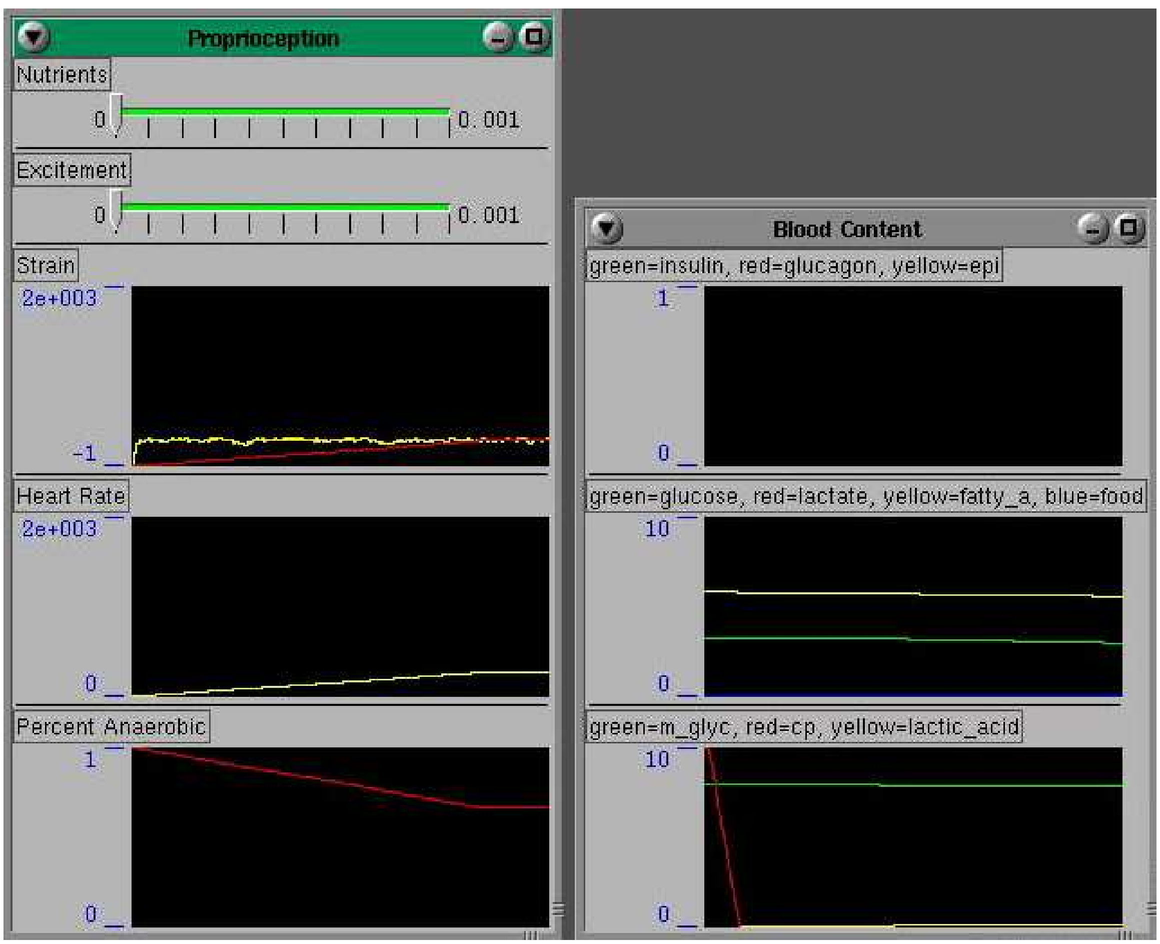

Figure 6 (time displayed: 1m 25s): With a low exertion level (yellow trace, top panel, left window), the heart rate stayed relatively low. This caused some anaerobic combustion, however the levels of lactic acid (yellow trace, bottom panel, right window) and fuel drain (yellow and green traces, middle panel, right window) are acceptable due to the low amount of fuel needed.

In this example, the robot is being driven at a level of exertion that is well below the target heart range. Although the exertion is roughly half anaerobic, because the total amount of fuel is small, the fuel levels experience little change, and there is little to no buildup of lactic acid (yellow trace, bottom panel, right window) in the system. First the phospho-creatine (red trace, bottom panel, right window) is completely depleted and then the muscle glycogen and blood glucose contribute fuel. At this level of exertion, their levels will be maintained by glucagon, a very small amount of epinephrine, and the glucose monitoring function in the liver.

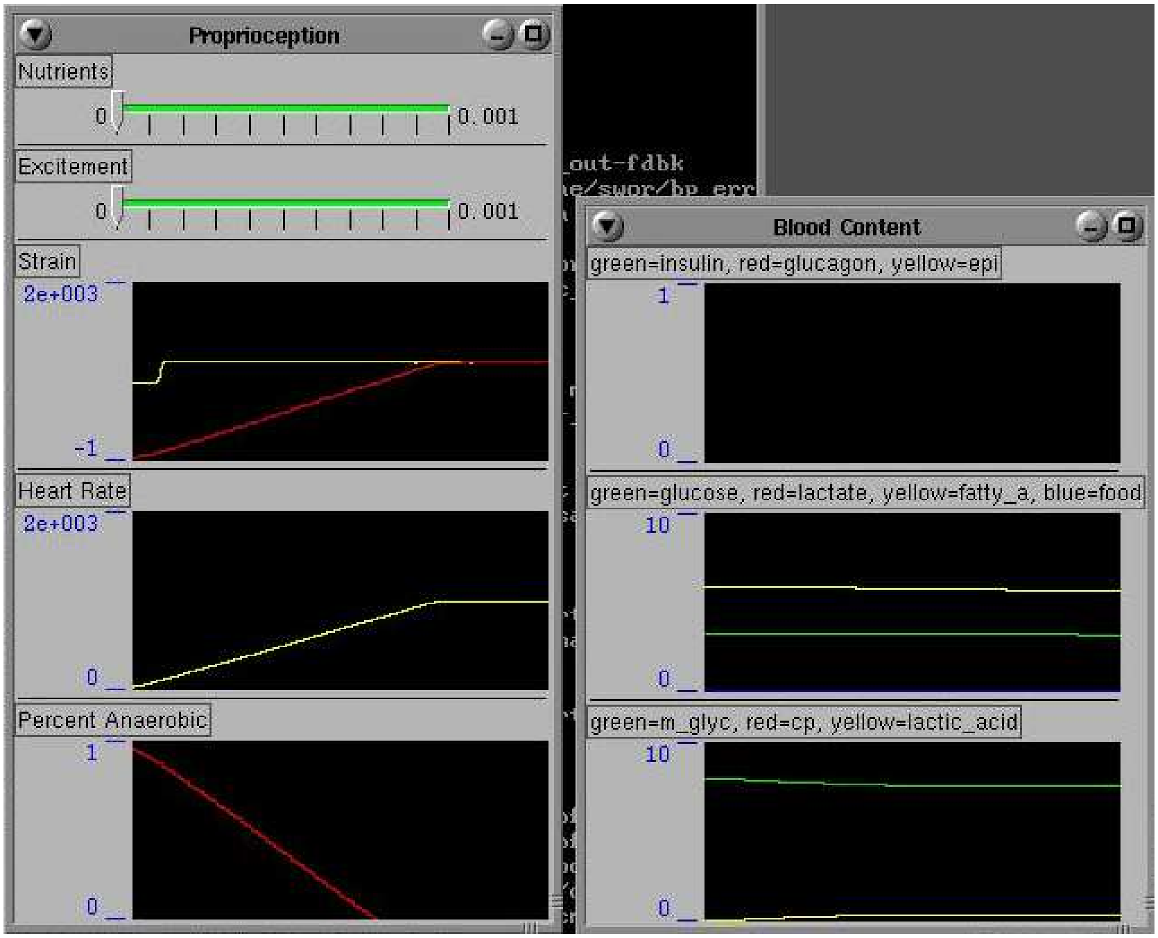

Figure 7 (time displayed: 1m 25s): The exertion in this example brings about an aerobic heart condition, and this allows for maximum fuel efficiency.

In figure 7, the robot s exertion level is elevated to the point where the heart rate is in the target zone. This means that the fuel combustion takes place almost entirely in the presence of oxygen. Consequently, even though the exertion level is higher, the blood fuel levels are still kept at relatively similar levels. Note that, although the exertion started out with an anaerobic component, as it vanished, the fuel consumption and lactic acid production tailed off into the more efficient state. A small amount of epinephrine (yellow trace, top panel, right window) is released due to the stress of the exertion. This causes the other organs to release fuels into the bloodstream, but the majority of the fuel being burned is glucose (provided by the liver s large glycogen store) and fatty acids (provided by the vast fat store in the body).

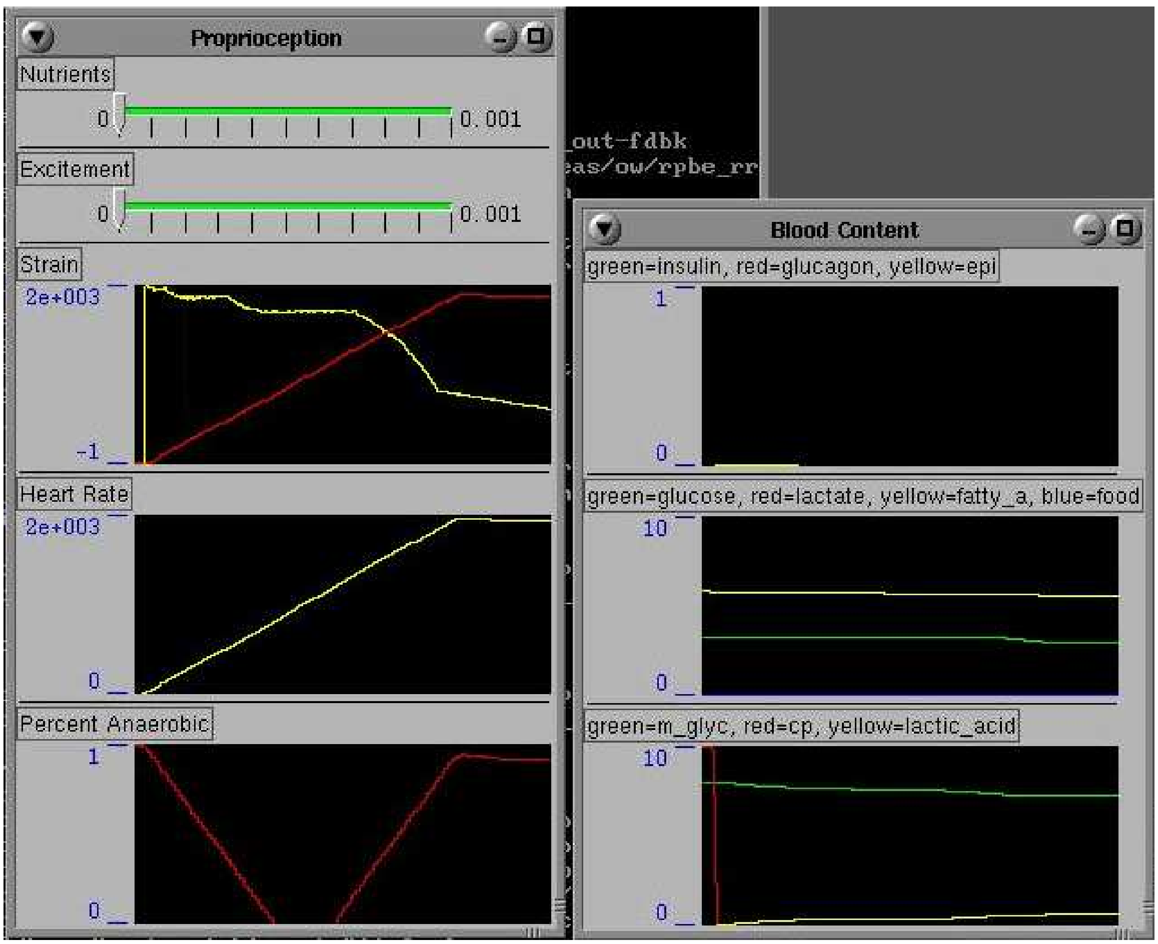

Figure 8 (time displayed: 1m 25s): The exertion in this example is much higher than the robot can sustain for a long period of time, and the fuel levels begin to go out of balance.

Figure 8 represents an exertion that brings the robot s heart rate well above the target zone, and this level of stress actually does bring the robot s fuel levels out of balance. At first, the high exertion is compensated for by a rise in the amount of epinephrine (the yellow trace in the upper right hand corner is visible for a short period), and the exertion can continue. However, as the combustion becomes completely anaerobic, the body is limited to burning large amounts of glucose without oxygen, and the lactic acid level begins to rise to unacceptable levels. These levels cause the reduction in output force (yellow trace, top panel, right window). Given more time, the reduced force would cause a lower need for fuel, and allows the system to return to homeostatic levels.

6 Conclusions

6.1 Summary of contributions

The model outlined in this thesis was designed and built to integrate the behavior of a real robot with biochemical limits and sensory values. Many biologists have built more sophisticated models of different aspects of the human metabolism with success. By simulating each reaction and linking the precursors and products, these models have been able to simulate a small set of biological reactions with a high degree of fidelity. However, they are also almost universally isolated from the real world. Success is achieved by having the simulated reaction result in chemical levels being the same as exist in humans.

This thesis works on a different premise from those models. Instead of attempting to carefully model each chemical reaction (at large computational expense), the goal of this model is to create a more comprehensive system that is efficient enough to run in real time. Further, it interfaces with a robot operating in the real world, so the values must conform not to a measured result of a biochemical reaction, but instead to a representation of how each of the values change in the human system given a pattern of real behavior. Finally, this model also has the ability to intervene and affect the behavior in accordance with the model.

The real contribution of this work, however, will come as more functional systems are constructed utilizing *meso* as a foundation. The goal of a system that generates a broad variety of humanoid gestures will require humanoid inputs. The construction of the robot (human form and range of motion, accurate force sensors) provides some of the foundation for humanoid inputs. However, a system hoping to emulate the way humans gesture must have access to the same information that the biological analog does in humans in order to be successful. *Meso* provides that information, allowing such a system to be developed.

6.2 Future work

Beyond the generalized gesture system that *meso* is intended to support, there are several other interesting interfaces to the biochemical system that could be successfully modeled on the robot. As was mentioned in chapter 4, the implementation currently only supports change to the state of the robot that occurs over the course of roughly an hour. However, longer-term changes could be integrated into the system, allowing the robot to improve its performance over the course of weeks and months. For this work to be meaningful, though, there must either be interactions that occur over these longer periods or some way for the robot to recognize its changes in fitness and work for improvement.

However, the biochemical systems modeled in *meso* also have other effects on the body that could inspire additional modeling on the robot. While this thesis primarily models the effect of energy storage and consumption from the muscle point of view, these energy sources also drive the mental systems of the robot. This effect is easily observed when a tired person will exercise to wake up. The exercise creates a demand for energy from the muscles, and as the various organs put fuel into the bloodstream to provide for this exertion, the brain is also able to gather more fuel and work harder. As the robot develops more cognitive processes, they can interface with the *meso* system as an energy provider to create the ebb and flow of mental exertion that is characteristic of human cognitive processes.

In addition to the effect that energy level has on cognitive processes, the emotional machinery that is currently being investigated by other robots has connections with the systems modeled in *meso*. In our lab, the robot Kismet (Brezeal 2000) has a fatigue drive that increases monotonically. Because Kismet is only a head, this drive may make sense, however, for a fully embodied robot, fatigue should be regulated by the overall behavior of the robot. This model provides that interface, and, with it, a more fully realized emotional model.

Finally, the very inspiration of the *meso* system, to provide a humanoid robot with access to human metabolic senses, could be slightly altered to create a very different system. Instead of giving the robot a simulated human metabolic system, one could imagine a metabolic system that is based around the robot s real energy source, namely, electricity. While Cog is afforded an unlimited and fairly regular supply of energy, a robotic metabolism would be particularly useful in the event that Cog ever needed to limit its power consumption. It might be possible to create a system in which models for the use of temporary power sources in humans (local glycogen stores, phospho-creatine) could be used to regulate a temporary power supply such as a battery pack. Ideally, *meso* could serve to allow the benefits of the human energy regulation system to apply to a humanoid robotic system.

7 References

Breazeal, C. *Sociable Machines: Expressive Social Exchange Between Humans and Robots*. Doctoral Dissertation, Department of Electrical Engineering and Computer Science, M.I.T., 2000.

Brooks, R.A., *The Cog Project* Journal of the Robotics Society of Japan, Special Issue (Mini) on Humanoid, Vol. 15, No. 7, Toshihiro Matsui, editor, October 1997.

Champe, P. C. and R. A. Harvey. *Lippincott s Illustrated Reviews: Biochemistry*. Lippincott-Raven Publishers. 1994.

Chase, P. Bryant and Martin J. Kushmeric. *Effects of pH on Contraction of Rabbit Fast and Slow Skeletal Muscle Fibers*. Biophys. J, Vol. 53, June 1988, pp. 935-945.

Fell, David. *Understanding the Control of Metabolism*. Portland Press, 1997.

Gilbert, Hiram F. *Basic Concepts in Biochemistry*. McGraw-Hill, Inc., 1992.

Hogan, Neville. *Mechanical Impedance of Single- and Multi-Articular Systems*. Multiple Muscle Systems: Biomechanics and Movement Organization, J. M. Winters ans S. L-Y. Woo, editors. Springer-Verlag, 1990.

Marjanovic, Matthew. *Sok: An Architecture for Behavior-Based Control*. M.I.T. A.I. Laboratory Internal Document, 2000.

Marjanovic, M., Brian Scassellati, and Matthew Williamson. *Self-Taught Visually-Guided Pointing for a Humanoid Robot.* Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior (SAB-96), Cape Cod, MA, September 1996.

Massaquoi, S. G. and Jean-Jacques E. Slotine. *The intermediate cerebellum may function as a wave-variable processor*. Neuroscience Letters, Volume 215, 1996. pp. 60-64.

McMurray, W. C. *Essentials of the Human Metabolism*. Harper & Row Publishers, 1977.

Okunieff, Paul G. *A Dynamic Systems Model for Muscle Biochemistry*. B.S. Thesis, Department of Electrical Engineering and Computer Science, M.I.T., 1978.

Pratt, G. and Matthew Williamson. *Series Elastic Actuators*. In Proceedings of IROS95, Pittsburg, PA. 1995.

Scassellati, B. *Eye Finding via Face Detection for a Foveated, Active Vision System.* In Proceedings of AAAI98, Madison, WI., 1998.

Sizer, F. S. and E. N. Whitney. *Nutrition Concepts and Controversies*. West/Wadsworth Publishing. 2000.

Stein, R. B. *What muscle varible(s) does the nervous system control in limb movement?* The Behavioral and Brain Sciences, Volume 5, 1982. pp. 535-577.

Williamson, M. *Robot Arm Control Exploiting Natural Dynamics*. Doctoral Dissertation, Department of Electrical Engineering and Computer Science, M.I.T., 1999.