A Control Centred Approach to Designing Interaction with Novel Devices

G.J. Doherty[a], T.M. Anderson[b], M.D. Wilson[a] and G. Faconti[c]

[a]CLRC Rutherford Appleton Laboratory, Oxfordshire, U.K.
G.J.Doherty@rl.ac.uk, M.D.Wilson@rl.ac.uk
[b]Drake Music Project, U.K. http://www.drakemusicproject.com.
timanderson@drakemusicproject.com
[c]CNR Istituto CNUCE, Pisa, Italy. G.Faconti@cnuce.cnr.it

Abstract

Modern information technology is becoming both increasingly ubiquitous and increasingly varied in the possible ways the user can interact with it. Accompanying this, there is a trend towards interfaces where the user is in constant interaction with the computer system, communicating with it by many different means, such as gestures, speech and haptics as well as discrete communication. This development requires new interface design approaches that allow for the analysis of the continuous as well as the discrete aspects of the interface, and that support reasoning about real−time issues. In the area of manual control, theories have been developed for continuous control of systems by human operators. In this paper we examine how we can apply manual control concepts in a qualitative fashion to the design and analysis of interactive systems. This involves a focus on control and feedback signals, transformations of these, and control characteristics of user, device and controlled process. While we make reference to the particularly challenging application area of performance control systems for disabled musicians, we believe that control issues of this nature will become increasingly common in interface design.

1. INTRODUCTION

In this paper, we look at the analysis of a class of novel and emerging interface technologies where interaction between system and user is, in some sense, continuous. In modern interfaces using techniques such as gesture recognition, speech recognition, animation and haptic feedback, the user is in constant and closely coupled interaction with the computer system over a period of time. The interaction is no longer based on a series of steps or discrete interactions, but the input provided by the user and/or the output provided by the computing system involve a continuous exchange of information at a relatively high resolution. Applications that use continuous interaction techniques can be found in virtual reality and teleconferencing, but also in less obvious areas such as ubiquitous computing (including active and intelligent environments), teleoperation, and alternative interface technologies for people with sensory and motor impairments.

To help us with the analysis of such interfaces, we look to a branch of engineering psychology dealing with manual control.

2. MANUAL CONTROL

Manual control theory was originally developed by feedback control engineers modelling tasks such as tracking for anti−aircraft gunners. However, the theory is applicable to a wide range of tasks involving vigilance, tracking, stabilising etc. (for example driving a car or piloting an aircraft). The theory, particularly that branch developed from control theory, has been refined to a very high degree over the years. A general introduction can be found in (Salvendy, 1997). There is a large base of both predictive (McRuer 1980) and explanatory theory (Hess, 1985) based upon, and validated by, a wealth of experimental data. In the control theory approach, continuous mathematics is used to model human performance. The focus of the approach is on the perception and transformation of signals representing for example the actual and desired state of a process. Motor performance is viewed in terms of information transmission, with inaccuracy viewed as additive noise.

A number of qualitative concepts from control theory can be used in describing human−machine interaction. This is shown by (Jagacinski, 1977) which looks at a number of these. The first of these is that of open and closed loop control. In open loop control, only the target signal is available to the user; thus there is no ability to account for noise or environmental interference. In closed loop control by contrast, both the target and output signal (fed back) are available, giving the user the opportunity to compensate for error. In musical performance for example, the target signal is the desired sequence of musical outputs; the fed−back signal is the current musical output. Another concept is that of positive and negative feedback. In a negative feedback system, the tendency is to minimise errors caused by disturbance; in positive feedback systems the tendency is to amplify disturbance (ie. they are unstable). While positive feedback may be useful in the design of experiments, the vast majority of manual control scenarios involve negative feedback systems. A third concerns gain and time delay, and is discussed below.

2.2 Gain and time delay

Consider a simple closed loop negative feedback system which we describe with two parameters, firstly a delay or latency *t* which is the time taken by the controlled element to react to it's input, and secondly the gain *K* which determines the rapidity of adjustment. If *K* is low, the system will respond very sluggishly moving only slowly towards the target signal. Conversely, if *K* is high, then the system is likely to overshoot, requiring adjustment in the opposite direction which itself may overshoot, leading to oscillation. The delay *t* can also contribute to this behaviour − a high delay makes oscillatory behaviour much more likely. Additionally, for most performance systems where music is directly output, *t* must be low (of the order of 20ms) to produced a perceived *immediate* output. If *t* is much higher, most musicians become unwilling to perform using the system.

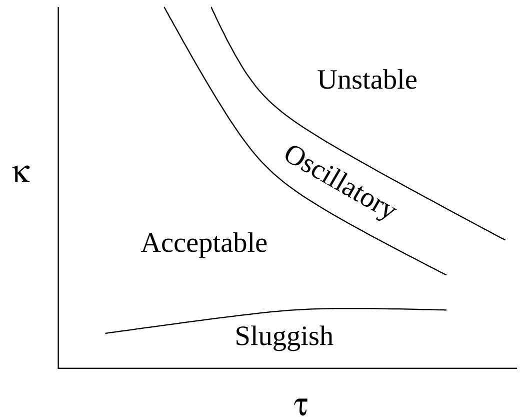

Figure 1: Effect of gain and time delay parameters on control (adapted from Jagacinski, 1977)

This is a particular type of real time control, but it is a particularly common one, useful wherever system delay is a performance shaping factor. For example, the above view was developed with reference to a small time scale (on the order of the delay between seeing a system output and carrying out the motor actions for an appropriate response), one can also consider it over longer time intervals. For example, consider an in−car navigation system where instead of gain, we have the frequency of decisions taken on which route to take and the time delay is that between a certain position being reached and appropriate instructions being displayed within the car. The parameter space of such an application should be very similar to that in figure 3 above.

3. APPLICATION: CONTROL SYSTEMS FOR DISABLED MUSICIANS

In this section, we give some background on the particular application area which we focus on, before looking at the control and feedback signals present, possible transformations between these, and control characteristics of user, device and controlled process.

The Drake Music Project has for many years been developing control systems to enable musicians with a wide range of physical disabilities to play music, solo or within a group (Anderson, 1997), adapting commercially available components, as well as developing its own "E−Scape" software. The latter can allow limited bandwidth control signals to be converted into complex or subtle music output, by allowing a performer to do some of the creative work "offline", ie. pre−compose and assemble musical material. This allows the number of input parameters and/or value ranges to be reduced during subsequent continuous interaction with the system, such as in a live performance (Anderson, 1999).

A user's input can be derived from a variety of actions, eg:

- movement in space, with position detected with up to 6 degrees of freedom (eg three x,y,z Cartesian coordinates, plus three orientations: azimuth, elevation, roll), eg by video, radio, ultrasonic, capacitive, or infrared sensors),
- interaction with physical devices, eg balls, joysticks, switches, pads, again with 1−6 degrees of freedom.

However, for disabled performers, it is most important to maintain flexibility of detection and filtering of actions: for example, many performers initially find their best musical results come from utilising only 1 degree, but then want to progress to more, with increasing input range and discrimination. User actions to operate music systems in current use can include:

1. Generating a series of related values, by:−

(i) gestural movement through an area in space. In typical music sensors, movement is detected in 1 direction, eg. radial movement within a hemispherical detection zone, or longitudinal distance along the axis of a conical zone.

(ii) 2D movement of finger, toe (or even nose) on mousepad, or trackball.

(iii) varying pressure on a squashable pad, eg. the "MIDIpad" device developed by Drake at the University of York, Department of Electronics.

(iv) 2D movement *and* pressure, eg. mousepad−like device "MIDIslate" (York).

2. Generating a trigger event, by movement *at a specified time*, by:−

(i) movement into or out of the detection zone (1.i. above)

(ii) movement (without contact) to a specific location in space,

(iii) movement to, and pressing of, a physical switch device (or a key or position on computer or concept keyboard)

These actions can then provide input signals to the E−Scape music engine, such as:

1.a. Apply continuously varying timbral parameters to music events already playing (or previously started by other signals). Such parameters can include:

- pitch "bending",
- low pass filter frequency (often used with some resonance to give a "filter sweep"),
- volume,
- stereo pan position,
- relative volume or pitch of sonic components ("vector synthesis")

1.b. Trigger events from a pre−composed set (usually related, eg. an arpeggio, or set of phrases which make up a piece). Points at specific locations within the detection zone are mapped to event triggers; a many to one mapping can enable low accuracy positioning to reliably trigger a smaller set of music events.

2.a. Trigger a pre−composed music event to start playing (a single note, a chord, short phrase, or larger music segment). Additional input parameters (eg. to set attack time, onset loudness etc.) also have their values pre−composed (ie. planned beforehand) and embedded in each event.

2.b. Trigger a music event, plus set parameter values (as in 1a). The range of input values needed can be reduced if desired, by again having pre−composed values embedded in the event, which the input value can then alter, to a greater or larger extent.

2.c. Trigger events in turn from a pre−composed list, eg either the next or previous event.

Trigger events (2) can also be derived from *processing* continuous user movement (1). A good example in E−Scape is the creation of trigger events from analysis of user path. For example each reversal of motion direction could trigger the next event in a series. This can enable a performer to produce a natural "conducting" action, eg. by nodding head, or waving a leg.

The challenge facing the designer of this type of application is how to match the capabilities of the users, who have varying degrees of motor skill, via a control system, to the space of needed input parameters. Particularly interesting is the question of how to achieve acceptable performance in continuous real−time control, as might be the case in the context of a live musical performance.

3.1 Control characteristics of user, device and controlled process

A useful place to start is to examine how the capabilities of the user may vary, the kinds of input the system may require to achieve a task, and the possible mappings between the two. To match a control system to the *users abilities*, we require a characterisation of these abilities, since familiar results like Fitts' Law (see Mackenzie, for a review) may not hold for users with sensory and motor impairments, or where there are environmental constraints.

- What independent motor control capabilities does the user have available
- What range of movement (distance, angle, discrete values) can be produced
- What is the accuracy or precision of movement (avg. distance, angle or rate of error).
- The speed of the movement (m/sec, rad/sec, inputs/sec)

Embedded within the physical form of the *input device* there may be a number of transformations of the forces applied by the user, or their movement in free space detected by a sensing mechanism. For example a microswitched joystick transforms an angular input into a number of discrete possibilities; an ultrasonic beam (eg. EMS Soundbeam) has a number of discrete positions it can discriminate between, or can produce different output depending on the direction of approach to a position. The input device may also encapsulate a given *control dynamics*:

- the order of the control system (e.g. is it a distance, velocity (first order) or acceleration (second order) control). Higher order systems may allow us to produce a wider range of outputs, and quickly move between very different output values, but they are also more difficult to control, and more sensitive to feedback latencies.
- the input gain (what magnitude of change in output is produced by a given change in the input). In music applications, this can often be constrained, eg. a pitchbend of more than two semitones is musically inexpressive, and so the gain must be chosen with this limited range in mind.
- time delay (if there is feedback at input level). What is the delay between an input and some response to that input

An important distinction concerns the nature of the output of both the physical controller and the control system; whether it generates *event* or *continuous output.* An event output is generated at a particular point in time. With continuous output, some output is provided constantly. (This could be the "neutral" position of a bang−bang controller). The values of both event and continuous output channels can have either a discrete range or a continuous range. This characterisation applies to both the physical controller and the control system. The control system might in fact transform between these, eg. sampling will turn a continuous output channel to an event output channel. The point at which a sample is taken could also be assigned to a user controlled channel (eg. a single switch). Transformations can also be applied to the output values, for example quantisation to transform from a continuous range to a discrete range. In the context of the music application, the transformation of a continuous−valued output channel (eg lateral position of foot over a music keyboard) into one with discrete values is prone to errors. The number of discrete events depends on the accuracy of continuous output, and hence filtering of the signal produced is very important. Hence, it has been found to be useful to have "null" values within a continuous output channel, eg. where user cannot guarantee achieving the desired output value reliably enough. A good example of such processing in E−Scape is the dynamic re−mapping of music keyboard input − any zone of keyboard keys (eg foot positions) can map to a single music event (eg note), and other keys (eg the "black notes") can be mapped to zero output. Thus, if the user presses keys at the side of the desired white key, only that note will sound, and slipping onto the black keys makes no sound. Of course, the downside is that fewer output events can be directly controlled, but with careful preparation and/or splitting of control channels this is not a problem − see 3.2 below.

The *input parameters* to the system or controlled process could concern many different aspects of a given task. If the controlled process is to trigger phrases of music, then aspects would include which phrase to play, loudness, transposition, tempo, voice (instrument) used, etc. At the most basic level, we have:

- range − what possible values the input can take
- whether the inputs are continuous or discrete
- delay (if there is feedback at output level).

An additional concern where there are more variables to control than independent input channels is that some form of input moding must be implemented. For example, a music engine might accept a limited number of values from a user input to trigger musical events. When a particular value (eg. an end point) is received, the input is used to trigger events from a different set.

3.2 Design tradeoffs

We have from the above some requirements for information in order to design a system. There are a number of parameters we wish to control, for which we may identify a range of values, and the nature of the control required (discrete precisely timed inputs, rate of input needed, continuous control). For each user or class of user, there are the available motor control channels and the allocation of these channels to control parameters. As stated above, the control system which matches these two may itself transform the control signals. These three facets of the application are mutually constrained, and design is necessarily an iterative process. Inevitably, there is a tradeoff to be made of *expressiveness* against the *accuracy* required for real−time performance. This increased accuracy may be achieved by decreasing the resolution of the controlled process parameters and also by use of filtering. The main design goal here is to make *best use of available motor control*, in the case of our particular application area, giving the *greatest degree of musical expression*.

Where there is difficulty, there are two obvious design alternatives;

- Where the information required by a parameter is more than that on any available motor control channel, more than one channel can be assigned to a given parameter. An example of this would be to combine foot and knee position. The foot used to control a set of 10cm wide pads in a row on the floor, with a sideways range of 1m allowing 10 discrete pad values, and the knee rising to enter an ultrasonic beam. If the parameter to be controlled is pitch, then a number of combinations are possible, for example the foot selecting between 10 pitch values within a set, and the knee changing the set of pitches in use each time it is raised.

- Conversely more than one parameter may be assigned to a given control channel. An example of this would be use foot position to control chord type, root note and volume. Again with a set of foot pads, one might be used to toggle high or low volume, several could be used to pre−select (but not trigger) chord shapes (eg. minor triad, major 7th), with the remaining pads used to play a chord with a choice of root notes (eg. C,E,F,G,A) with the selected volume and shape.

Such compromises can still produce surprisingly good musical results if carefully designed, particularly when events consist of higher−level musical structures such as loops or phrases, which give "covering" delay while another input is selected.

The designer should be aware that particularly for real−time tasks, the information transmission rates of the combined channels is likely to be less than the information transmission rates of the channels when used in isolation, and similarly for split channels. Where no acceptable assignment of motor control to process parameters can be found, the informational requirements of the controlled process must be decreased; in many cases preparation of pre−composed material can mitigate the loss of expressive control.

3.3 Design representations

A simple box diagram and filter notation can be very useful for high level modelling, development of this might also be useful for resolving issues of software and hardware architecture. This can also encode issues such as event and continuous output channels, continuous and discrete values and so on. Having a structured design representation can help us to consider the likely effect of varying the dynamics of the controlled system (input gain, order), and system performance (feedback delay) or additional feedback earlier in the loop. We can also anticipate the need for filtering (a tradeoff of expressiveness vs. accuracy).

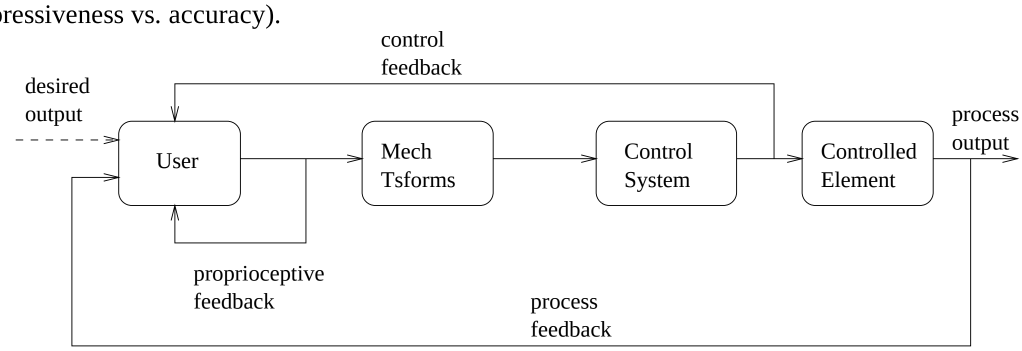

Figure 2 − Different levels of feedback within control loop

For example, consider the issue of feedback. We discussed above how feedback gain and latency parameters can affect real−time control performance. However, when discussing this, we should remember that feedback can be present at several levels. At the lowest level, the user’s own body provides proprioceptive feedback on position of limbs (although some disabled users have far less). Also, physical device characteristics may provide feedback at this level (eg. where there is a limited range of movement). Next, we may have feedback about the control inputs generated by the input device (eg. position of a head controlled switch). This is extremely important where there is significant latency in the controlled process, for example the position of an electric wheelchair (see Doherty and Massink, 1999 for more discussion of time issues), or where a delay is deliberately introduced (eg. where notes are placed in a queue before playing) and it becomes important to be able to tell exactly when (and whether) the input event was triggered. Finally we have the controlled process state (the music heard), which in many cases will be the most important source of feedback.

Consider for example the pitch control configuration described in section 3.2 above (see figure 3). The output channels and the values carried can be concisely represented, allowing us to see at a glance the association between motor control channels and controlled process, and the transformations applied to both control and feedback signals. Such representations serve a dual purpose, helping us both to evaluate designs with respect to feedback and control issues (expressiveness), and to encode design alternatives in a fashion accessible to participants from different backgrounds in the design team. Representing design options in a structured and concise fashion helps us to reduce complex design issues to a form where we can hope to find answers in the

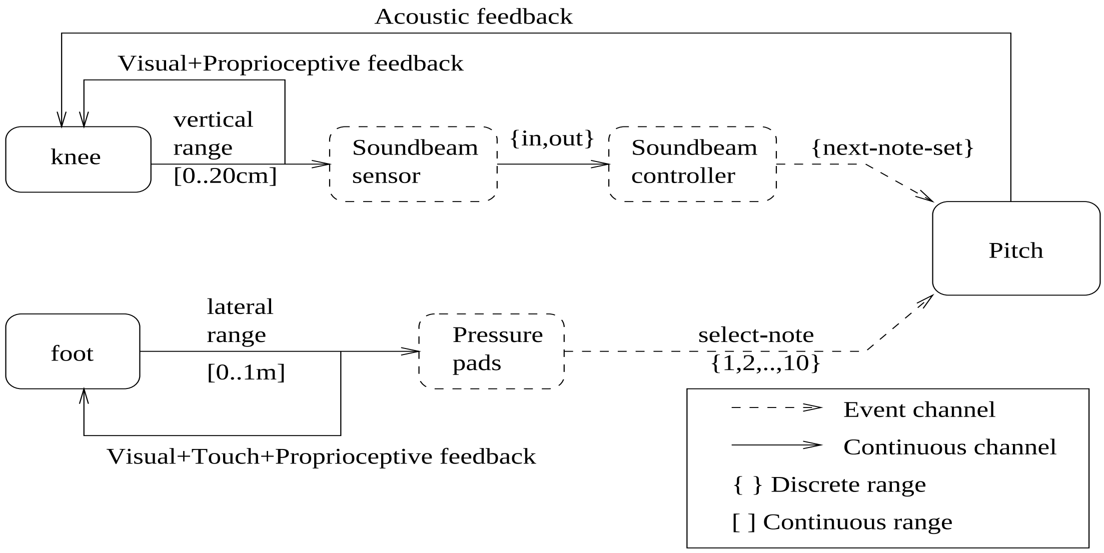

Figure 3 − diagrammatic representation of a control configuration

human factors literature, or by means of straightforward reasoning.

4. CONCLUSIONS

The advent of a number of modern interaction technologies means techniques must be developed to allow designers to consider issues raised by continuous interaction between users and computer systems. In this paper, we have looked at the application of concepts from a branch of engineering psychology (particularly manual control theory) to the problem. This approach requires that special consideration is given to control and feedback signals, and transformations of these signals. We have investigated the application of this view to interactive system design by considering systems for music performance by disabled musicians. In this context, we have looked at characterisation of control aspects of user, device and controlled process, some of the tradeoffs involved in designing a control system, and simple graphical representations of control system configurations.

The needs of disabled musicians within the Drake Music Project when performing music provide a wide−ranging and demanding test−bed for these ideas. In terms of future work we are interested in the guidance that manual control theory can give in the choice of control system. Qualitatively, we would like to develop some guidelines, driven by user and controlled process characteristics. A full control theoretic treatment of this application would be problematic for several reasons; individual differences in sensory motor skills are very large, very few control scenarios are "pure" enough to facilitate an equational specification, and the benefits of the research effort would not justify the cost. However, with appropriate empirical data, we believe some simple quantitative estimations could be developed.

Acknowledgement This work was supported by the TACIT network under the European Union TMR programme, contract ERB FMRX CT97 0133.

BIBIOLOGRAPHY

T.M. Anderson (1997). Making music with Computers. *Ability* 209:9−15.
T.M. Anderson (1999). Using Music Performance Software with Flexible Control Interfaces for Live Performance by Severely Disabled Musicians. *Proc. EuroMicro 25(Vol. 2):* 20−27.
G.J. Doherty and M. Massink (1999). Continuous Interaction and Human Control. *Proceedings of XVIII European Conference on Human Decision Making and Manual Control*, Group−D Publications.
R. A. Hess (1985). A model based theory for analyzing human control behaviour. *Advances in Man−Machine Systems Research*, 2:129−175.
R.J. Jagacinski (1977). A qualitative look at feedback control theory as a way of describing behaviour. *Human Factors*, 19:331−347.
G. McRuer (1980). Human dynamics in man−machine systems. *Automatica* 16(3):237−253.
G. Salvendy, editor (1997). *Handbook of Human Factors and Ergonomics.* Wiley Interscience, 2nd edition.
I. Scott MacKenzie (1992). Fitts’ Law as a Research and Design Tool in Human−Computer Interaction, *Human Computer Interaction*, 7:91−139, Lawrence Erlbaum Associates, Inc.