Performance of Mobile Haptic Interfaces

A.Formaglio, A.Giannitrapani, F.Barbagli, M.Franzini, and D.Prattichizzo

***Abstract*— One of most interesting aspects in haptic research deals with the extension of application workspace, thus allowing haptic simulation within large virtual environments. Several devices have been realized that allow this kind of interaction, in particular our interest focuses on *mobile haptic interfaces*, realized by combining classic grounded haptic devices with mobile platforms. While grounded haptic interfaces feature spatial limitations due to manipulator dimensions, mobile haptic interfaces are characterized by dynamical limitations due to performance of employed devices. In this paper we introduce an experimental analysis to evaluate performance of mobile haptic interface from a transparency standpoint. Moreover, some basic guidelines are presented to enhance MHI performance by setting the controller depending on technical parameters characterizing used devices.**

I. INTRODUCTION

The workspace of haptic interfaces varies largely on their design and usage, ranging from few planar centimeters of the Pantograph [1] to several cube meters of the Scaleable Spidar device [2]. Most haptic devices, however, share two main traits: they are grounded and they have limited workspace. While this is not a problem in many applications, it can become one in cases where users need to interact with large virtual environments while navigating inside of them.

A possible solution for this problem is to use locomotion interfaces, i.e. trademill-like interfaces that simulate some of the inertial feedback that a user would experience while navigating through a large virtual environment [6]. Another possible approach is to create haptic interfaces featuring unlimited workspace by combining mobile robots and standard grounded force-feedback devices. This type of interface, which in part resembles the *cobot* [3], was introduced by Nitzsche et al [4] and is referred to as *mobile haptic interface* (MHI). Such mobile device features unlimited workspace, allowing user to walk around during haptic interaction. However, since mobile robots are generally characterized by slow dynamics, delays may be present when attempting to track the human operator motion. As a consequence the operator may feel the boundary on the haptic device workspace, which in turn can create spurious forces and ultimately cause a total loss of transparency for the MHI. Therefore, MHI should be designed to avoid these critical events, but mechanical limitations affect both haptic device and mobile robot, thus determining some dynamical constraint to end-effector motion.

This paper introduces novel techniques to study transparency of a mobile haptic interface with respect to some performance indicators, and basic guidelines to improve performance of a given MHI by tuning controller parameters. Theoretical results have been validated through several experimental trials involving a MHI prototype composed by Phantom Premium 1.5 haptic interface and Nomad XR4000 mobile robot.

The paper is structured as follows. Section II introduces the dynamical model of a mobile haptic interface. In Section III, performance of a MHI is discussed from a transparency viewpoint. Sections IV-VI present theoretical results and experimental validation with respect to different classes of input signals. Such results are discussed in Section VII. Finally, the experimental setup is described in Section VIII, whereas in Section IX conclusions and future perspectives are reported.

II. MODELLING MOBILE HAPTIC INTERFACES

A. Control algorithms

MHIs are designed to allow users to interact with objects displaced in large virtual environments. For this purpose, a MHI is made up of a mobile platform (MP) and an impedance-type haptic device (HD), grounded to the MP (Fig. 2, left side). In order to transparently render any impedance inside an unlimited environment we propose a simple control algorithm, which mimics the one proposed in [4], where forces are rendered using standard constrained based methods such as the proxy algorithm [8].

Fig. 1. A general purpose interface based on the holonomic Nomad mobile robot and a Phantom

For the sake of clarity in the following we will model a MHI featuring a single degree of freedom, i.e. motion is allowed only along a straight direction.

A.Formaglio, A.Giannitrapani, M.Franzini and D.Prattichizzo are with the Dipartimento di Ingegneria dell'Informazione - University of Siena - Via Roma 56, 53100, Siena, Italy formaglio@dii.unisi.it

F. Barbagli is with the Stanford Robotics Lab - Stanford University 353 Serra Mall, Stanford, California, 94131 barbagli@robotics.stanford.edu

Let us consider the world reference frame Σ_W and reference frame Σ_M which is attached to the mobile platform (see Fig. 2). X_p represents the position of the HD end effector

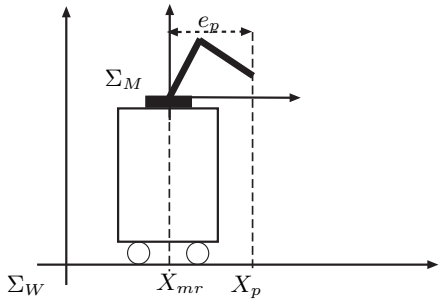

Fig. 2. Model of a MHI featuring 1 DoF.

with respect to Σ_W, X_{mr} the position of the MP with respect to Σ_W, e_p is the position of the HD end effector with respect to Σ_M (as well as the tracking error for the mobile platform). $X_{ref}(s) = C(s)e_p(s)$ (see Fig. 3, right side) is the position commanded to the MP, $H(s)$ is the transfer function representing the MP, and finally $C(s)$ is the transfer function of the MP control algorithm. A PD controller is chosen in order for the platform to track the position of the user with respect to the world, i.e. to bring e_p to zero [4]. Thus

$$C(s) = sD_m + P_m$$

In order to control the interaction forces between user

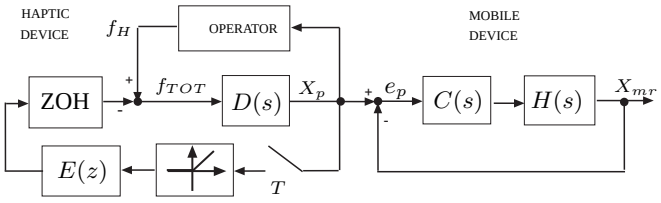

Fig. 3. Control scheme for a mobile haptic interface: interaction force rendering algorithm (left side) and position control algorithm for the mobile platform (right side).

and virtual environments we use a classic scheme that was introduced by Colgate in [7], see Fig. 3 (left side). $D(s)$ is the MHI reflected inertia felt by the operator; $E(z)$ is the discrete time transfer function of the virtual environment impedance; finally ZOH is a zero holder hold. The impedance model of a virtual object can be chosen as a discrete spring-damper system, yielding

$$E(z) = D_h \frac{z-1}{zT} + P_h$$

where T is the haptic servo-loop period. This simplified model is normally used to represent virtual walls, nevertheless it can be generalized to more complex cases by adding details on collision detection.

B. Modelling mobile robot

Since analytic dependencies between controller parameters (P_m, D_m) and given performance specifications are sought, a simple LTI model is adopted for the mobile platform. A possible choice, trading off accuracy and complexity, is to represent the MP as a second order mass-spring-damper system [4]:

$$F_{mr}(s) = K_m X_{ref}(s) - B_m s X_{mr}(s).$$

Hence the MP transfer function $H(s)$ (see Fig.3) takes on the form:

$$H(s) = \frac{K_m}{s(M_m\, s + B_m)} = \frac{X_{mr}}{X_{ref}}. \qquad (1)$$

where K_m is the spring stiffness, B_m is the damping coefficient and finally M_m is the mass of the platform.

While this is far from being an exact dynamical model of a mobile robot and cannot account for non-linear effects that are present in the real world, it has the advantage of being simple and of being characterized by a small set of parameters that can easily be interpreted.

III. PERFORMANCE OF A MOBILE HAPTIC INTERFACE

The $Z-$width of a MHI depends on how MP and HD are controlled [7]. The addition of a MP, whose inertia is usually fairly large and whose dynamics are normally slower than those of the HD, may affect the $Z-$width of a MHI. For example, the MP may lag behind the HD, depending on user motion dynamics and on performance limitations of MHI. In this case, while the end-effector reaches the boundary of HD workspace, the user feels a spurious force due to HD singular configuration, i.e. $Z^* \neq Z$. This causes a total loss of transparency, and the control scheme on Fig. 3 ceases to be valid.

At each instant t and at each position X_p, the total force f_{TOT} acting on the end-effector is given by

$$f_{TOT} = ZX_p - f_{HO} + f_F + f_{WS}$$

where ZX_p is the virtual reaction force to be rendered by MHI, f_{HO} is exerted by the human operator during simulation, f_F represents the friction force due to MHI mechanical structure, and finally f_{WS} is the spurious force felt by user while end-effector reaches the boundary of its workspace. In order for $Z^* = Z$ to hold, it should always be $f_{TOT} = ZX_p - f_{HO}$, i.e. user feels the correct virtual world impedance. Such condition yields $f_F = 0$ and $f_{WS} = 0$. We can neglect friction force f_F, due to high backdriveability of used HD, but f_{WS} cannot be eliminated. It is only possible to evaluate within what conditions its influence cannot be felt.

In this paper a methodology is presented to experimentally evaluate performance limitations of a given MHI, yielding some dynamical constraints to end-effector motion in order for it to never reach the boundaries of its workspace. Along

this line, an analytic method is presented to tune the parameters of PD controller in order for MHI to reach desired performance specifications. Such analysis will focus on three main types of inputs that closely resemble an operator's movements: step input, ramp input, and sinusoidal input. This does not cover all possible scenarios, since X_p is a hand generated signal, notwithstanding, the results must be considered as rough indicators of performance.

IV. STEP DISPLACEMENT

In this section we analyze the case of end-effector step displacement. The target is to evaluate what is the maximum amplitude of a step displacement X_p that can be correctly rendered by the MHI. Let us consider an ideal step signal of amplitude A_u. At time $t = 0^+$ we have

$$X_{mr}(0^+) = 0 \text{ and } X_p(0^+) = A_u$$

i.e. while the system output is still at zero the input has jumped to A_u. In this case $e_p(0^+) = A_u$, i.e. the haptic device is at a A_u distance from the center of its workspace, and thus it is necessary that the maximum amplitude of a step position signal applied to a MHI is such that $A_u < X_{ws}$ (in second order stable systems, the step response envelope is monotone decreasing, so the maximum of the response is in $t = 0^+$):

$$A_u < X_{ws} \;\; (Step \;\; limitation)$$

A. Experimental results

During experimental trials, position steps $X_p(t) = X_0\mathbf{1}(t)$ with different amplitudes, have been used as reference signals. As correctly predicted by the models, the results obtained confirmed that as long as the amplitude of the position steps is inside the workspace limit, the end-effector will never reach its maximum extension (see Figure 4). The

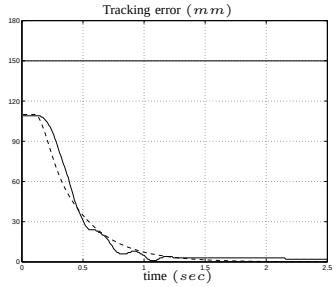

Fig. 4. Step input: $X_p(t) = 100$. Step response of a mobile haptic interface realized combining HD Phantom Premium 1.5 with MP NomadXR4000.

figure above shows actual (solid line) and predicted (dashed line) tracking error. Thick solid line represents the desired maximum error, due to workspace limits. As shown in the graphs, model error matches real system behavior, and the prediction of the maximum error $e_p(t)$ was obviously exact.

B. Performance enhancement

Since the system stability is preserved for any positive value of the controller parameters $P_m,\ D_m$, the maximum tracking error is still attained at $t = 0^+$ and does not depend on the choice of $P_m,\ D_m$.

V. RAMP DISPLACEMENT

In this section, ramp response limitation are analyzed. The target is to evaluate what is the maximum position ramp of X_p that can be correctly rendered by the MHI. Let us consider an ideal ramp $X_p = V_R t$. Referring to MP model of Fig. 3, let be $G(s)$ the error transfer function, such that:

$$e_p(s) = G(s)X_p(s) = \frac{1}{1 + C(s)H(s)}X_p(s)$$

The MHI tracks ramp input with a finite steady-state error, since $H(s)$ has one pole in the origin. Such error is given by

$$e_p(\infty) = \lim_{s\to 0} s e_p(s) = \frac{V_R}{K_v}$$

where K_v is the velocity gain and is defined as follows:

$$K_v = \lim_{s\to 0} sC(s)H(s) = \frac{K_m P_m}{B_m}$$

The maximum error between HD and MP positions e_{max} can be reached during the transient and in such case it can be found using

$$e_{po} \triangleq \frac{e_{max} - e_p(\infty)}{e_p(\infty)}$$

where e_{po} does not depend on the slope of the ramp but only on $H(s)$ and $C(s)$. It is clear that if the system tracking error has not overshoot peak, i.e. $e_{max} = e_p(\infty)$, the equation above becomes:

$$e_{po} = 0$$

Combining the equations above, it yields

$$e_{max} = (1 + e_{po})\frac{V_R}{K_v}$$

that shows that e_{max} is proportional to V_R. By experimentally determining the value of e_{po} for a given MP, we can then compute e_{max}. In order for $e_{max} < X_{ws}$ to be true, when applying $X_p(t) = V_R t$, V_R must be such that

$$V_R < \frac{X_{ws}\, K_v}{(1 + e_{po})} \;\; (Slope \;\; limitation) \qquad (2)$$

Note that such limitation is valid both in case of systems with an overshoot in the ramp response and in systems without overshoot.

A. Experimental results

During the experimental trials, the MHI was excited with the desired reference signal. A HD featuring lower workspace limits than the actual real ones was simulated (thick solid lines in Figure 5). The target was to show both inputs yielding $f_{WS} = 0$ and inputs leading to cross HD simulated workspace limit, thus leading to $f_{WS} \neq 0$. Figure 5 shows step response of the MHI realized with NomadXR4000: the solid line represents the actual error while the dashed line represents the predicted error. The maximum tracking error of the Nomad MHI is almost proportional to the input velocity, as predicted by the theoretical analysis.

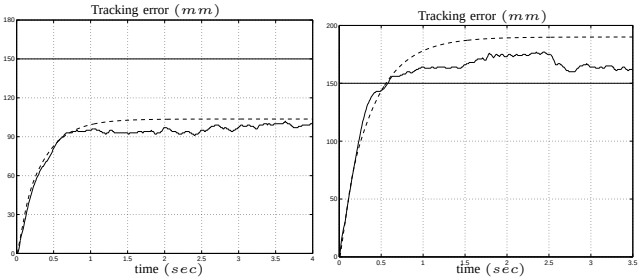

Fig. 5. Ramp tracking errors of a mobile haptic interface realized combining MP NomadXR4000. On the left, ramp input is $X_p(t) = 300t$, on the right it is $X_p(t) = 500t$.

B. Performance enhancement

In order to increase the maximum slope of the trackable ramps given a prescribed X_{ws}, the controller parameters can be determined as follows. First, notice that $e_{max} \geq e_p(\infty)$. Thus, a slope V_R can be transparently tracked only if $\frac{V_R}{K_v} \leq X_{ws}$. Recalling the expression above of K_v, the latter inequality yields a lower bound on P_m

$$P_m \geq \frac{B_m}{X_{ws} K_m} V_R. \tag{3}$$

Once a suitable value has been set for P_m according to (3), a proper choice of the derivative gain D_m can prevent the error signal $e(t)$ from exhibiting overshoot, thus ensuring $e_{max} = e_p(\infty)$. By straightforward computation [11], it turns out that the error transfer function $G(s)$ has complex poles iff

$$\Delta = aD_m^2 + bD_m - cP_m + d < 0 \tag{4}$$

where a, b, c, d are positive coefficients depending only on the robot parameters. The equation $\Delta = 0$, represents a parabola in the plane (D_m, P_m), as shown in Figure 6. Hence, a necessary condition in order to avoid overshoot is that $\Delta \geq 0$. However, even in case of real poles, the error signal can still exhibit overshoot for some choice of the controller gains. This values can be analytically determined (see [11]) as those satisfying the following relations

$$\Delta > 0 \qquad \text{Real poles} \tag{5}$$

$$P_m \geq P_m^c \qquad \text{Possible overshoot} \tag{6}$$

$$P_m \geq \frac{D_m}{\tau} \qquad \text{Overshoot} \tag{7}$$

where $P_m^c = \frac{B_m^3}{K_m M_m^2}$, and $\tau = \frac{M_m}{B_m}$ denotes the time constant of the mobile platform. Hence, there exist real poles giving rise to overshoot if $P_m > P_m^c$ and (P_m, Dm) are between the line $P_m = \frac{D_m}{\tau}$ and the parabola $\Delta = 0$ (see Figure 6). Thus, from equations (4)-(7) it follows that the error signal has no overshoot iff the pair (P_m, D_m) is such that $\Delta > 0$, and either

$$P_m < P_m^c$$

or

$$P_m \geq P_m^c \qquad \text{and} \qquad D_m > \tau P_m$$

The values (P_m, D_m) which ensure that no overshoot shows up are depicted in Figure 6 (shaded region), for a Nomad-like model. Notice that the boundary of such a region can be analytically computed very easily and it depends only on the mobile platform parameters (M_m, B_m, K_m). At this point, the design of the controller guaranteeing $e_{max} < X_{ws}$ for a given ramp V_R can be summarized as follows:

1) Choose a value P_m^* s.t. (3) is satisfied.
2) If $P_m^* < P_m^c$, choose a value D_m^* s.t. $\Delta > 0$, otherwise $(P_m^* \geq P_m^c)$ choose a value D_m^* s.t. $D_m^* > \tau P_m^*$.

The above procedure has a simple graphical interpretation. For the chosen P_m^* according to (3):

- if $P_m^* < P_m^c$ select a value for D_m^* s.t. the pair (P_m^*, D_m^*) lies on the right of the parabola $\Delta = 0$ (solid line in Figure 6);
- if $P_m^* > P_m^c$ select a value for D_m^* s.t. the pair (P_m^*, D_m^*) lies on the right of the line $D_m = \tau P_m$ (dashed line in Figure 6).

Clearly, the controller designed for a given $\bar{V}_R$ ensures $e_{max} < X_{ws}$ for all slopes $V_R \leq \bar{V}_R$.

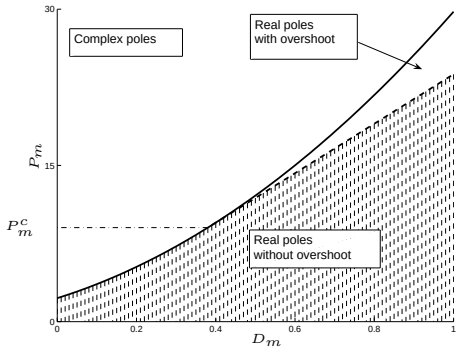

Fig. 6. Controller design for ramp response: parabola $\Delta = 0$ (solid line), line $D_m = \tau P_m$ (dashed line). The shaded region represents the values (P_m, D_m) ensuring no overshoot.

VI. SINUSOIDAL DISPLACEMENTS

In this section we investigate what is the maximum amplitude/frequency of $X_p(t) = A_s sin(\omega t)$ that can be correctly rendered by the MHI. Referring to Fig. 3 let be $G(s)$ the error transfer function as defined in Section V. Given the linearity of the overall system model, in steady state we have

$$e_p(t) = A_s \|G(j\omega)\| sin(\omega t + \angle(G(j\omega)))$$

and thus in order for $e_p \in (-X_{ws}; X_{ws})$, inequality $A_s\|G(j\omega)\| < X_{ws}$ must hold. Since $G(s)$ has high-pass filter behavior, higher-frequency sinusoids must have lower amplitude in order for the MHI to track them and viceversa.

It is always possible to analytically compute the following region of the (ω, A_s) plane:

$$\mathcal{I} = \{(\omega, A_s) : A_s\|G(j\omega)\| < X_{ws}\},$$

More specifically the curve

$$\gamma : A_s\|G(j\omega)\| = X_{ws}$$

representing the border between $\mathcal{I}$ and the rest of the (ω, A_s) plane can be numerically computed. The open region $\mathcal{I}$ represents the sinusoidal inputs that can be correctly rendered by a MHI:

$$(\omega, A_s) \in \mathcal{I} \;\; (Sinusoidal \;\; limitation)$$

It is important to note that such analysis only applies to steady-state behavior of the system.

A. Experimental results

This experimental trial have been performed with sinusoidal inputs $X_p(t) = A_s \sin(\omega t)$, featuring different amplitudes A_s and frequencies ω.

The experimental results obtained with sinusoidal inputs support the theoretical analysis. In Figure 7 sine tracking error of the tested system is shown (solid line) compared to predicted errors (dashed line), and again we simulated a workspace limit X_{ws} (thick solid line). The maximum predicted and actual error for the Nomad MHI are very close.

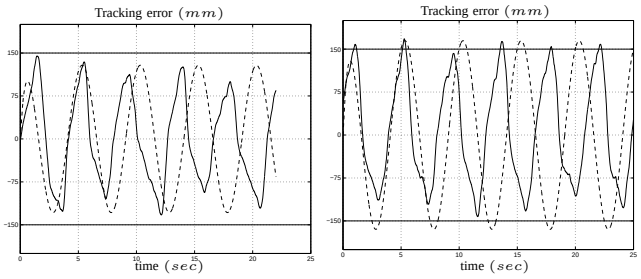

Fig. 7. Sine tracking errors of a mobile haptic interface realized with MP NomadXR4000. On the left, sinusoidal input is $X_p(t) = 250sin(2\pi 0.2t)$, on the right it is $X_p(t) = 300sin(2\pi 0.2t)$.

B. Performance enhancement

In order to make the MHI able to track a sinusoidal reference input (ω, A_s) without exceeding the workspace, the controller parameters must be chosen s.t. $\|G(j\omega)\| \leq \frac{X_{ws}}{As}$. To this purpose, let us study the dependency of $\|G(j\omega)\|$ on (P_m, D_m). After some calculation, it can be shown that

$$\|G(j\omega)\| = \frac{\sqrt{\beta_0}}{\sqrt{\beta_0 + \beta_1 P_m^2 + \beta_2 D_m^2 - \beta_3 P_m + \beta_4 D_m}} \quad (8)$$

where β_i are positive functions of ω and of the robot parameters. Denoting by $\mathcal{E}(P_m, D_m)$ the argument of the square root at the denominator, the required parameters (P_m, D_m) are those s.t.

$$\mathcal{E}(P_m, D_m) \geq \beta_0 \frac{A_s^2}{X_{ws}^2} \triangleq C. \quad (9)$$

Notice that the curve $\mathcal{E}(P_m, D_m) = C$ represents an ellipse in the (P_m, D_m) plane, whose center and radii can be analytically computed from the coefficients β_i and the desired value C [11]. From a geometrical viewpoint, inequality (9) states that all the parameter values ensuring $e_{max} < X_{ws}$ are those lying outside the ellipse $\mathcal{E}(P_m, D_m) = C$. By tracing all the ellipses $\mathcal{E}_i$ corresponding to a set of sinusoids (w_i, A_{s_i}) at different frequencies and amplitudes, it is possible to define the region $\mathcal{F}$ in the (P_m, D_m) plane defined as

$$\mathcal{F} = \bigcup_i \mathcal{E}_i$$

In order for the MHI to transparently track the chosen set of sinusoids, the controller parameters (P_m, D_m) must lie outside $\mathcal{F}$.

VII. DISCUSSION

In Sections V-B and VI-B, two different methods are presented to design PD controller, in order for the MHI to feature required performance with respect to two different input signals. Designers can choose the pair (P_m, D_m) that matches both project criteria, thus simultaneously guaranteeing performance specifications w.r.t. ramp and sinusoidal input. Finally, such design method is completely based on the model of a MHI. Possible differences between analytic results and real system behaviors can arise accordingly to some implementation issues described in Section VIII.

Now some observations can be drawn. The performance limitations of a MHI propagate to hand effector position velocity: for ramp signals the limitation depends on the ramp derivative V_R; for sinusoids, dynamical limitation is roughly connected to maximum velocity $A_s\omega$ of the end-effector, i.e. the maximum admissible amplitudes of sinusoidal inputs are roughly inversely proportional to the sinusoid frequency. This result can be extended to a wider class of input signals (i.e. not necessarily periodic or persistent). To this purpose the derivative of the end-effector position is considered. Hence, let us define the transfer function

$$F(s) = \frac{1}{s(1 + C(s)H(s))}$$

that represents closed-loop relationship between position tracking error of MHI and the velocity of the end-effector: $E_p(s) = F(s)V_p(s)$. For general L_2-norm bounded signals V_p, the L_∞ limitation on the tracking error can be written as

$$\|e_p\|_\infty \leq \|F\|_2 \|V_p\|_2$$

The worst case analysis yields:

$$\|V_p\|_2 < \frac{X_{ws}}{\|F\|_2} \Rightarrow |e_p(t)| < X_{ws}, \forall t. \qquad (10)$$

This conservative constraint can be relaxed: the signal V_p, being the velocity of a hand-made motion, features a limited frequency bandwidth [9]. This can be modelled through a fictitious low-pass filter $G_{lp}(s)$ in cascade to $F(s)$, whose bandwidth corresponds to spectrum of V_p. Hence, the condition (10) becomes:

$$\|V_p\|_2 < \frac{X_{ws}}{\|G_{lp}F\|_2} \Rightarrow |e_p(t)| < X_{ws}, \forall t.$$

Using a first order low-pass filter, it is possible to numerically compute the L_2-norm $\|G_{lp}F\|_2$ and observe its qualitative dependency on ω_c. While increasing ω_c, (i.e. the bandwidth of V_p), also $\|G_{lp}F\|_2$ increases, thus the maximum norm $\|V_p\|_2$ of the admissible inputs decreases, and viceversa.

VIII. EXPERIMENTS

The experimental results described in Sections IV-A, V-A and VI-A, have been validated using the Phantom Premium 1.5 haptic interface and the mobile platform Nomad XR4000 [10]. A preliminary identification campaign has been carried out before experimental validation. Several sets of input-output data $\{X_{ref},\ X_{mr}\}$, corresponding to different classes of input signals (e.g., square waves, ramps, sinusoids), have been collected. The values of the model parameters K_m and B_m have been tuned by comparing the actual and simulated outputs. Note that M_m was known a-priori from the technical specifications of the employed devices [10].

All tests have been performed along a single degree of freedom and for each experimental trial, the actual tracking error has been compared to the one predicted by the corresponding model. As pointed out in Section II-A, the tracking error e_p was directly available from the readings of the haptic interface encoders.

To perform experiments several strategies have been applied to excite real MHI systems with the right input. The ideal step has been generated by fixing the haptic device's end-effector to a given position A_u to the side of the center of the workspace, before the robot starts the tracking. Thus HD encoder reading was $e_p(0^-) = A_u$ and once tracking was started, system input at time $t = 0^+$ was exactly $X_p(0^+) = A_u$, where A_u was the desired step amplitude.

To excite MHI system with signals as near as possible to ideal ramps, a second mobile robot was used as a driver. Thus, while the MHI stood still, the driver robot was accelerated in order to reach a desired velocity V_R and then hooked up to the MHI end-effector (through a velcro connection), thus exciting the MHI with the desired reference signal.

Finally, sinusoidal reference signals have been generated by a human operator who, with the aid of periodic acoustic and visual stimuli, moved the haptic device's end-effector sideways, approximately describing a time dependent sinusoid.

It is worth to note that several factors affected the correct execution of experiments. We can point out the most relevant implementation issues: the MP model cannot account for non-linear dynamics; difficulties in generation of correct input signals exciting end-effector; unavoidable communication delays between HD and MP [10]. However, despite the difficulty to accurately reproduce the time evolution of the tracking error, the procedures proposed in this paper were able to correctly predict the outreach of the HD workspace.

IX. CONCLUSIONS

This paper presents various procedures that can be used to pre-evaluate how and how much Z-width that can be rendered by a MHI is affected by MP dynamics. The proposed analysis may serve as a useful tool for the evaluation of MHI's performance limitations and theoretical results are in good agreement with the real behavior of two different MHI. However there are some limitations, mainly due to the fact that only some classes of reference signals have been considered and non linear MP dynamics have been neglected. In this respect, the proposed results should only be considered as qualitative indicators of the likely performance of a MHI, and not as exact ones. Finally, some basic guidelines for controller design are presented, in order to enhance MHI performance.

REFERENCES

[1] C. Ramstein and V. Hayward, "The Pantograph: A Large Workspace Haptic Device For A Multi-Modal Human-Computer Interaction", *CHI'94, Conference on Human Factors in Computing Systems ACM/SIGCHI Companion-4/94*, pp. 57-58, 1994.

[2] L. Bouguila, M. Ishii and M. Sato, "Scaleable SPIDAR: A Haptic Interface For Human-Scale Virtual Environments", *Haptic Human-Computer Interaction*, pp. 182-193, 2000.

[3] M. Peshkin, J. E. Colgate, W. Wannasuphoprasit, C. Moore, B. Gillespie and P. Akella, "Cobot Architecture", *IEEE Transactions on Robotics and Automation*, 17(4):377-390, 2001.

[4] N. Nitzsche, U.D. Hanebeck, and G. Schmidt "Desgin Issues of Mobile haptic Interfaces", *Journal of Robotic Systems*, vol. 20:9, pp. 549-556, 2003.

[5] R.P. Darken, "Spatial orientation and wayfinding in large-scale virtual spaces II", *Presence 8* (1999), iii-vi.

[6] J. M. Hollerbach et al., "Simulating side slopes on locomotion interfaces using torso forces," *Haptic Symposium*, pp. 91-98, March 22-23, 2003.

[7] J. E. Colgate and J. M. Brown, "Factors Affecting the Z-Width of a Haptic Display", in *Proceedings of the IEEE International Conference on Robotics & Automation*, pp. 3205-10, San Diego, CA, May 1994.

[8] D. C. Ruspini, and K. Kolarov and O. Khatib, "The Haptic Display of Complex Graphical Environments", Siggraph97, pp. 345–352, 1997.

[9] D.A. Lawrence, L.Y. Pao, A. Salada and A.M. Dougherty, "Quantitative Experimental Analysis of Transparency and Stability in Haptic Interfaces", in *Proceedings of the ASME International Mechanical Engeneering Congress and Exhibition*, Atlanta, GA, Vol.58, pp 441-449, November, 1996.

[10] F. Barbagli, A. Formaglio, A. Giannitrapani, and D. Prattichizzo "An Experimental Study of the Limitations of Mobile Haptic Interfaces", in *Proceedings of International Symposium of Exeprimental Robotics (ISER2004)*, Singapore, 2004.

[11] F. Barbagli, A. Formaglio, A. Giannitrapani, and D. Prattichizzo "Controller design for a Mobile Haptic Interface", *Technical Report, Dipartimento di Ingegneria dell'Informazione, Universitá di Siena*, July 2004.