

Improving the performance of Input Interfaces through Scaling and Human Motor Models

ABSTRACT

The performance of interfaces is affected by human factors, which vary from one person to another, and by the inherent characteristics of the various devices involved. A set of techniques has been studied in order to improve the efficiency and efficacy of input interface devices. These techniques are based on the modification of the motor scaling factor, a transformation similar to the known Control-Display ratio (CD ratio). Operation time, the accuracy of the task and user workload are the indicators used in this work. By means of models based on the various human motor behaviors, the improvement of such indicators has been demonstrated. Using some common input interface devices, a number of experiments have been carried out to evaluate the presented methodology. The results show that the overall performance of input interfaces is significantly improved by applying such methodology.

Keywords – Index of difficulty, Fitts's Law, Throughput, CD ratio.

CONTENTS

1. INTRODUCTION
2. HUMAN MOTOR BEHAVIOUR MODEL
 - 2.1. Navigation mode
 - 2.2. Ballistic mode
3. SCALING THE INTERFACE
 - 3.1. Scaling in navigation mode
 - 3.2. Scaling in ballistic mode
4. EXPERIMENTATION
 - 4.1. Conditioning the input interface
 - 4.2. Test preparation
 - 4.3. Parameter characterization
 - 4.4. Unidirectional connection test
 - 4.5. Multidirectional tracing test
5. CONCLUSIONS AND FURTHER WORK

1. INTRODUCTION

The use of adequate computer interfaces allows users to reach the highest levels of efficiency and efficacy along the execution of a task. Complex tasks, which are difficult to be executed by humans or machines separately, can be better performed by interfaces combining the best characteristics of both by means of proper techniques. In this scenario, users can execute these tasks in a more effective and ergonomic manner.

The use of Input interface devices has progressively grown, not only for human-computer interaction, but also in vehicle control, aeronautics, robotics and machine control. Devices like joysticks, mice, trackballs, and steering wheels connect human hands to computers or machines, transforming movement orders into the desired actions.

Ergonomics is a key factor in considering the communication between humans and machines. Current systems need the introduction of methods and elements so that the involved tasks are more efficient and effective. The interaction with machines through the current computer interfaces allows users to run accurate and efficient tasks that would be impossible to perform using conventional methods. Computer interfaces are especially relevant in tasks such as those performed in teleoperation, the handling of biological materials, or micromanipulation of new materials or microelectronic devices. However, these interfaces can also be applied to a wide range of daily computer-based tasks. Thus, any progress on improving human-computer interaction through interface devices has a significant impact in many applications and in many fields.

Dynamic movements for display on a computer screen can be executed through typical control devices, such as a mouse or a control stick. The process can be considered a closed-loop system, which incorporates the human hand, musculature, and eyes, as well as the brain. This paper seeks to demonstrate how the human performance of such computer-control tasks can be improved. To that end, the constraints intrinsic to human movement control are here taken into account. We define a dynamic scaling algorithm that improves the operation speed and accuracy in comparison with more common static sensitivity specifications.

The most widely accepted mathematical model of human movement is known as Fitts' Law (Fitts, 1954). Fitts' law relates the execution time of a rapid movement from a starting point to an object of width W and distance D .

$$MT = a + b \log_2 \left(\frac{D}{W} + 1 \right) = a + b ID \quad (1)$$

where a and b are constants, and the logarithmic term in Equation 1 is the difficulty index, ID (MacKenzie, 1992).

The velocity profiles of the trajectories executed by a human operator tend to be symmetric, a bell shape. They become more asymmetric as the maximum speed value decreases (MacKenzie, Marteniuk, Dugas, Liske, & Eickmeier, 1987). Considering this, we formulate a computational transformation between the control output and the movement of the cursor on the screen in order to improve the effectiveness of the human control.

Its use being so wide, Fitts' law has been incorporated into an ISO standard for ergonomics (ISO 9241-9:2000). Some variations of this model describe specific hand movement behaviors, like the continuous movement followed along a narrow path (Accot, & Zhai, 1997). In this case, the difficulty index is represented by the quotient between the traveled distance and the margin between the object's and path's widths. From these models, it can be seen that changing adequately the object's width or the traveled distance in fast or continuous movements through narrow paths, not only the execution time can be reduced, but also the user's workload. Some techniques base their performance on changes on the appearance or arrangements of objects in a computer desktop, thus facilitating pointing or dragging. Other techniques achieve this same goal by enlarging the size of the target (McGuffin, & Balakrishnan, 2005) (Zhai, Conversy, Beaudouin-Lafon, & Guiard, 2003), while others produce similar results by modifying the size of the cursor near the targets (Grossman, & Balakrishnan, 2005) (Chapuis, Labrune, & Pietriga, 2009). Some techniques reduce the distance to be run by approaching the target to the cursor (Baudisch et al., 2003).

The modification of the ratio between the displacement of a master device and the cursor movements on the screen (the named CD ratio) affects the execution time, in the same way that a change on the visualized object size or the distance to the objects does. Some mice drivers reduce this CD ratio by increasing the speed of the device, thus reducing the execution time for large displacements. An increase of the CD ratio improves the final positioning around the target (Accot, & Zhai, 2001). A change of the CD ratio in function of the distance to objects (Blanch, Guiard, & Beaudouin-Lafon, 2004) produces better results, in terms of operation time, than the use of a constant CD ratio. Furthermore, a modification of the acceleration factor (Radix, Robinson, & Nurse, 1999), the CD ratio or the direction of the cursor, has an influence on the execution time (Wobbrock, Fogarty, Liu, Kimuro, & Harada, 2009).

Due to the nature of the study of human movement behaviors, any study focused on improving the interaction through a given interface must follow an analysis supported on experimental data. One such study provides an objective evaluation of the effect of the parameters under consideration and the validity of the method. In this same sense, Fitts' Law relates the experimental results with a linear relationship between execution time and the logarithm of the ratio of distances and targets' width. This law has shown to be of great utility in the evaluation of interaction methods and interface devices.

Experimental analyses using Fitts' model have been applied to the new interaction technologies. Some recent studies focus on tilt-actuated mobile devices, (MacKenzie, &

Teather, 2012), tactile devices (Bi, Li, & Zhai, 2013), or virtual gloves (Shoemaker, Tsukitani, Kitamura, & Booth, 2012). An experimental analysis using psychomotor models, such as Fitts' Law, allows comparing the performance of different devices when some techniques are introduced like a scale change or the use of models of movement. In (Medryc, & MacKenzie, 2013) accuracy and execution time are evaluated comparing tilt-based and touch inputs on a mobile device. A similar evaluation is performed over a mobile device in a game scenario comparing the performance achieved using a *nano-stick*, foam buttons and *wii* controller inputs (Zaman, & MacKenzie, 2013).

A human-machine interface can take advantage of interaction methodologies based on human motor and psychomotor models. Objective and subjective data obtained from experimentation will allow improving the performance of the combined action of the human and the input interface. The present study shows a methodology oriented to improve the user input interface through scaling functions that affect the CD ratio. This methodology is based on a human behavior model aimed at predicting the user's intention. These predictions are based on the regular velocity profile of the hand movement and on the knowledge of the arrangement of the objects to interact with. In our previous work (Munoz, Casals, Frigola, & Amat, 2011) an introduction to the methodology is presented and validated with a set of experiments. The proposed techniques are compared with others based on either a constant function of the CD ratio or on a static function defined according to the arrangement of objects. However, the aim of this work is to evaluate the effect of the scaling methodology and the relative variation of the performance, in terms of efficiency (i.e. operating time), efficacy (accuracy and number of errors) and user's workload, comparing four different input interface devices.

The structure of the paper is as follows. The human motor model is described in Section II, the scaling CD ratio method in Section III, and finally, section IV depicts the experimentation procedure of the proposed methodology.

2. HUMAN MOTOR BEHAVIOR MODEL

A human motor model is used to improve the performance of the orders given by the user through the input interface device. This model is generated from the observation of the velocity profile along the execution of slow and fast movements. In rapid approaching to objects, the velocity profile is composed of an initial symmetric bell-shaped impulse followed by a set of minor corrective pulses around the target (Woodworth 1899, Meyer, Smith, Kornblum, Abrams, & Wright, 1988-1990). The width of the initial impulse is very regular, from person to person. Thus, it can be modeled through experimentation. This behavior is produced in rapid voluntary hand movements giving a ballistic-like response. That is, during rapid movements, humans do not have an accurate control over the whole movement; their own experience determines its performance. However, for a given range of speed values the user can control the movement with certain accuracy. In continuous and controlled movements the deviation of a traced line with respect to an imaginary line can also be modeled.

Therefore, we can differentiate two essential modes: a *ballistic* mode for fast movements and a *navigation* mode for controlled precise movements. Although other modes could be used in the study of the velocity profile of the hand movement (e.g. oscillatory), the two modes described above include most of the cases that occur in a human - computer interaction. These two modes can be distinguished because in the ballistic mode a speed threshold, named V_U , is surpassed, while the speed in the navigation mode is always below this threshold. When a picking task (e.g. reaching an icon over the desktop) performed in a ballistic mode fails after the first impulsive movement, a sequence of lower impulsive movements is applied around the target until it is reached. These lower impulsive movements, which constitute a *corrective* phase, never surpass V_U and the successive maxima decrease as they approach the target. Other phases of the movement can be identified, like a *rest* state, of no movement, or an *acceleration* state, a change from the rest state to the maximum speed. Then, based on this human motor model, it is necessary to register the velocity profile during the hand movement to determine its phase or state. A finite state machine (FSM) is used to identify these states.

Figure 1 shows a pointing task velocity profile, which is composed of an initial impulse and a set of corrective impulses until the target is reached. The same sequence is shown in Figure 2 for a navigation movement (e.g. navigation over the menus on a windows interface), while Figure 3 shows the finite state machine that identifies the state of the movement depending on the previous state and the value of the instant velocity profile. For simplicity the states are named *R* (rest), *A* (acceleration), *B* (ballistic), *C* (corrective) and *N* (navigation). A change of state can happen when the instant velocity reaches either the threshold V_U , or the maximum V_{MAX} , or when $v = 0$.

If V_{MAX} is reached and $V_{MAX} > V_U$, then the movement is identified as a ballistic impulse. Otherwise it corresponds to a corrective impulse. The variable *b* shown in Figure 3 is activated on ballistic mode and continues with the corrective states until the object is

reached (distance to the object $r = 0$). Figure 4 summarizes all possible combinations of the future states in function of the present state, the instantaneous speed, V_{MAX} and b .

(Figure 1 about here)

(Figure 2 about here)

(Figure 3 about here)

(Figure 4 about here)

2.1. Navigation mode

During the navigation mode (state N), the user controls the movement and adjusts the speed according to the desired precision. This precision depends on the margin of movements between the path and the object to be moved. The maximum speed that the user can achieve in order to perform a task successfully is the threshold speed V_U , which depends on the *throughput* TP required for the task. This TP value can be obtained from norm (ISO 9241-9:2000) for tracing or dragging tasks:

$$TP = \frac{D}{S_{AV}T} \quad (2)$$

where D is the distance covered along a task lasting a total time T , and S_{AV} the average deviation around a central line. The ISO norm defines the Throughput as the quotient between the effective index of difficulty (IDE) and the movement time: $TP = IDE/MT$. For tracing tasks $IDE = D/We$. For selection, pointing or dragging tasks: $IDE = \log_2(1 + D/We)$. We is the target width of the displayed target.

The average speed during the task can be expressed as $V_{AV} = D/T$, then from Equation 2 it results:

$$V_{AV} = TP \cdot S_{AV} \quad (3)$$

Equation 3 represents the maximum average speed for the tracing task with an error S_{AV} . If ω_p is the precision required for the movement, then:

$$V_{AV} \leq V_U = K_T TP \omega_p \quad (4)$$

And the threshold speed V_U is defined from TP for a given task with precision ω_p , being K_T a constant of proportionality, the value of which depends on each user.

2.2. Ballistic mode

When V_U is surpassed, the movement is considered ballistic, since the user cannot control its smooth continuity like in the navigation mode. During a ballistic movement, the initial impulse becomes a large displacement, but executed in a short time compared with the total time required to reach the target. For this reason, an attempt to achieve the goal during the first impulse will strongly reduce the execution time. It can be shown (Meyer, Smith, Kornblum, Abrams, & Wright, 1988) that, after the initial impulse, the standard deviation S_1 of the displacement D_1 is directly proportional to the average speed during the initial impulse, with a constant of proportionality K_1 that depends on each user:

$$S_1 = K_1 V_1 = K_1 \frac{D_1}{T_1} \quad (5)$$

The correspondence between speed and variability is non-linear at very low speeds or very high speeds on. This has also been noted in other studies as (Liao, Jagacinski, & Greenberg, 1997). Nevertheless, the speed values obtained from the difficulty indexes that the ISO Norm suggests show linearity. For an object of width W , the probability of reaching the object after the impulse with an error ε_1 , considering a normal distribution with zero norm and deviation S_1 is:

$$P = \Pr \left[-\frac{W}{2} \leq \varepsilon_1 \leq \frac{W}{2} \right] = \frac{1}{S_1^2 \sqrt{2\pi}} \int_{-\frac{W}{2}}^{\frac{W}{2}} e^{-\frac{\varepsilon_1^2}{2S_1^2}} d\varepsilon_1 \quad (6)$$

Then, knowing the deviation S_1 , the error ε_1 can be obtained for a given object of width W and probability P of success (e.g. 95%). The lower the probability P to reach the object, the higher the error around the center of the target. When the user wants to reach an object, this is identified as a target if the error between the initial distance D and the total distance traveled D_1 during the initial pulse is lower than ε_1 for a given probability P according to Equation 6. Due to the fact that at the beginning of the movement the distance D_1 is not known, this distance could be predicted during the pulse as it will be described in section 3.2, and the movement could be modified through the scale in order to reach the target. If the prediction does not identify any object as a target, no action over the scale will be done, since it is assumed that the user is making movements with no intention on a specific target.

Figure 5 shows an example of a picking task where three possible movements from a starting position falls in three positions around an object; position a is far from the target, then it is not considered as a target; positions b and c are near the object with an error which is small enough to consider the object as a target.

(Figure 5 about here)

3. SCALING THE INTERFACE

The CD ratio relates the actions performed by the user's hand to the cursor movements over the visual display. The concept of scale used in this paper is the inverse of the CD ratio. Unlike some studies, which focus mainly on the use of a predefined scale (La Viola 1997) (Accot, & Zhai, 2001), the method used in this research performs a dynamic scaling. According to the models described in the previous section, the scale factor changes in function of the distance to the object and the velocity of the hand movement. This is possible provided that the position of the objects of interest is known. An example of application is the pointing to the icons on a desktop and buttons on a menu bar, which are located at known positions and have a known width. In this case, the object of interest becomes the target in function of the direction and magnitude of the initial impulse according to Equation 6 for picking tasks. Then, adapting the interface by means of a scale function allows reducing the operation time, the number of corrections and, finally, the total user's workload. On the other hand, Equation 4 must be satisfied around the objects to be manipulated by dragging or navigating through narrow menus.

The finite state machine described above is used to identify each operation mode. In order to achieve the *throughput* of the task, a scaling variable named s_M is introduced between the velocity movement command of the user (v_i) and the velocity of the cursor on the visual display (v_o):

$$v_o = s_M v_i \quad (7)$$

The conditions to be accomplished in order to apply the scaling methodology and the human motor model can be summarized as follows:

- The position of the objects must be known as well as their relative distances to the cursor.
- The objects should be classifiable into those that can be reached by ballistic movements (e.g. buttons, icons...), and those that are navigated through (e.g. dropdown menus, dragging, editing actions...).
- The interface requires a calibration process in order to characterize the parameters of the scaling functions.
- The actions performed with the input interface are continuously recorded. The direction and the speed determine which of the objects becomes the target.

Normally, the interface works with the settings configured by the user (usually constant scaling with, maybe, pointing acceleration). In function of the direction and the speed of the movement carried out by the user through the input interface, the scaling function helps the user to reach the targets or makes the tracing tasks faster and more precise.

The finite state machine determines the instants in which a change of state occurs in order to decide when to apply the proper scaling functions. During the states of rest,

acceleration or corrective, the scale remains constant. In the ballistic state, the dynamic scaling is applied when, during the movement, a potential ballistic target is identified. In navigation state, the dynamic scaling is applied when movements are performed over navigation targets.

This methodology is to be applied when the information of targets (objects, singular points, etc.) can be extracted from their context, either because the position of the potential targets is known or because it can be obtained from other sensors. When the position of the objects is known, as it is the case with the icons on a monitor, the analysis of movement, pointing, direction... can help to foresee the desired target or its distance. In other applications, vision or other sensors supply this information (e.g. holes in an assembly task). In some cases, the process could be automatized, but this work focuses on those situations in which it is the user who selects the targets. Once the direction of the movement and the potential target are identified, the scaling process allows the user to perform the movements in a fluent way, without jumps of the cursor or losing the control temporarily. The interface becomes not only more efficient and effective but also more ergonomic and transparent.

3.1. Scaling in navigation mode

During a navigation task in which the movement is below the threshold velocity V_U , a constant scaling (k_M) relates the velocities in Equation 7. However, when the velocity surpasses V_U , the scale s_M varies, so as to allow the user to perform the task under control according to Equation 4. As seen in Figure 6, a saturation technique that limits the velocity to V_U can be applied as follows:

$$|v_o| < V_U \Rightarrow v_o = k_M v_i \Rightarrow s_M = k_M$$

$$|v_o| \geq V_U \Rightarrow v_o = V_U \Rightarrow s_M = \frac{V_U}{v_i} \quad v_i \neq 0$$

Normally, k_M takes values between 1 and 2, which is a common scale used in conventional input interfaces (e.g. a mouse).

(Figure 6 about here)

3.2. Scaling in ballistic mode

During ballistic mode, Equation 6 determines, with a certain probability P , whether an object can be reached during the initial impulse or not. Thanks to the symmetric behavior of the velocity profile during the initial impulse, the displacement and time lapsed can be predicted. Therefore, the movement can be corrected using the scaling in order to reach the object at the first impulse. Once the maximum velocity is reached and

due to the natural symmetry of the impulse, the estimated displacement x_f during the impulse is:

$$x_f = 2x_p$$

where x_p is the occurred displacement until the maximum speed is reached, Figure 7. Therefore, if the estimated displacement falls within the interval corresponding to the initial distance D to the target and ε_1 , being ε_1 the error around the target, a correction will be needed.

$$x_f \in D \pm \varepsilon_1 \Rightarrow \text{correction}$$

(Figure 7 about here)

The correction performed by the scaling during the deceleration phase is based on an estimation of the time t_e with which the velocity impulse reaches the zero value. For this estimation, functions $v_{ie}(t)$ and $v_{oe}(t)$ are used to approximate the velocity profile at each instant in the input and output interfaces. Then:

$$t_e \rightarrow v_e(t_e) = 0 \quad (8)$$

$$x_e(t_e) = \int_0^{t_e} v_{ie}(t) dt \quad (9)$$

$$v_{oe}(t) = s_M v_{ie}(t) \quad (10)$$

$$y_e(t_e) = \int_0^{t_e} v_{oe}(t) dt \quad (11)$$

where y_e is the estimated displacement of the cursor on the screen, and x_e the estimated displacement on the input interface. If y and x are respectively the actual displacement on the screen and on the input interface, and v_o and v_i the corresponding actual velocities, we can solve Equations 8 to 11 with $v_{ie}(t)$ and $v_{oe}(t)$ as linear functions of t , Figure 8. Here the impulse velocity is approximated to a linear function, the one used in the implementation of the experiments. No significant difference in performance is observed when adjusting the impulse to a higher order function.

(Figure 8 about here)

The Equation that describes the estimated linear velocity in the input interface v_{ie} at each instant t is:

$$v_{ie}(t) = a_2 \cdot t + a_1 \quad (12)$$

$$a_2 = \frac{dv_i}{dt} \quad (13)$$

Since the slope a_2 in Equation 12 corresponds to the actual acceleration dv_i/dt , the offset a_1 can be obtained from Equations 12 and 13 for a given t and the actual v_i . Then:

$$a_1 = v_i - a_2 \cdot t \quad (14)$$

The estimated instant at which the speed is zero is determined by $v_i = 0$ in Equation 16. Then:

$$t_e = -\frac{a_1}{a_2}, a_2 \neq 0 \quad (15)$$

In order to obtain the input estimated total movement x_e , by integrating Equation 12 we can obtain the following Equation:

$$x_e(t) = \frac{1}{2}a_2t^2 + a_1t + a_0 \quad (16)$$

At each instant t with a known actual position x , the parameter a_0 will be:

$$a_0 = x - \frac{1}{2}a_2t^2 - a_1t \quad (17)$$

then, for $t = t_e$ in Equation 16:

$$x_e = \frac{1}{2}a_2t_e^2 + a_1t_e + a_0$$

which, from Equations 14, 15 and 17, can be reduced to:

$$x_e = x + \frac{(t_e - t) \cdot v_i}{2} \quad (18)$$

At the output interface, being the final position value y_e , then, for a given t and v_o , and operating as in Equation 18, we obtain:

$$y_e = y + \frac{(t_e - t) \cdot v_o}{2} \quad (19)$$

By combining Equations 18 and 19, and knowing that the relationship between the velocities in the input and output interfaces is the motor scale, Equation 7, we obtain:

$$s_M = \frac{v_o}{v_i} = \frac{y_e - y}{x_e - x}, x_e \neq x$$

4. EXPERIMENTATION

This study evaluates how the performance of an interface is affected by the use of the described dynamic scaling. Then, it is compared with a conventional constant scaling with acceleration (the commonly used configuration on a windows environment) and a position dependent progressive scaling, named here static scaling, (some of the techniques in the literature use this form of variation of the CD ratio). The efficiency and efficacy of the interfaces for a set of tasks (i.e. operating time and number of faults) as well as the user workload are evaluated. Two experiments are chosen from those proposed in the standard ISO 9241-9, the *unidirectional connecting test* and the *multidirectional tracing test*, which evaluate ballistic rapid movements and navigation precise movements. The experiments were carried out on 25 voluntary subjects, college students of both sexes, all of them right-handed.

At the beginning of both experiments, each participant performs a set of random trials in order to get used to the task, reduce the effect of practice and allow characterizing the parameters of the model (V_U , K_1 , ϵ_1). At the end of the experiments each participant is asked to fill in a subjective evaluation in order to assess the workload and the way they perceive the execution of the task, with or without scaling variation. This test contemplates the participant's perception of the workload, precision, speed, frustration and attentiveness, which is graded from 1-low to 5-high for the first scaling method. Afterwards, the other scaling methods are evaluated as well, and compared to the previous ones. As the ISO norm suggests, the *workload* is evaluated in terms of subject's fatigue (physical and/or visual); *precision*, to which extent it helps the user to accurately reach the targets; *speed*, understood as to which extent the task is executed in a faster way; *frustration*, understood as the degree of frustration as a consequence of failures; and *attentiveness*, understood as the degree of attention required to perform a task successfully. All the participants know how to interpret these parameters before the experimentation starts. Each parameter is evaluated from 1(low) to 5(high) for the first test and, in a comparative way, the following tests from 0(lower) to 5(higher), and 3(no change), see Figure 9.

The order of the scaling methods is randomly set among the participants. The interface runs over an Intel i5 PC with a 17" and 1280×1024 pixel resolution screen. The data were analyzed through balanced two way analysis of variance (ANOVA) within subjects with random factors for each method, using the Minitab[®] R17.

The aim is to compare four input interface devices and determine how dynamic scaling affects their performance. The devices used are shown in Figure 10: a Logitech G9 laser mouse, a Logitech TrackMan Marble trackball, a 3Dconnexion SpaceNavigator static joystick and a Trust GM-2500 conventional Joystick. The tests used for the evaluation of the ballistic and navigation movements are described below.

(Figure 9 about here)

(Figure 10 about here)

4.1. Conditioning the input interface

The commands from the input interface device must be converted into two-dimensional movements on the output interface, the screen. Due to the different nature of the four input interfaces, it is necessary to incorporate a function that conditions the input movements before transforming them into the output interface movements. The conditioned input command velocity v_i , is scaled by applying the scaling methodology and the cursor movement, with velocity v_o , is obtained in the output interface following Equation 7, Figure 11.

(Figure 11 about here)

Each input interface has particular working characteristics. The movement characteristics can be classified in function of the way they are normally used to input commands; incremental or absolute and angular or linear. Other characteristics that affect the movement are the range and the number of degrees of freedom (DOF). The characteristics that affect the accuracy are resolution, hysteresis and linearity. Joysticks usually have springs to maintain the stick centered. This introduces hysteresis in such position, which affects the accuracy achieved with low speed inputs. Figure 12 summarizes the characteristics of the four input interfaces used in the experiments.

(Figure 12 about here)

Figure 13 shows how each interface must be conditioned in order to convert the native physical movements into a linear velocity command. The Mouse and the Trackball are commonly used as pointing devices; working in an incremental way, they need only a scale to convert the movement commands into velocity commands. The Joystick presents nonlinearity and hysteresis that should be compensated, and its angular and absolute behavior must be converted into linear incremental commands. The Joystick 3D is highly linear and has low hysteresis compared to a conventional Joystick, but its very short range of movement makes it static or quasi-static, which allows working from position to velocity or from force to velocity.

The ISO norm suggests that the TP of the hand must reach at least 2bit/s for the hand (wrist), and 3 bit/s for the fingers. Using Fitts' Equation 1 for an object of width 2mm located at a distance of 350mm (the width of a 17" screen monitor) we obtain the resulting speed:

$$ID = \log_2 \left(1 + \frac{D}{W} \right) = \log_2 \left(1 + \frac{350}{2} \right) = 7.5 \text{ bits}$$

$$TP = \frac{ID}{MT} \Rightarrow V = \frac{D}{MT} = TP \frac{D}{ID}$$

$$TP = 2 \text{ bits/s} \Rightarrow V = 94 \text{ mm/s}$$

$$TP = 3 \text{ bits/s} \Rightarrow V = 141 \text{ mm/s}$$

The Mouse and the Joystick are hand-wrist actuated, while the Trackball and the Joystick 3D are finger actuated. Taking into account the operation ranges of each interface and the value of speed V computed above, the conversion factor C between the physical input command and the input velocity v_i can be computed:

$$C = \frac{V}{range} fs$$

being fs a safety factor (here set to 1.25, which allows a 25% of speed margin with respect to the available range).

Since the Mouse has a theoretically infinite range, it has been calibrated so that its movement and that of the cursor match. Then $C = 1$ being the CD ratio fixed to 1. On this basis, any of the three scaling methods studied can be applied. The Mouse and Trackball relate the input velocity to the output velocity through scaling (position control). The Joystick and Joystick 3D relate the input position to the output velocity through scaling (velocity control). Although some studies demonstrate that position control has a better performance than velocity control for stationary target acquisition (Jagacinski, Repperger, Ward & Moran, 1980), joysticks are velocity-controlled because the cursor must remain still when they are at their central position.

Figure 14 shows the parameter C that conditions each interface. In this study none of the hysteresis effects and non-linearities has been compensated, since its focus is on the study of the influence of scaling functions upon the performance.

(Figure 13 about here)

(Figure 14 about here)

4.2. Test preparation

Before the execution of the tests, the participants are submitted to a set of random practice trials followed by calibration trials. The practice trials allow the user to get familiar with the interface and the tasks to be performed. Therefore, the variability of the results is reduced due to the practice effect. The calibration trials provide the parameters that characterize the dynamic scale functions.

The initial practice trials consist of 25 trials of the unidirectional connection test, where targets appear at random distances D and with random widths W . This results in difficulty indexes between 1 and 6 bits, as they appear in the final test. Afterwards, the participant performs 4 trials of the multidirectional tracing test. The combinations of diameter of circumferences and width of the path are the same as in the final test. A Duncan's Range test verifies that this number of trials is enough to ensure that the learning effects are stabilized. After a short pause, the same sequence of trials is repeated for calibration. In this case, the goal is to capture the parameters of the user that characterize the dynamic scaling functions.

The user is encouraged to perform the trials as quickly as possible and to make as few errors as possible. This test is repeated for the four input interface devices.

4.3. Parameters characterization

All the parameters presented in this work have been characterized through experiments. The finite state machine presented in section 2 has not been used because only one known target is presented on the screen at a time.

During the calibration trials, the input movement from the interface is recorded every 20 ms. The measurement of the displacement D_1 and the time T_1 after the initial impulse in the unidirectional connection test provides the deviation S_1 , according to Equation 5. In the multidirectional tracing test, the total displacement D , the total execution time T without faults (without contact with the circumferences) and the average deviation S_{AV} around a central line are recorded. These values are used to obtain the TP and the maximum average speed V_{AV} to execute the task successfully, according to Equations 2 and 3. The constant K_T is estimated after repeating the test, for the different combinations of diameter of circumference and width of the path. From K_T , the threshold speed V_U is obtained in function of the TP and the required precision ω_p , according to Equation 4.

The constant scaling factor k_M is fixed to 2 (Munoz, 2012), a trade-off between accuracy movements near the target and reduced displacements during the movement toward the target. The pointer acceleration factor is only considered in the constant scaling. The interface driver used in the experiments provide three levels of acceleration (low,

moderate and high). Although the literature is not conclusive on the effect of each of these levels on the interface performance, in general, users have a preference for the moderate level. The majority of participants in this experiment also showed a preference for this level. Quantitatively speaking, this level of acceleration means that the speed increases a 10% every 10ms.

The static scaling is a function $s_M(r)$ of the distance r to the objects. This function takes the shape shown in Figure 15. The low scale s_{min} near the objects ($r < r_{min}$) allows reducing the corrective movements around the target; and the largest scale s_{max} far from the objects ($r > r_{max}$), allows reducing the traveled distance. In a previous work (Munoz, 2012), the best performance was reached with the values: $r_{min} = 5$, $r_{max} = 50$, $s_{min} = 1$, $s_{max} = 8$. The values of distances r are measured in mm, the scales s are dimensionless.

(Figure 15 about here)

4.4. Unidirectional connection test

The unidirectional connection test (known as the Fitts' experiment) consists in moving the cursor towards a vertical rectangle as quickly as possible, as shown in Figure 16. The width W of the rectangle and the distance D to the cursor vary along the test, giving an index of difficulty between 1 and 6 bits, as the ISO norm suggests.

(Figure 16 about here)

Every participant performs the test with each device, combining three distances D and two object widths W for a total of 6 difficulty indexes ID . The number of samples in this test results from the combinations of 3 scaling modes \times 6 ID \times 5 reps \times 25 participants \times 4 devices = 9000 samples. Figure 17 shows the D and W combinations used for every ID and the exact value of ID computed according to Equation 1. The execution time and the Throughput are measured in all cases. Figure 18 shows the average and standard deviation for all the data in every method and device. The results obtained after the analysis of variance are shown in Figure 20 with the values of the F -test and the probability p of null hypothesis. The results of the average execution time MT and Throughput TP , with their 95% confidence intervals, are shown in Figure 19.

(Figure 17 about here)

As shown in Figure 20, there is a significant difference, in the average execution time and Throughput, among the three scaling methods in all the devices. The average Throughput has been calculated for both static and constant scaling. The former has produced higher results in all cases, the increase ranging from an 18% for the Mouse to a 46% for the Joystick 3D. In the dynamic scaling method the range of improvement

over the static scaling is around 32% on the Trackball and 95% on the Joystick 3D, as seen in Figure 19 and Figure 18.

In the constant scaling mode, the device showing the largest *TP* is the Mouse, with an average of 5 bit/s, being the conventional Joystick the one that shows the lower *TP*, with 2 bit/s. By means of the scaling methods the Joystick's *TP* can reach similar values to those obtained by the Mouse at constant scaling. In fact, certain combinations of *ID* can be observed, in which some devices behave better than others when applying the scaling functions.

Interaction terms appear in some *ID**Method combinations, as seen in Figure 20. There are two main causes of interaction, Figure 19. Firstly, the lower *ID* does not show significant difference among the three methods. That is to say, when the task is very "easy" to perform, the aiding method does not make any difference on the execution of the task. Secondly, the Index of difficulty *ID*2 produces higher *MT* than the *MT* obtained from the Index of difficulty *ID*3. A similar effect occurs with *TP*. Although this result conflicts with Fitts' law, it appears that the smaller target ($W = 2\text{mm}$) in the *ID*2 task penalizes more the execution time than the larger distance ($D = 100\text{mm}$) in the *ID*3 task, despite having similar *ID*, see Figure 17. The presence of the dynamic scaling method reduces drastically the number of corrections that appear especially on the smaller target.

Figure 21 shows the result of the test of subjective evaluation, which compares the perception of the scaling methods over a constant scaling. According to this quiz, the participants perceived that the dynamic scaling allows executing the task with lower workload, higher precision, higher speed, lower levels of frustration and requiring lower attentiveness.

(Figure 18 about here)

(Figure 19 about here)

(Figure 20 about here)

(Figure 21 about here)

4.5. Multidirectional Tracing test

The multidirectional tracing test consists in dragging an object, as quickly as possible, through a circular path composed of two concentric circumferences, and without touching them, Figure 22. The object width W and the diameter between the circumferences D are modified along the test giving different indexes of difficulty, as the ISO norm suggests. In this test, the method of dynamic scaling is compared with

respect to a conventional constant scaling with acceleration. There is no use applying the static scaling method here since the cursor navigates close to the targets.

(Figure 22 about here)

The 25 participants performed the test with the four input interface devices over combinations of the object's sizes and the diameters of the circumference, giving four *ID*. The test is composed of two diameters *D* (40 and 100mm) and two object widths (4 and 7mm). The width of the path formed by the two circumferences is 10mm, allowing a margin of movement of 6 and 3mm. The resulting number of samples in this test is: 2 scaling methods \times 4 *ID* \times 5 reps \times 25 participants \times 4 devices = 4000 samples. Figure 23 shows the *D* and *W* combinations used for every *ID* and the exact value of *ID*. In a tracing task the *ID* is computed through the quotient between the length of the path πD , and the margin of movement between the width of the path *W* and the width of the target *K*. Then $ID = \pi D / (W - K)$. Being *D* the diameter of the circular path, with *K*=7mm the width of the target. The execution time, the *TP*, and the number of faults (number of times that the object touches the path, causing the participant to begin the trial) are measured in all cases. Figure 24 shows the average and standard deviation for all the data in each method and device. The results after the analysis of variance are shown in Figure 25 with the values of the *F*-test and the probability *p* of null hypothesis. The results of the average execution time *MT*, Throughput *TP* and number of faults, with their 95% confidence intervals are shown in Figure 26.

(Figure 23 about here)

As shown in Figure 25, there is a significant difference in the average execution time, in the average Throughput and in the average number of faults, between the two scaling methods in all the devices. Interaction terms appear in some *ID**Method combinations, see Figure 25. The narrow path task has more impact over *MT*, *TP* and number of faults, than the larger path task, see Figure 26. On the other hand, the presence of the scaling method benefits more the tasks with the narrower path.

Dynamic scaling allows a reduction in the execution time, therefore increasing *TP* in all cases, from an average around 50% for the Joystick 3D up to a 98% on the Trackball, and reducing almost completely the number of faults, see Figure 24. It is important to highlight the improvement obtained on the conventional Joystick. This device is characterized, on the one hand, by a difficult control due to the presence of hysteresis on the neutral position (when the Joystick rests at its central and resting position) and, on the other hand, by the nonlinear force required in this position. The dynamic scaling method produces an increase of around a 70% in the *TP* and an important reduction in the number of faults.

(Figure 24 about here)

(Figure 25 about here)

(Figure 26 about here)

The particular operation of the Trackball makes the tracing of circular trajectories difficult due to the continuous change in finger movements to modify the advance direction along the circumference. Applying a constant scaling, this device's performance is similar to that of the conventional Joystick, but the *TP* doubles up when applying the dynamic scaling.

The Joystick 3D, with an isometric behavior characterized by a very low range of movements, provides the highest performance at constant scaling, making it suitable for navigation tasks. In dynamic scaling the task performance becomes even more fluid and effective since there is a reduction in the execution time and in the number of faults.

Figure 27 shows the results of the test of subjective evaluation, which compares the perception of dynamic scaling over a constant scaling. This is the case with the unidirectional connection test. Here, the quiz also shows that the participants have a better perception of the dynamic scaling method because it allows executing the task with lower workload, with higher precision, at a higher speed and with lower levels of frustration and attentiveness with respect to the constant scaling.

Finally, Figure 28 shows the average *TP* for all the data obtained for each device applying the dynamic scaling method and comparing them with those obtained with the constant scaling. All the devices show a better performance using the dynamic scaling method, as the above results show. Besides, the average throughput of some devices with poor performance at constant scale (e.g. the Joystick) can reach the performance of the Mouse obtained at a constant scale when applying a dynamic scale. This is remarkable because, with the proposed method, some devices could become suitable for tasks that otherwise would not perform with the required level of efficiency, effectiveness and ergonomics.

(Figure 27 about here)

(Figure 28 about here)

5. CONCLUSIONS AND FURTHER WORK

For general purpose applications, a constant scaling method applied to the input interface —with some aids such as the acceleration provided on conventional computers—, is useful. The acceleration aid reduces the total displacement of the input interface when the cursor is far from the objects. The static scaling varies the CD ratio in function of the distance to the targets. It has been demonstrated that this scaling method improves the execution of tasks reducing operation time and the number of errors. However, this is only possible when certain knowledge about the layout of the objects in the computer interface is available. The introduction of the motor behavior model allows advancing a step further, thus making it easier for the user to execute tasks at maximum throughput with minimum workload and attentiveness for all the input interfaces analyzed. Another conclusion refers to the way in which the performance of a determined input interface device changes with the introduction of the scaling method for certain tasks. It has been demonstrated that whereas a determined device shows a poor performance when operating with constant scaling, other devices perform successfully. Introducing dynamic scaling instead, the performance of such a device (e.g. the joystick) reaches a fluent operating mode. From the users' point of view, the continuous scaling allows the feeling of a natural movement of the cursor. Participants in the experiment adapted quickly to the assisted interface.

The next step will be the application of these techniques in teleoperation systems. Further experimentation in this motor behavior and dynamic scaling techniques is being performed by applying them on tactile devices.

REFERENCES

- Accot, J., Zhai, S. (1997). Beyond Fitts' Law: Models for Trajectory-Based HCI Tasks. *Proceedings of the CHI 1997 Conference on Human Factors in Computing Systems*. New York: ACM.
- Accot, J., & Zhai, S. (2001). Scale effects in steering law tasks. *Proceedings of the CHI 2001 Conference on Human Factors in Computing Systems*. New York: ACM.
- Baudisch, P. et al. (2003). Drag-and-Pop and Drag-and-Pick: techniques for accessing remote screen content on touch- and pen-operated systems. *Proceedings of INTERACT 2003*. Laxenburg: IFIP.
- Bi, X., Li, Y., & Zhai, Sh. (2013). FFitts law: modeling finger touch with Fitts' law. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York: ACM.
- Blanch, R., Guiard, Y., & Beaudouin-Lafon, M. (2004). Semantic Pointing: Improving Target Acquisition with Control-Display Ratio Adaptation. *Proceedings of the CHI 2004 Conference on Human Factors in Computing Systems*. New York: ACM.

- Chapuis, O., Labrune, J.B., & Pietriga, E. (2009). Dynaspot: Speed-Dependent Area Cursor. *Proceedings of the CHI 2009 Conference on Human Factors in Computing Systems*. New York: ACM.
- Grossman, T., & Balakrishnan, R. (2005). The Bubble Cursor: Enhancing Target Acquisition by Dynamic Resizing of the Cursor's Activation Area. *Proceedings of the CHI 2005 Conference on Human Factors in Computing Systems*. New York: ACM.
- Fitts, P.M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47, 381-391.
- ISO 9241-9:2000. Ergonomic requirements for office work with visual display terminals, Part 9: Requirements for non-keyboard input devices.
- Jagacinski, R.J., Repperger, D.W., Ward, S.L., & Moran, M.S. (1980). A test of Fitts' Law with moving targets. *Human Factors*, 22:2, 225-233.
- La Viola, J. (1997). Analysis of mouse movement time based on varying control-to-display ratios using Fitts' law. (Technical Report CS-97-17). Brown University, Providence, RI.
- Liao, M., Jagacinski, R.J., & Greenberg, N. (1997). Quantifying the performance limitations of older and younger adults in a target acquisition task. *Journal of Experimental Psychology: Human Perception and Performance*, 23, 1644-1664.
- McGuffin, M.J., & Balakrishnan, R. (2005). Fitts' law and expanding targets: experimental studies and designs for user interfaces. *ACM Transactions on Computer-Human Interaction*, 12:4, 388-422.
- MacKenzie, C.L., Marteniuk, R.G., Dugas, C., Liske, D., & Eickmeier, B. (1987). Three-dimensional movement trajectories in Fitts' task: Implications for control. *The Quarterly Journal of Experimental Psychology, Section A*, 39:4, 629-647.
- MacKenzie, I.S. (1992). Fitts' law as a research and design tool in human-computer interaction. *Human-Computer Interaction*, 7, 91-139.
- Meyer, D.E., Smith, J.E., Kornblum, S., Abrams, R.A., & Wright, C.E. (1988). Optimality in human motor performance: ideal control of rapid aimed movements. *Psychological Review*, 95, 340-370.
- Meyer, D.E., Smith, J.E., Kornblum, S., Abrams, R.A., & Wright, C.E. (1990). Speed-accuracy tradeoffs in aimed movements: Toward a theory of rapid voluntary action. *Attention and performance XIII*. Hague: IASAP.
- MacKenzie, I. S., & Teather, & R. J. (2012). FittsTilt: The application of Fitts' law to tilt-based interaction. *Proceedings of the Nordic Conference on Human-Computer Interaction – NordiCHI 2012*, New York: ACM.
- Medryk, S., & MacKenzie, I. S. (2013). A comparison of accelerometer and touch-based input for mobile gaming. *Proceedings of the International Conference on Multimedia and Human-Computer Interaction - MHCI 2013*. Ottawa: International ASET Inc.
- Munoz, L.M. (2012). Transformación escalar de la interfaz de operador en teleoperación asistida. Unpublished doctoral dissertation. Universitat Politècnica de Catalunya.

- Munoz, L.M., Casals, A., Frigola, M., & Amat, J. (2011). Motor-Model-Based Dynamic Scaling in Human-Computer Interfaces. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 41:2, 435-447.
- Radix, C., Robinson, P., & Nurse, P. (1999). Extension of Fitts' law to modelling motion performance in man-machine interfaces. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, 29:2, 205-209.
- Shoemaker, G., Tsukitani, T., Kitamura, Y., & Booth, K. (2012). Two-Part Models Capture the Impact of Gain on Pointing Performance. *ACM Transactions on Computer-Human Interaction*, 19:4, 28.
- Wobbrock, J.O., Fogarty, J., Liu, S.S., Kimuro, S., & Harada, S. (2009). The Angle Mouse: Target-Agnostic Dynamic Gain Adjustment Based on Angular Deviation. *Proceedings of the CHI 2009 Conference on Human Factors in Computing Systems*. New York: ACM.
- Woodworth, R. S. (1899). The accuracy of voluntary movement. *Psychological Review*, 3:13, 1-119.
- Zaman, S., & MacKenzie, I. S. (2013). Evaluation of nano-stick, foam buttons, and other input methods for gameplay on touchscreen phones. *Proceedings of the International Conference on Multimedia and Human-Computer Interaction - MHCI 2013*. Ottawa: International ASET Inc.
- Zhai, S., Conversy, S., Beaudouin-Lafon, M., & Guiard, Y. (2003). Human on-line response to target expansion. *Proceedings of the CHI 2003 Conference on Human Factors in Computing Systems*. New York: ACM.

Figure 1. Response of a pointing task composed of an initial ballistic impulse followed by a set of corrective pulses around the target. a) Velocity profile of impulsive movements b) Chronogram with the states involved (*R*, *A*, *B*, *C*).

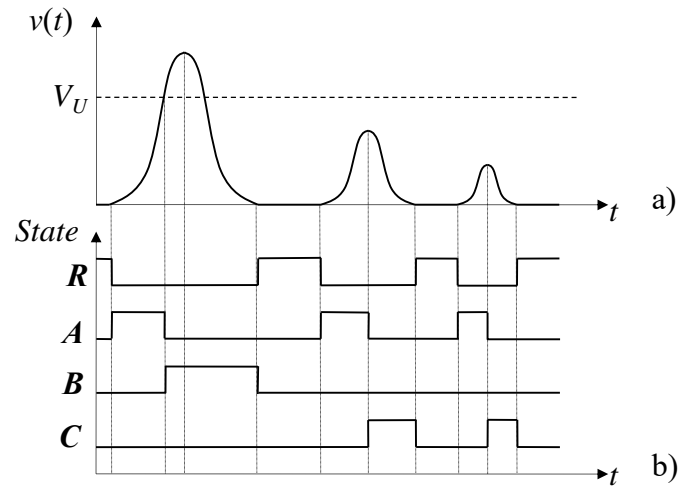


Figure 2. Schema of the navigation movement (typical drag or menu navigation).
a) Velocity profile. b) Chronogram with the states involved (R , A , N).

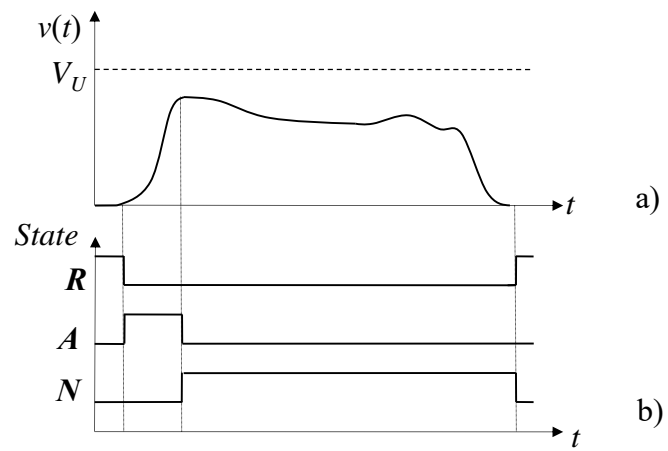


Figure 3. Finite state machine with the states and transitions involved, in function of the instant value of the velocity profile and distance to the object. The dashed line separates the parts of the state machine that correspond to a navigation task (top) and a ballistic task (bottom).

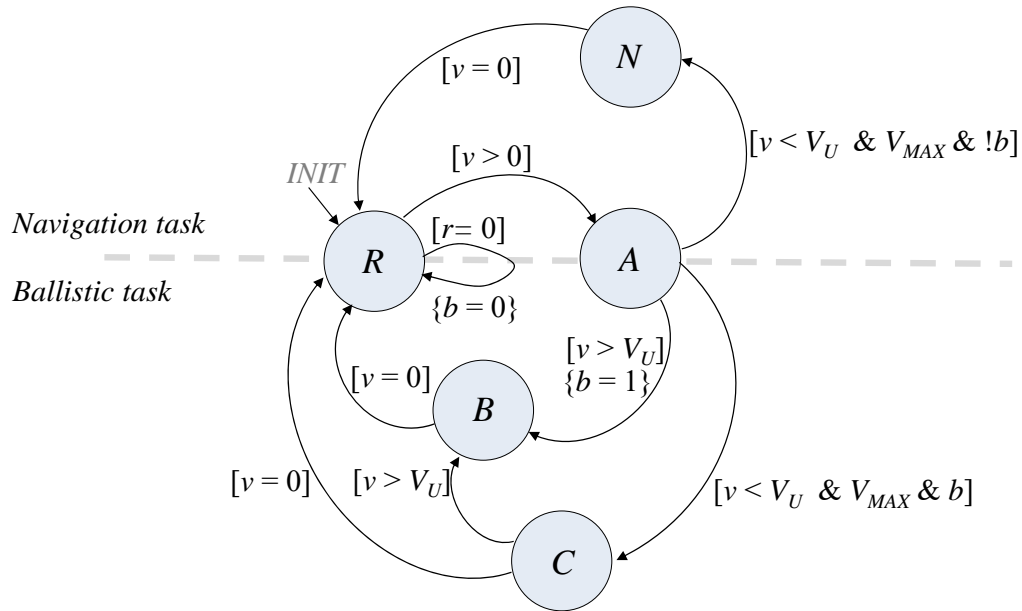


Figure 4. Activation of the states.

b	V_{MAX}	<i>Instantaneous speed</i>			<i>State n</i>				
		$v < V_U$	$v > V_U$	$v = 0$	<i>R</i>	<i>A</i>	<i>B</i>	<i>N</i>	<i>C</i>
-	-	-	-	1	<i>R</i>	<i>R</i>	<i>R</i>	<i>R</i>	<i>R</i>
0	0	1	0	0	<i>A</i>	<i>A</i>	-	-	-
0	1	1	0	0	-	<i>N</i>	-	<i>N</i>	-
1	0	1	0	0	<i>A</i>	<i>A</i>	-	-	-
1	1	1	0	0	-	<i>C</i>	<i>B</i>	-	<i>C</i>
0	0	0	1	0	<i>A</i>	<i>B</i>	<i>B</i>	-	-
0	1	0	1	0	-	<i>B</i>	<i>B</i>	<i>N</i>	-
1	0	0	1	0	<i>A</i>	<i>A</i>	-	-	<i>C</i>
1	1	0	1	0	-	-	<i>B</i>	-	<i>B</i>

State n+1

Figure 5. Distribution around a target of width W and distance D after the initial pulse. *a)* the pulse falls far from the target. *b)* the pulse falls over the target, and *c)* the pulse falls near the target with an error lower than $\varepsilon_1 - W/2$ for a given probability P .

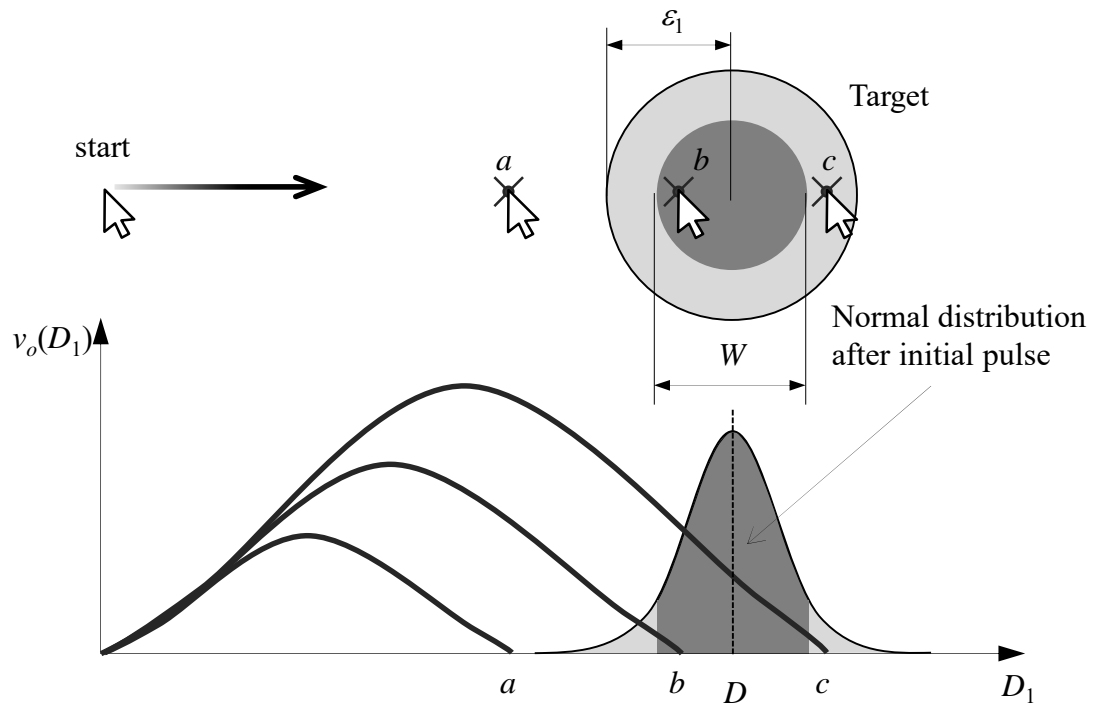


Figure 6. Variation of the velocity v_o and the scale s_M in function of the velocity v_i .

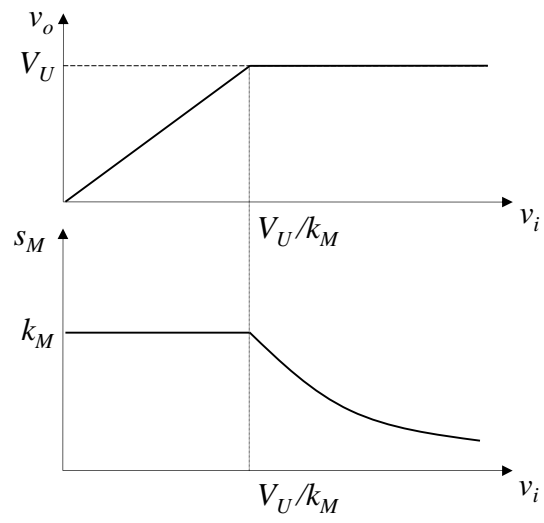


Figure 7. Position and velocity profiles of the initial impulse in the input interface, considering time zero at maximum speed for convenience.

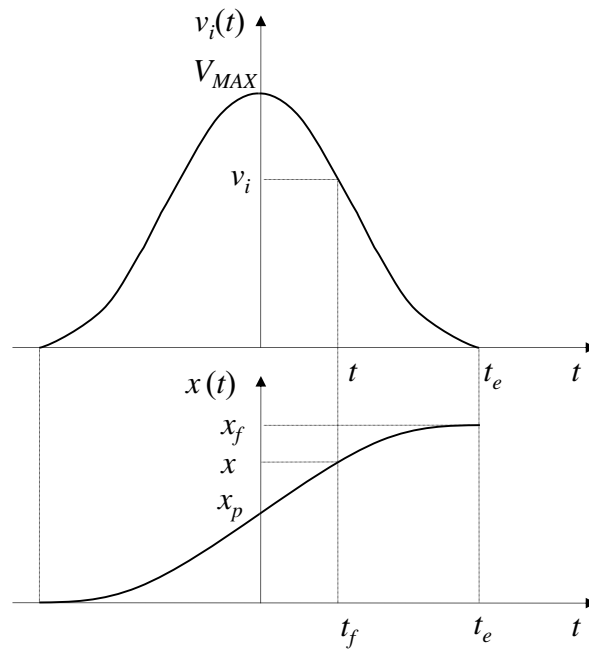


Figure 8. Estimated trajectories at the input and output interfaces considering a linear velocity at each instant t . a) Estimated speed as a linear function. b) Estimated position as a quadratic function.

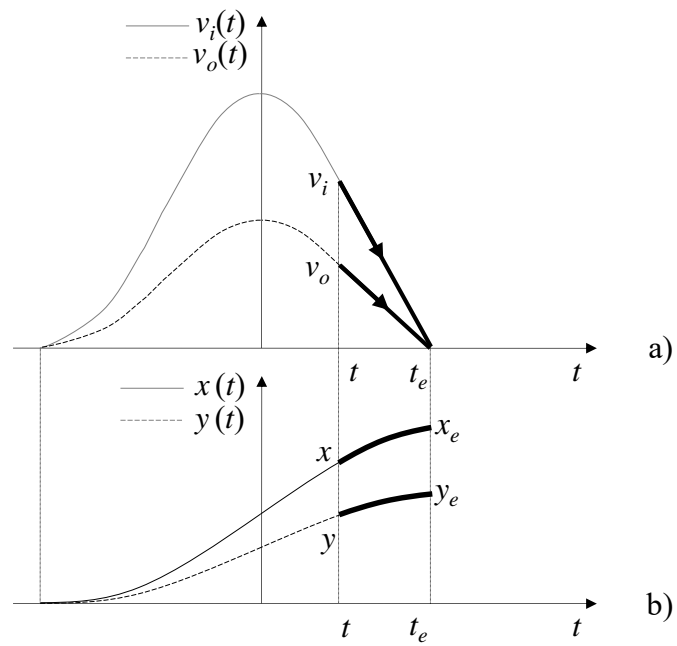


Figure 9. Quiz of subjective evaluation.

EVALUATION

Quiz for subjective evaluation

Evaluate the test performed placing the mark on the place that better fits your experience on the test.

Workload

Low Precision High

Precision

Low Speed High

Speed

Low Frustration High

Frustration

Low Attentiveness High

Attentiveness

OK

EVALUATION

Quiz for subjective evaluation

COMPARE this test with the previous one, placing the mark on the place that better fits your experience on the test.

Workload

Lower No change Higher

Precision

Lower No change Higher

Speed

Lower No change Higher

Frustration

Lower No change Higher

Attentiveness

Lower No change Higher

OK

Figure 10. Input interface devices used on the experiments. a) Logitech G9 Laser Mouse
b) Logitech TrackMan® Marble® c) 3Dconnexion SpaceNavigator® d) Trust Joystick GM-2500.

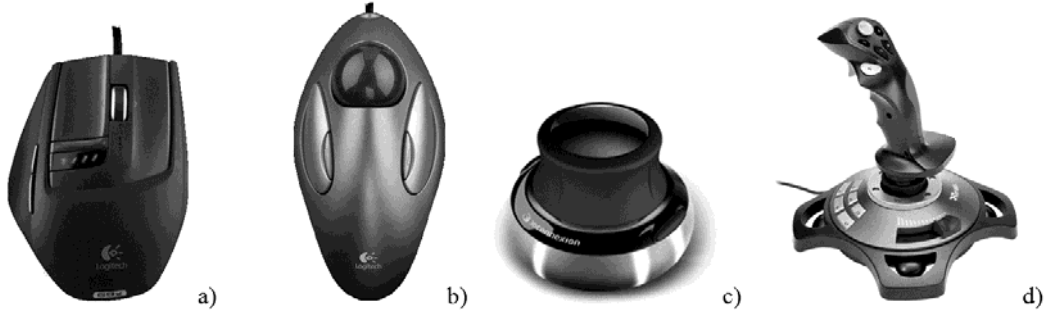


Figure 11. From the input to the output device through conditioning and scaling.

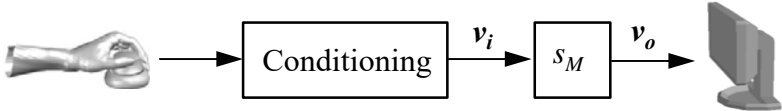
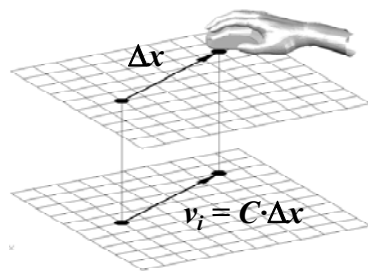


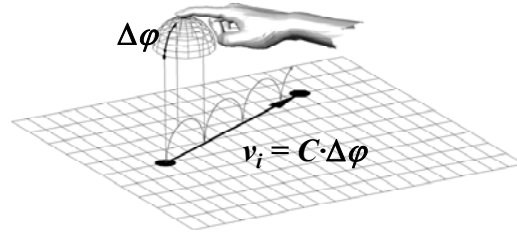
Figure 12. Input interfaces characteristics.

Device	Movement		DOF	Range	Hysteresis	Linearity	Resolution
Mouse	Incremental	Linear	2	∞	± 0.05 mm	± 0.05 mm	500dpi
Trackball	Incremental	Angular	2	$\pm 180^\circ$	± 0.1 mm	± 0.1 mm	300dpi
Joystick	Absolute	Angular	3	$\pm 30^\circ$	1°	1°	10 bits
Joyst-3D	Incremental	Lin./Ang.	6	± 2 mm/ 5°	~ 0	~ 0	12 bits

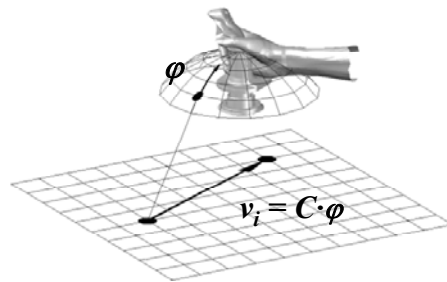
Figure 13. Schematic relationship between the physical space of the input interfaces and the linear command movements, after its conditioning, to obtain the velocity input v_i .



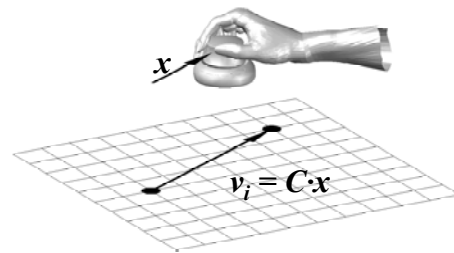
a) Mouse



b) Trackball



c) Joystick



d) Joystick 3D

Figure 14. Input interfaces conditioning.

Device	Physical input	Range	V(mm/s)	C
Mouse	Δx	∞	94	1
Trackball	$\Delta \varphi$	180°	141	1 mm/°s
Joystick	φ	30°	94	4 mm/°s
Joystick 3D	x	2mm	141	88 s ⁻¹

Figure 15. Static scaling function.

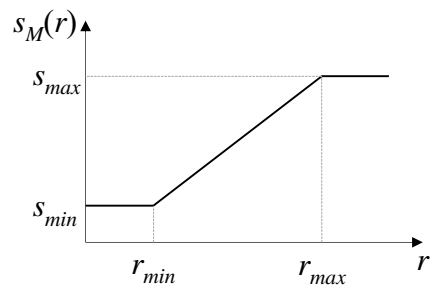


Figure 16. Unidirectional connection experiment setup.

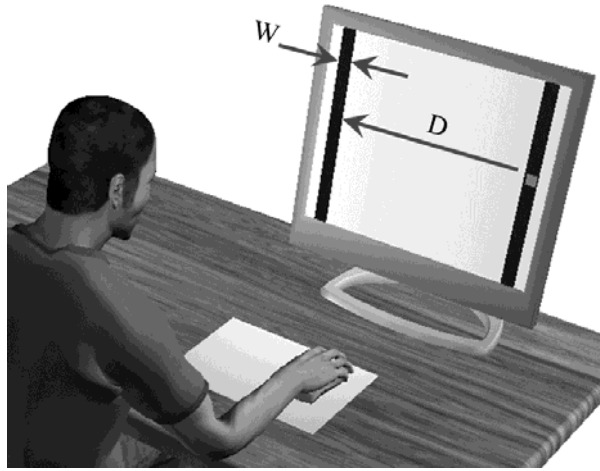


Figure 17. *ID* vs *D-W* combinations.

	D	W	ID = $\log_2(D/W+1)$
ID1	20	8	1.8
ID2	20	2	3.4
ID3	100	8	3.8
ID4	200	8	4.7
ID5	100	2	5.7
ID6	200	2	6.7

Figure 18. *MT* and *TP* for all devices and methods.

av:average	Mouse			Trackball			Joystick			Joystick 3D		
sd: std deviation	Constant	Static	Dynamic	Constant	Static	Dynamic	Constant	Static	Dynamic	Constant	Static	Dynamic
avMT (s)	1.050	0.881	0.656	1.292	1.124	0.945	2.907	2.295	1.464	1.532	1.035	0.680
sd MT (s)	0.379	0.286	0.178	0.453	0.387	0.341	1.798	1.667	1.502	0.750	0.471	0.380
avTP (bits/s)	4.902	5.944	8.690	3.708	4.541	5.837	1.966	2.551	5.279	3.219	4.731	9.245
sdTP (bits/s)	0.801	1.276	3.098	0.923	1.170	2.371	0.8837	1.105	3.090	1.211	1.553	4.541

Figure 19. Average and 95% confidence intervals for all the combinations of *ID* and methods (C: Constant, S: Static and D: Dynamic) for the execution time *MT* and throughput *TP* for all devices.

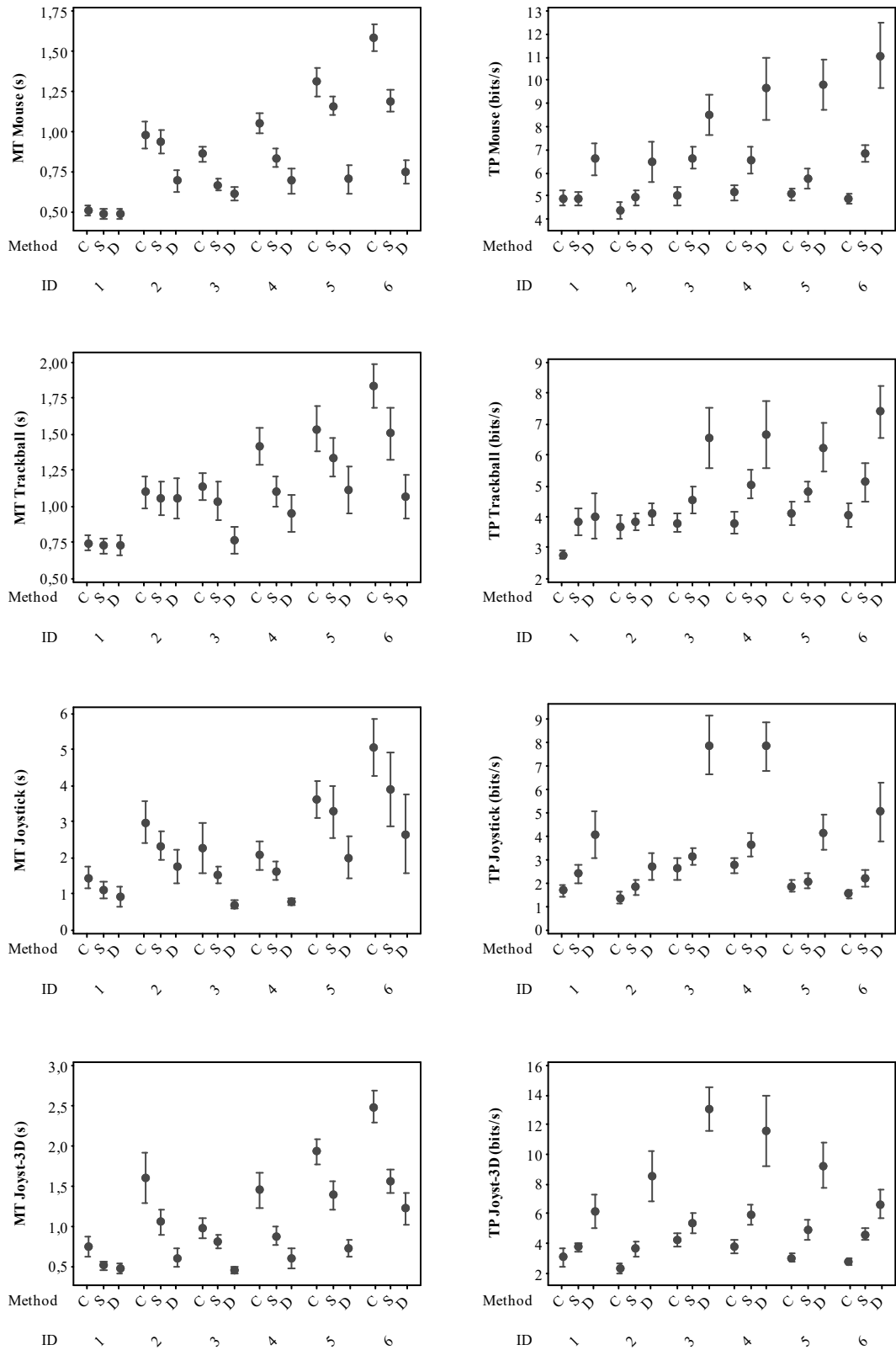


Figure 20. Statistical results from the ANOVA two factors test.

		<i>Execution Time</i>				<i>Throughput</i>			
Mouse									
Source	df	SS	MS	F	p	SS	MS	F	p
Method	2	11.72	5.86	238.4	0.000	1148.7	574.4	83.6	0.000
ID	5	21.89	4.38	431.0	0.000	330.19	66.04	46.74	0.000
Subject	24	6.17	0.26	30.39	0.000	382.52	15.94	10.96	0.000
Method*ID	10	5.82	0.58	68.72	0.000	207.28	20.73	14.26	0.000
Method*Subject	48	1.20	0.03	2.95	0.000	329.83	6.97	4.73	0.000
ID*Subject	120	1.22	0.01	1.20	0.119	169.54	1.41	0.97	0.565
Error	240	2.03	0.01			348.94	1.45		
Total	449	50.05				2917.1			
Trackball									
Source	df	SS	MS	F	p	SS	MS	F	p
Method	2	9.02	4.51	107.38	0.000	345.26	172.6	59.78	0.000
ID	5	25.27	5.05	124.56	0.000	235.18	47.04	49.41	0.000
Subject	24	30.17	1.26	106.84	0.000	413.29	17.22	23.18	0.000
Method*ID	10	5.27	0.53	44.80	0.000	89.47	8.95	12.04	0.000
Method*Subject	48	2.02	0.04	3.57	0.000	138.61	2.89	3.89	0.000
ID*Subject	120	4.87	0.04	3.45	0.000	114.24	0.95	1.28	0.054
Error	240	2.82	0.01			178.30	0.74		
Total	449	79.45				1514.3			
Joystick									
Source	df	SS	MS	F	p	SS	MS	F	p
Method	2	157.29	78.65	327.74	0.000	938.06	469.03	139.87	0.000
ID	5	413.27	82.65	59.98	0.000	464.53	92.91	89.71	0.000
Subject	24	546.87	22.79	80.63	0.000	466.55	19.44	15.48	0.000
Method*ID	10	27.05	2.71	9.57	0.000	203.23	20.32	16.18	0.000
Method*Subject	48	11.62	0.24	0.85	0.747	160.96	3.35	2.67	0.000
ID*Subject	120	165.35	1.39	4.88	0.000	124.28	1.04	0.82	0.882
Error	240	67.82	0.28			301.41	1.26		
Total	449	1389.18				2659.01			
Joystick 3D									
Source	df	SS	MS	F	p	SS	MS	F	p
Method	2	54.87	27.44	343.16	0.000	2948.4	1474.20	112.98	0.000
ID	5	68.08	13.62	293.98	0.000	689.4	137.88	51.16	0.000
Subject	24	40.86	1.70	45.78	0.000	994.64	41.44	16.34	0.000
Method*ID	10	11.30	1.13	30.40	0.000	408.6	40.86	16.11	0.000
Method*Subject	48	3.84	0.08	2.15	0.000	626.34	13.05	5.14	0.000
ID*Subject	120	5.56	0.05	1.25	0.078	323.40	2.67	1.06	0.344
Error	240	8.92	0.04			608.83	2.54		
Total	449	193.43				6599.6			

Figure 21. Results of a subjective evaluation quiz comparing the static and dynamic scaling with respect to the constant scaling in the unidirectional connection test.

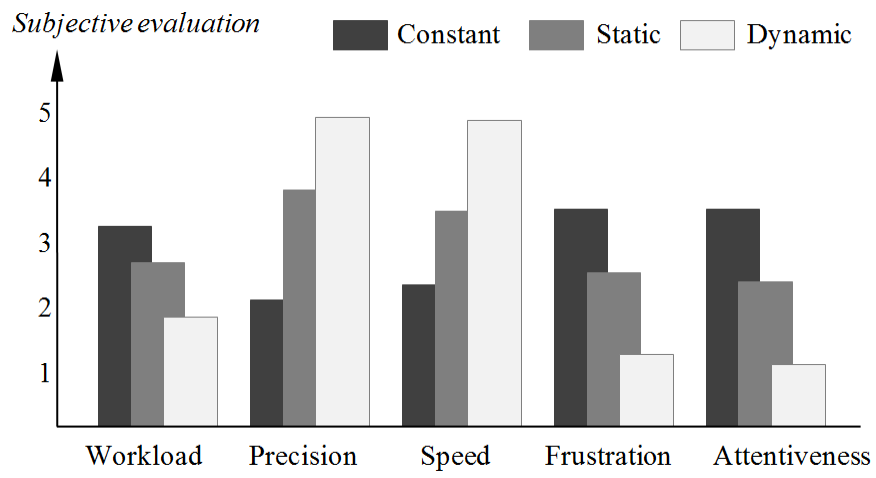


Figure 22. Multidirectional tracing experiment setup.

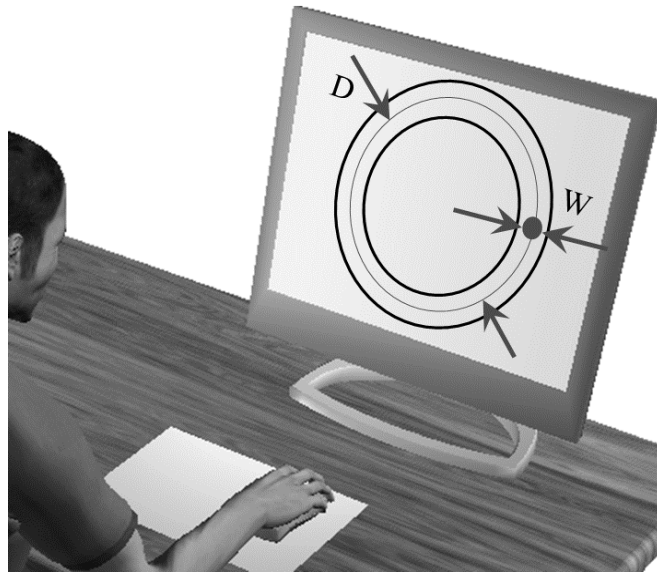


Figure 23. *ID* vs *D-W* combinations.

	<i>D</i>	<i>W</i>	$ID = \pi D / (W - 7)$
ID20	40	4	20.9
ID40	40	7	41.8
ID50	50	4	52.3
ID100	50	7	104.7

Figure 24. *MT*, *TP* and number of faults for all devices and methods.

av: average sd: std deviation	Mouse		Trackball		Joystick		Joystick 3D	
	Constant	Dynamic	Constant	Dynamic	Constant	Dynamic	Constant	Dynamic
avMT (s)	19.57	9.10	33.33	13.10	38.12	19.35	10.91	6.48
sd MT (s)	19.88	5.16	20.66	7.04	28.58	13.64	6.77	2.545
avTP (bits/s)	25.97	46.35	11.14	28.83	12.37	22.51	33.21	57.17
sdTP (bits/s)	10.26	24.00	4.94	16.43	7.93	9.77	14.27	28.23
avFaults	7.17	0.00	4.80	0.09	7.01	0.22	4.23	0.04
sdFaults	9.85	0.00	7.55	0.28	6.91	0.52	4.84	0.19

Figure 25. Statistical results from the ANOVA two factors test.

		<i>Execution Time</i>				<i>Throughput</i>				<i>Number of Faults</i>			
Mouse													
Source	df	SS	MS	F	p	SS	MS	F	p	SS	MS	F	p
Method	1	5471.7	5471.7	92.47	0.000	20702	20702	90.99	0.000	2570.4	2570.4	27.22	0.000
ID	3	19148	6382.8	75.65	0.000	12955	4318.2	38.44	0.000	1169.3	389.78	20.48	0.000
Subject	24	3781.6	157.57	3.36	0.000	19096	795.7	7.24	0.000	2266.7	94.45	4.96	0.000
Method*ID	3	7972.8	2657.6	56.69	0.000	13919	4639.5	42.21	0.000	1169.3	389.78	20.48	0.000
Method*Subjt	24	1420.2	59.17	1.26	0.223	5460.4	227.5	2.07	0.010	2266.7	94.45	4.96	0.000
ID*Subject	72	6075.2	84.38	1.80	0.007	8088.2	112.3	1.02	0.463	1370.0	19.03	1.00	0.500
Error	72	3375.3	46.88			7913.4	109.9			1370.0	19.03		
Total	199	47245				88133				12183			

Trackball													
Source	df	SS	MS	F	p	SS	MS	F	p	SS	MS	F	p
Method	1	20467	20467	189.5	0.000	15643	15643	107.8	0.000	1109.2	1109.2	39.56	0.000
ID	3	20737	6912.3	212.4	0.000	6684.5	2228.2	87.68	0.000	795.26	265.09	13.94	0.000
Subject	24	13925	580.24	43.65	0.000	9178.5	382.44	12.40	0.000	694.52	28.94	1.61	0.062
Method*ID	3	6600.4	2200.1	165.5	0.000	5769.6	1923.2	62.35	0.000	831.90	277.30	15.47	0.000
Method*Subjt	24	2591.9	108.00	8.12	0.000	3481.2	145.05	4.70	0.000	672.92	28.04	1.56	0.075
ID*Subject	72	2343.6	32.55	2.45	0.000	1829.7	25.41	0.81	0.793	1369.1	19.02	1.06	0.401
Error	72	957.13	13.29			2220.9	30.85			1290.5	17.92		
Total	199	67623				44807				6763.4			

Joystick													
Source	df	SS	MS	F	p	SS	MS	F	p	SS	MS	F	p
Method	1	17605	17605	47.20	0.000	5134.1	5134.1	92.70	0.000	2305.2	2305.2	47.56	0.000
ID	3	13931	4643.7	158.7	0.000	3076.4	1025.5	68.82	0.000	565.41	188.47	20.26	0.000
Subject	24	70336	2930.7	133.0	0.000	9348.5	389.52	35.54	0.000	1160.7	48.36	4.91	0.000
Method*ID	3	2392.9	797.63	36.2	0.000	71.70	23.90	2.18	0.098	487.62	162.54	16.49	0.000
Method*Subjt	24	8952.3	373.0	16.93	0.000	1329.2	55.38	5.05	0.000	1163.2	48.47	4.92	0.000
ID*Subject	72	2106.4	29.3	1.33	0.116	1072.8	14.90	1.36	0.097	669.71	9.30	0.94	0.596
Error	72	1586.6	22.0			789.11	10.96			709.51	9.85		
Total	199	116911				20822				7061.4			

Joystick 3D													
Source	df	SS	MS	F	p	SS	MS	F	p	SS	MS	F	p
Method	1	979.11	979.11	93.05	0.000	28702	28702	593.2	0.000	877.81	877.81	47.07	0.000
ID	3	2624.6	874.87	178.5	0.000	33264	11088	137.4	0.000	374.38	124.79	27.00	0.000
Subject	24	1274.0	53.08	37.95	0.000	36357	1514.9	29.70	0.000	483.73	20.16	4.19	0.000
Method*ID	3	573.04	191.01	136.6	0.000	18809	6269.8	122.9	0.000	346.46	115.49	23.99	0.000
Method*Subjt	24	252.53	10.52	7.52	0.000	1161.3	48.40	0.95	0.540	447.57	18.65	3.87	0.000
ID*Subject	72	352.91	4.90	3.50	0.000	5808.5	80.7	1.58	0.027	332.75	4.62	0.96	0.569
Error	72	100.71	1.40			3672.9	51.00			346.67	4.82		
Total	199	6156.9				127775				3209.4			

Figure 26. Average and 95% confidence intervals for all de combinations of *ID* and methods (C: Constant, and D: Dynamic) for the execution time *MT*, throughput *TP* and number of faults for all devices.

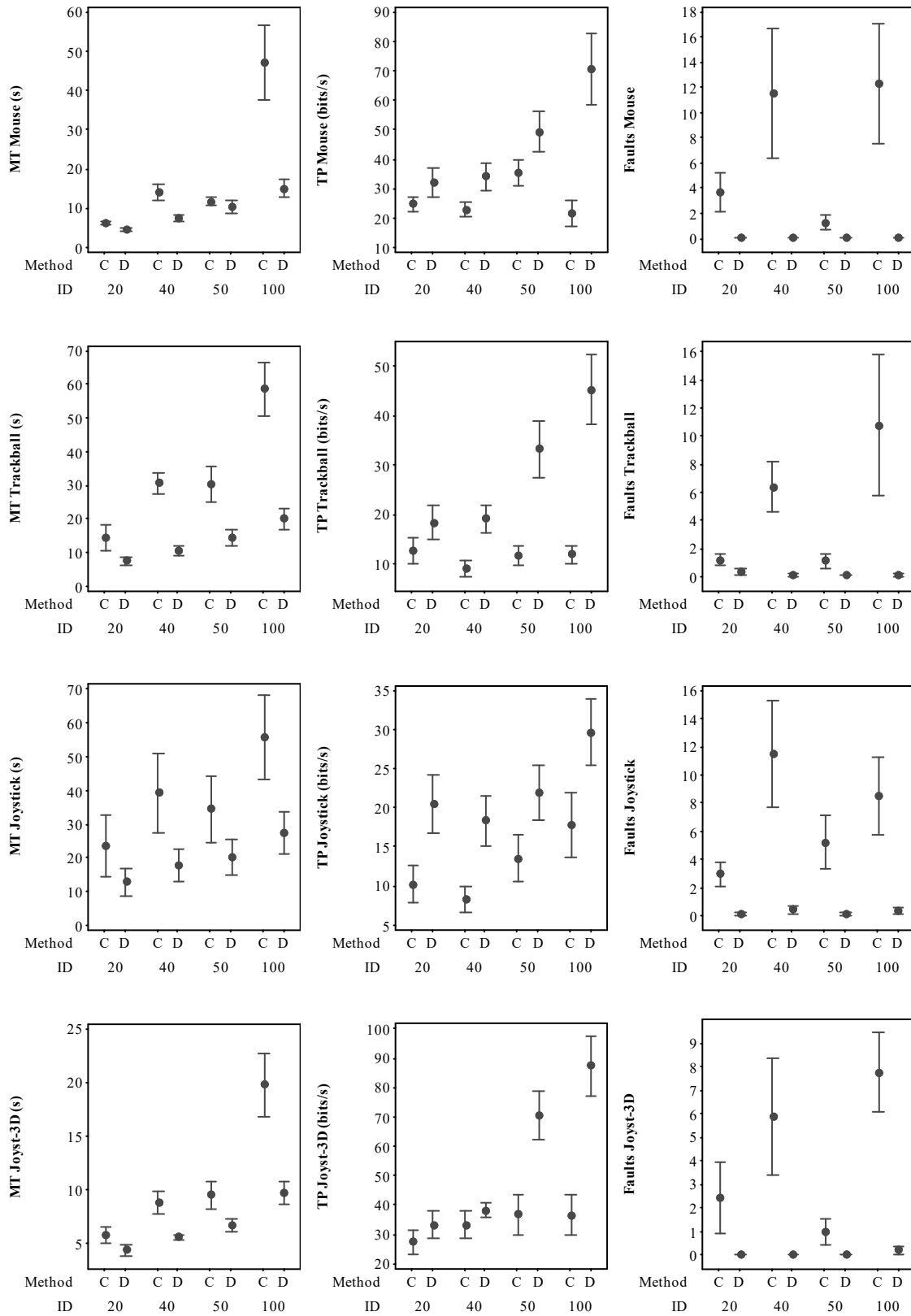


Figure 27. Results from a subjective evaluation quiz comparing static and dynamic scaling with respect to constant scaling on the unidirectional connection test.

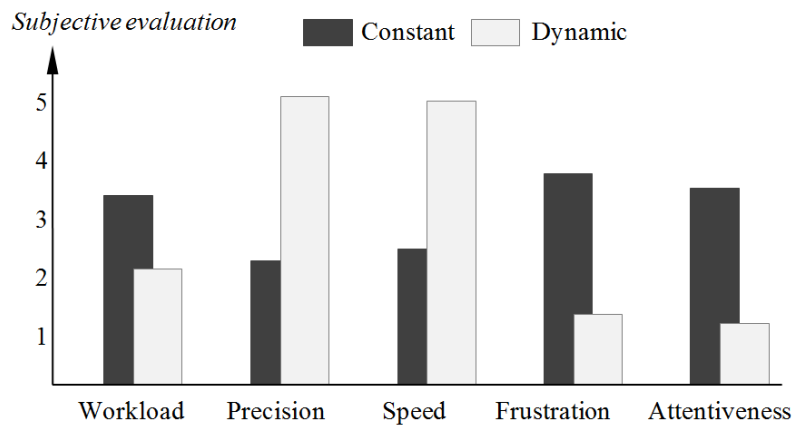


Figure 28. Average TP comparison between the Constant and Dynamic scaling methods obtained for the four input devices and for all data on the experiments a) Unidirectional connection test b) Multidirectional tracing test.

