

Real-Time Control for VIDET using Simulink

G.E. Smid*, C. Melchiorri

DEIS-LAR, Dip. Elettronica Informatica e Sistemistica

University of Bologna

Via Risorgimento 2, 40136 Bologna, Italy

Abstract

In this paper we discuss the real-time force-feedback control structure for the VIDET system. In this application the scheme for the controller is defined as a dual cooperative task. The control force setpoint is derived from the human response interface, whereas the actuator position is controlled in real-time so that the force setpoint is maintained.

After introducing the application, the control system architecture will be explained. Finally the implementation and realization in MATLAB-Simulink is discussed.

Keywords : Real time, tension control, Virtual Reality.

1 Introduction

The VIDET (acronym for *Visual Decoder by Touch*) is a haptic interface based on tensioned wires. The device has been developed to be tailored on the back of a human body in order to obtain a portable haptic interface, that can be used for achieving haptic perception by visually impaired people. The VIDET project is explained more elaborously in [1] and [2]. Figure 1 shows a diagram of the system.

Suitable applications for the system are for ex-

*Department of Electrical & Systems Engineering, 128 Science and Engineering Building, Oakland University, Rochester, MI 48309 USA.

ample approaching and grasping hot or fragile objects, or to create virtual walls before traffic or water or other hazardous areas.

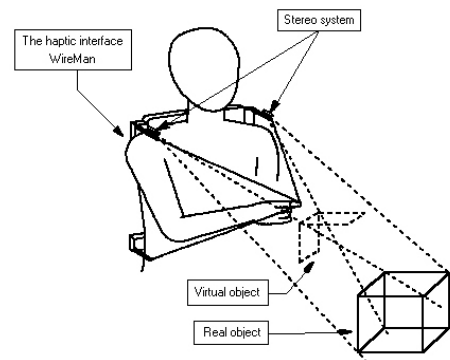


Figure 1: The VIDET haptic device is carried on the back.

The control structure has the task to read the thimble position and to apply a tension on the wires when the thimble approaches a virtual surface in space. Therefore, another part of the ViDeT subsystem will define the reference database that is used by the control system as the virtual environment of the surfaces in space. This vision subsystem is described in more detail in [4].

As an illustration of the controller operation, the example of picking up a cup of coffee. Imagine a visual impaired person who would like to grab a mug of coffee, that is located in front of him/her.

The stereoscopic vision system will create a virtual surface database of the frontal environment of the person, one inch away from actual surfaces. During the free space movements, the wires are kept under minimum tension, so that a correct com-

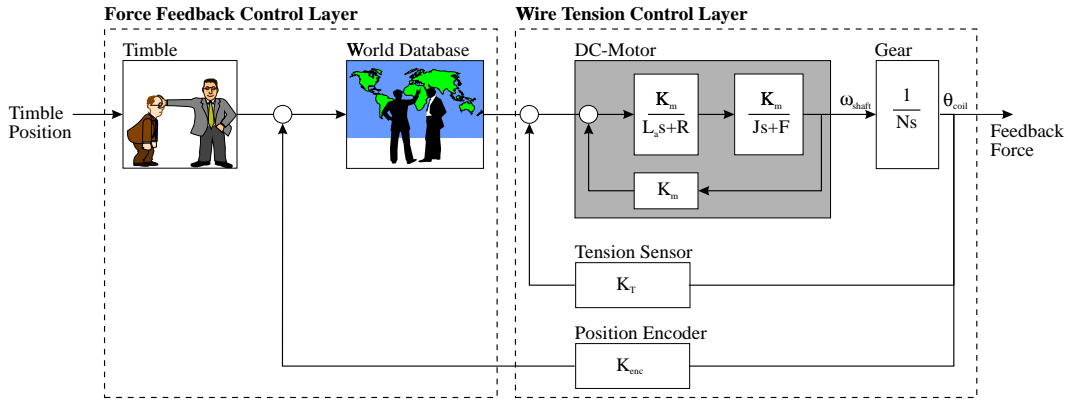


Figure 2: Control system diagram.

putation of the finger's position can be conducted. This tension is weak enough to allow the user a full freedom of movements. Reaching out with the thimble to explore the environment, the person will experience force feedback along a virtual surface. The person will thus be able to precisely find the coffee mug before touching it.

When the thimble is "in contact" with the virtual surface, a force is computed, which is proportional to the penetration of the finger into the surface.

The force is calculated to be perpendicular to the surface, which is the desired force that would realize the correct feeling. However, this force can be generated at the thimble only if its vector lies inside the pyramid shape that is formed by the wires. All other vectors will have negative force components along some of the wires, so they cannot be realized.

In the latter cases, the wires that would apply the negative force are kept at a positive minimum tension, only to be able to wind the wire as the thimble approaches. Different approaches can be thought of to approximate natural force feedback. A more in-depth study on the force control strategies is described [3].

2 Control System Design

The control system has been designed from the above task requirements. For efficient computation

speed, two different control layers have been designed, with two adequate timing methods. They are the wire tension control layer and the force feedback control layer. They are the two dashed boxed areas in Figure 2. The wire tension control layer is implemented as an interrupt handler, and the force feedback control in a Matlab-Simulink model. The implementation and communication is graphically represented in Figure 4.

2.1 Wire tension control

This layer maintains the setpoint tension in the wires. Three actuation modules are mounted on the system in such a way that the enclosing pyramid by the three wires is maximized.

Each module contains a main collecting pulley and an idle pulley over which the wire is routed. The idle pulley is connected to a small flexure beam that is instrumented with a strain gauge bridge, from measuring the wire tension. The collecting pulley is driven by a worm gear speed reducer with transmission ratio of 1:25, which connects to a brushless minimotor. The construction of the actuator is drawn in Figure 6.

The wire tension is maintained at setpoint level by a real-time control algorithm. The implementation for the algorithm is shown in the diagram in Figure 4. A hardware clock generates CPU interrupts at a 200 Hz frame-rate. During each clock pulse an interrupt handler will execute one control

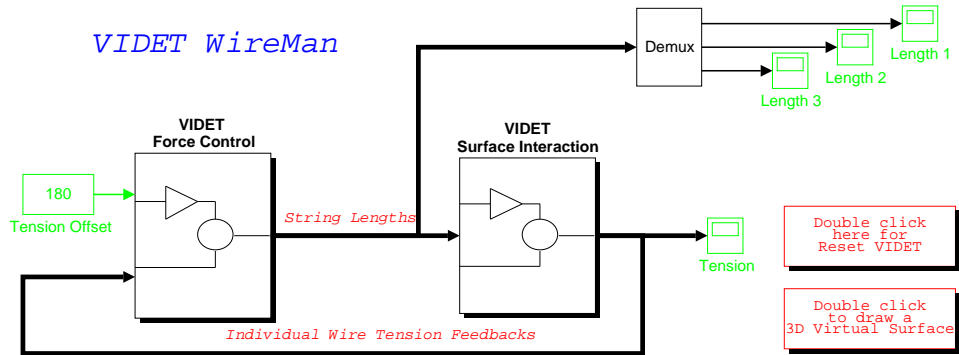


Figure 3: Control system diagram for the VIDET controller in Simulink.

sample step.

The implementation of the interrupt code in a dynamic link library (DLL) to be used in MATLAB functions is given below.

```

void Controller(void) {
    for (i=0;i<3;i++)
        torques[i]=(controls[i]-forces[i])*P[i];
}

void interrupt far
    (*oldhandler)(__CPPARGS);

void interrupt far handler(__CPPARGS) {
    function
        INTF = TRUE;
        ReadPos();
        GetForces();
        Controller();
        SetTorque();
        INTF = FALSE;
        _enable();
        outp(0xa0,0x20);
        outp(0x20,0x20);
}

void InitVIDET(void) {
    oldhandler=_dos_getvect(INTR);
    _dos_setvect(INTR,handler);
    _disable();
    outp(0xa1,0x1d);
    _enable();
    StopMotori();
    InitTimers();
    initialized = 1;
}

}

void UnInitVIDET(void) {
    outp(CMUX,0x19);
    _disable();
    outp(0xa1,0x9d);
    _dos_setvect(INTR,oldhandler);
    _enable();
    outp(DISABLE_MOT,0);
    initialized = 0;
}

void far mexFunction(
    int nlhs, Matrix *plhs[],
    int nrhs, Matrix *prhs[]) {
    double far *sens_val;
    int motor, n=1, m=3;
    if (initialized==0) InitVIDET();
    if (nrhs==0) UnInitVIDET();
    if (nrhs>=1) {
        motor = mxGetScalar(prhs[0])-1.0;
        if (motor<0 | motor>2)
            mexErrMsgTxt("Invalid Parameter");
    }
    if (nlhs==1) {
        plhs[0] = mxCreateFull(n, m, REAL);
        sens_val = mxGetPr(plhs[0]);
        if (motor<3 & motor>=0) {
            sens_val[0] = positions[motor];
            sens_val[1] = speeds[motor];
            sens_val[2] = forces[motor];
        }
    }
    if (nrhs==2)
        controls[motor]=mxGetScalar(prhs[1]);
    return;
}

```

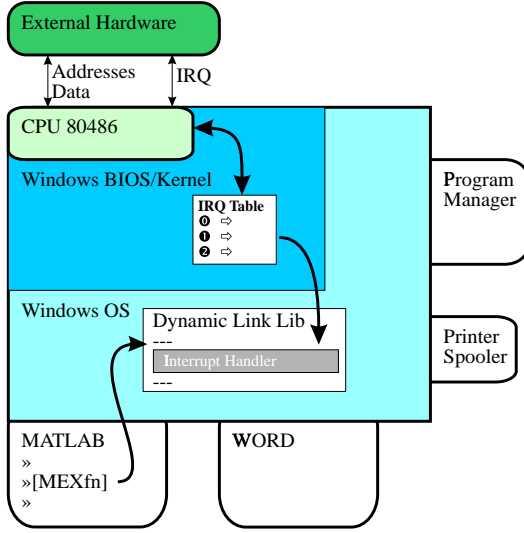


Figure 4: Diagram of the real-time control structure in the windows environment.

From the MATLAB environment, the execution point into the code is at the declaration of `mexFunction`. If the function is called for the first time, the interrupt handler and the counters will be initialized.

The function of successive calls depends on the calling syntax. If two parameters are passed, the tension control setpoint is set for the particular wire. If only one parameter is passed, then sensor values for position, speed and forces are returned for the requested actuator.

From the code it can be seen that the implemented force control method is a simple first order P-controller. It is not hard for the system to become unstable, since the strings and the strain gauge bridge do have a certain spring coefficient. The instability can be noticed when the thimble is mounted to a rigid anchor. The oscillation will eventually break the string.

However, the thimble is a human interface and the motion of a human hand has sufficient compliance to damp the oscillations. More than that, oscillations in the force control can be felt only when a fast course motion manouever is demanded. An alternative solution for force control has been studied and will be discussed in Section 3.

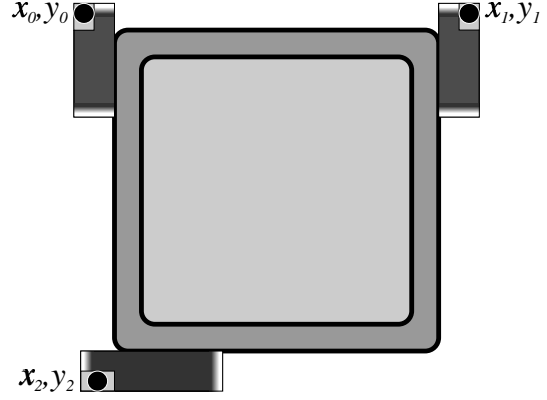


Figure 5: The Videt-System Wire-man, seen from the front.

2.2 Force feedback control

The force feedback control layer derives new tension setpoints from thimble position and the virtual environment database. This layer computes the kinematic formulation for deriving the thimble position in space from the wire lengths. The comparison with the environment database will solve for the penetration depth into the virtual surface. This virtual surface is created by the stereoscopic vision subsystem [4].

Since perceptual exploration is assumed to be a relatively slow exercise, the sampling rate will be less than for the force control. Therefore the simulation sampling rate of the Simulink software will be used. Figure 3 shows the model diagram for the overall force feedback control system.

In the diagram, the block VIDET Force Control communicates with the tension controllers, to set the wire tensions and read the wire lengths.

The block VIDET Surface Interaction applies the following coordinate transformations to compute the 3D location from the wire lengths,

$$x_t = \frac{(l_0^2 - l_1^2 + x_1^2)y_2}{2x_1y_2} \quad (1)$$

$$y_t = \frac{(l_0^2 - l_2^2 + x_2^2 + y_2^2)x_1}{2x_1y_2} - \frac{(l_0^2 - l_1^2 + x_1^2)x_2}{2x_1y_2} \quad (2)$$

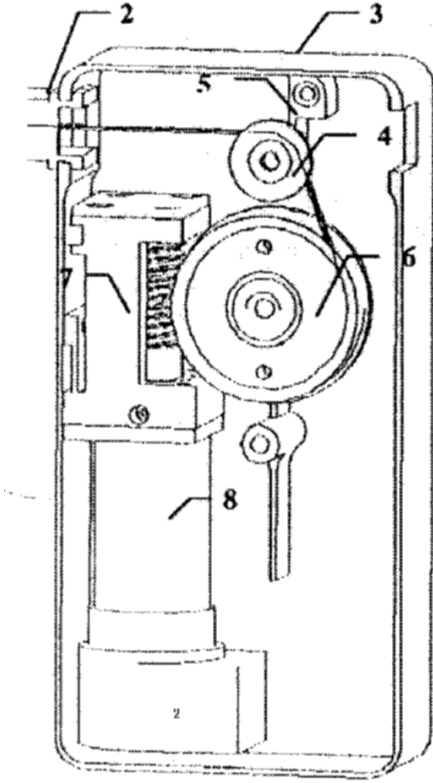


Figure 6: The wire actuation module

$$z_t = \sqrt{l_0^2 - x_t^2 - y_t^2} \quad (3)$$

where x_1 , x_2 and y_2 are the non-zero actuator locations as can be seen in Figure 5. This block also reads the virtual surface and computes the appropriate perceptual force for the thimble.

The control strategies for computing the feedback force with the three wires is discussed in [3].

3 Co-axial Force Control

It was discussed in Section 2.1 that the current construction of the actuation module may show some oscillations. Experiments have been conducted to search for a different configuration of the winding construction, that allows a more stable controller design.

Therefore the general second order dynamic model for the brushless motor is considered. See

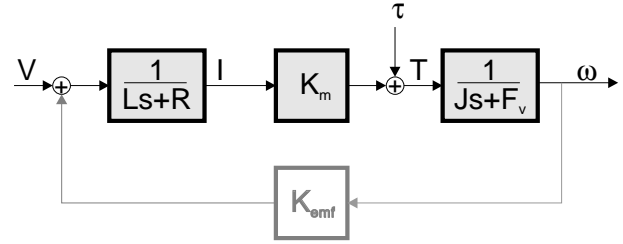


Figure 7: The general model for a DC-motor

Figure 7. When the axis of the motor is stalled, the emf-feedback force is negligible. The signal is drawn in gray in Figure 7.

Consider a co-axial winding construction where the winding pulley is mounted directly onto the motor shaft. Assuming relative slow perceptual motion, this feedback gain can be reasonably ignored. Therefore, the current through the motor coils is proportional to the torque that the motor applies to the motor shaft.

With this assumption, the wire tension control problem is translated into an electrical dc-motor current control problem. The co-axial actuator construction has been built for a 2-dimensional VIDET. A thimble on two wires, can be moved across a surface, in which the virtual wall is represented as a curved boundary in the plane.

Experiments show excellent capability for exploring an artificial surface with co-axial force control. In Figure 8, the top-right figure is the curve for the artificially created boundary. The bottom graph is the recorded 2D data points from experimental exploration with the co-axial actuator construction. The graphs on the left in Figure 8 display the torques and the lengths for the left and right wires respectively.

The disadvantage however, is that the dissipation in the dc-motors is considerable, especially when a reasonable feedback force is desired. the non-reversible construction in Figure 6 does not need a constant current in order to apply a constant wire tension.

Table 1 shows the average currents for the co-axial (without sensor) and the irreversible (with force-sensor) actuator constructions. The table

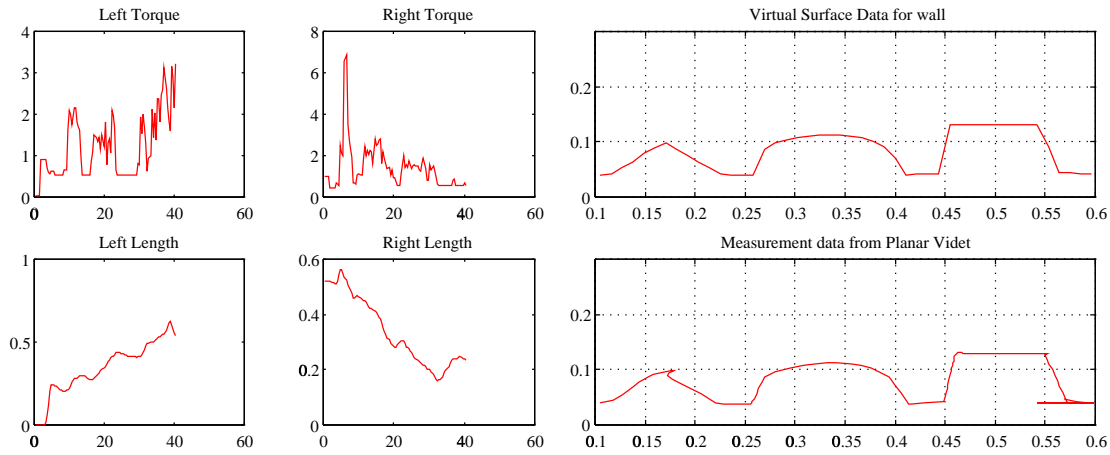


Figure 8: Experimental results when exploring a 2D virtual boundary surface

Average Current	Free Space		In Contact	
	Stationary	Motion	Stationary	Motion
Co-axial	0.1 Amp	0.1 Amp	<0.75 Amp	<0.75 Amp
Module	0.06 Amp	<0.25 Amp	0.06 Amp	<0.25 Amp

Table 1: Typical currents per motor with and without force-sensor.

shows four different typical working conditions. Eventhough the force is proportional to the penetration depth into the virtual surface, for an average human perception, and a natural feedback force, the typical load is measured as given in table 1.

Since energy efficiency and perceptual force feedback are the objectives for the project, the VIDET system will be equipped with the current actuator modules.

Acknowledgements The VIDET project is a joint research programme of DEIS, DIEM, DM of the University of Bologna. Financial support is provided by University of Bologna CNR and ASI. Stimulating discussions with Prof. C. Bonivento (project coordinator) and Dr. A. Eusebi and the contribution to the experimental activity of D. Rossi are gratefully acknowledged.

References

[1] L. Di Stefano, A. Eusebi, M. Ferri, C. Mel-

chiorri, M.Montanari, G. Vassura, "A Robotic System for Visually Impaired People Based on Stereo-Vision and Force-Feedback", International Workshop on Medical Robots, Vienna, Austria, Oct. 1-2, 1996.

[2] C. Bonivento, A. Eusebi, C. Melchiorri, M. Montanari, G. Vassura, "WireMan: A Portable Wire Manipulator for Touch-Rendering of Bas-Relief Virtual Surfaces", 8th International Conference on Advanced Robotics, ICAR97, Monterey, CA, July 7-9, 1997.

[3] C. Melchiorri, M. Montanari, G. Vassura, "Control Strategies for a Defective, Wire-Based, Haptic Interface", International Conference on Intelligent Robots and Systems (IROS'97) Grenoble, F, September 8-12, 1997.

[4] L. Di Stefano, G. Paolinelli, "Stereo Research within the VIDET Project", Proceedings of the EUSIPCO'98 (IX European Signal Processing Conference), Rodi (G), Sept. 8-11 1998.

[5] G. E. Smid, Ka C. Cheok, J. L. Overholt, T. K. Tan, "Multi-CPU real-time simulation of vehicle dynamics", International Conference on Intelligent Systems, Paris (F), July 1998.

[6] G. E. Smid, Ka C. Cheok, "Human Integration in Simulation", 5th Workshop on Advanced Motion Control (AMC'98). Coimbra, June 1998.