Haptic Interface Protocol for Complex Dynamic Virtual World

Shoichi HASEGAWA, Masahiro ISHII, Makoto SATO, and Yasuharu KOIKE
Precision and Intelligence Laboratory, Tokyo Institute of Technology
4259 Nagatsuta, Midoriku, Yokohama, 226-8503 Japan
TEL: +81-045-924-5050, FAX:+81-045-924-5016
Email: {hase, mishii, msato, koike}@pi.titech.ac.jp

This paper proposes a novel interprocess communication, between a virtual world manager with slow update rate and a haptic controller with fast update rate, which incorporates the integrated force given from the user to the world. High-speed refresh rate, generally 1kHz, is required to present stiff virtual objects that have spring-damper character with haptic display. On the other hand, the managing of an interactive and rich virtual world that takes a lot of computational time makes the refresh rate as low as 20Hz that is enough for CG. The dividing of the process into two has been proposed to interact with such a world and interprocess communication between the processes has been discussed. Although the methods proposed previously work well on condition that the virtual world is static, goes wrong on dynamic world. For instance, the movement of an object flicked by a user depends on the timing of sampling to measure the finger position, even though the finger moved on equal terms and the user felt the same haptic sensation. The importance of the force integration on the object motion acquisition is described and a novel interprocess communication that cares about the force integration is proposed to remove the influence of the sampling timing. The proposed haptic controller integrates the force given from user to virtual world during the cycle of the world manager, and sends the information on the integrated force to the world manager. The world manager updates the world with the information. A simple experimental result shows that the proposed methods works well on dynamic and complex virtual worlds.

1 Introduction

High-speed refresh rate generally 1kHz is required to present stiff virtual objects that have spring-damper character with haptic display. Lonnie and Wayne [1] reported that the update rate of 100Hz for 6.7kN/m stiffness or 1kHz for 70kN/m is necessary for stable operation. Chang and Colgate, using a haptic display, made a two-dimensional virtual world that consists of three fixed and four movable objects with 1kHz updating [2]. The cycle (getting finger position, updating virtual world, and giving haptic sensation) is managed in the same process. More complex virtual world that takes a lot of computational time must make the refresh rate low. The dividing of the process into two, to manage virtual world with low update rate and to control haptic display with high update rate, has been proposed to make more complex world, and interprocess communication between the world manager and the haptic controller has been discussed [3][4][5]. With these previous works, the world manager converts the local feature of the space near the user's finger into the intermediate representation (e. g. some planes or primitives) and sends them to the haptic controller. The haptic controller generates the force that acts on the user's finger based on the intermediate representation with 1kHz. Although this generally works well on condition that the virtual world is static, goes wrong on dynamic virtual world. This paper discusses the problem of the previous works and proposes a novel interprocess communication between virtual world manager and haptic controller, to make dynamic virtual world with haptic displays. A simple experimental result shows that the proposed methods works well on dynamic and complex virtual worlds.

2 Basic concept

2.1 Problem

Imagine that a finger flicks a movable virtual object floated in the virtual world. The movement of the object after the flicking depends on the timing of sampling to measure the finger position. The difference between Fig.1-A and Fig.1-B is sampling timing (phase). The finger moves on equal terms. The reflection forces, which act on the users, are same in both cases. In the case of Fig.1- A, the flicking is detected and the object moves. On the other hand, in the case of Fig.1-B, the flicking is not detected and the object does not move. We call the instability caused by timing difference of sampling "influence of sampling timing" in this paper.

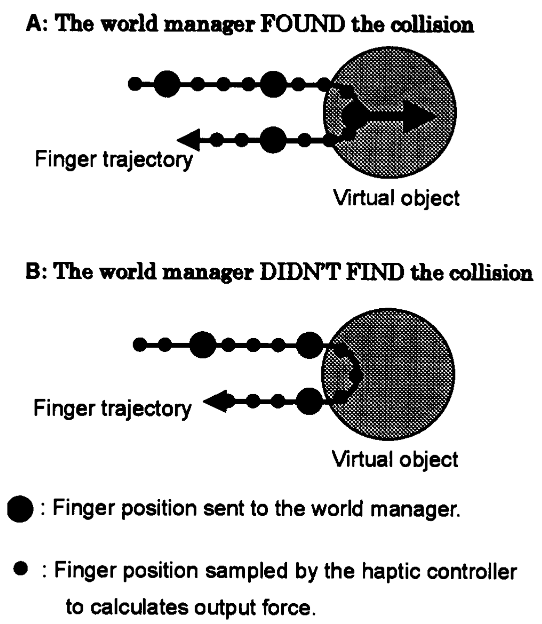

Figure 1: The influence of coarse sampling

2.2 Object motion and force

integration This section describes the role of the force integration to know the object motion.
Notation:
m: the mass of the object.
I: the matrix of inertia (3x3 matrix).
v: the velocity of the center of the gravity.
r: the position of the center of the gravity.
ω: the anguler velocity of the object.
f: the force acts on the object.
p: the postion where f acts on.
Dynamics of a rigid body is represented by:

$$m\frac{dv}{dt} = f \tag{1}$$

$$I\frac{d\omega}{dt} = (p - r) \times f \tag{2}$$

Integrate equation (1) from t to $t + \Delta t$:

$$\int_t^{t+\Delta t} m\frac{dv}{dt}dt = \int_t^{t+\Delta t} fdt$$

The velocity of the center of the gravity of the object at the next update is:

$$v(t + \Delta t) = v(t) + \frac{1}{m}\int_t^{t+dt} fdt$$

Integrate equation (2) from t to $t + \Delta t$:

$$\int_t^{t+\Delta t} I\frac{d\omega}{dt}dt = \int_t^{t+\Delta t} (p - r) \times fdt$$

In this case, Δt is small and p and r are considered to be constant.

$$\Delta\omega = I^{-1}((p - r) \times \int_t^{t+\Delta t} fdt)$$

The angular velocity of the object at the next update is:

$$\omega(t + \Delta t) = \omega(t) + I^{-1}((p - r) \times \int_t^{t+dt} fdt)$$

The motion of the objects depend on the integration of the force.

2.3 Approach to solution

As described in the previous section, each force is not important but the integrated force is important to acquire the motion of the object. To remove the influence of the sampling timing, we propose a novel interprocess communication, between the world manager and the haptic controller, which cares about the force integration. Fig.?? shows the difference between the previous and the proposed interprocess communication. The proposed haptic controller integrates the force given from user to virtual world during the cycle of the world manager, and sends the information on the integrated force to the world manager. The world manager updates the world with the information. The haptic controller measures the finger position, gives haptic sensation to the user, and integrates the force given from the user to the world with the frequency of 1kHz update. Fig.3 shows the effect of the proposed method on the motion of tapped object.

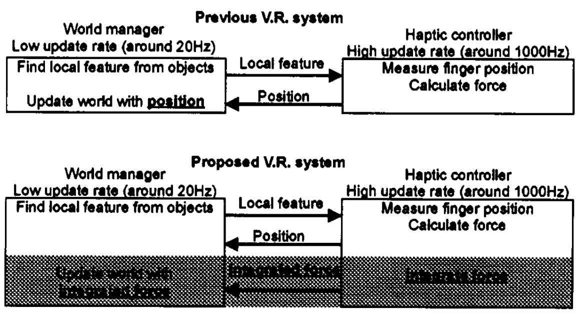

Figure 2: Difference between previous and proposed method

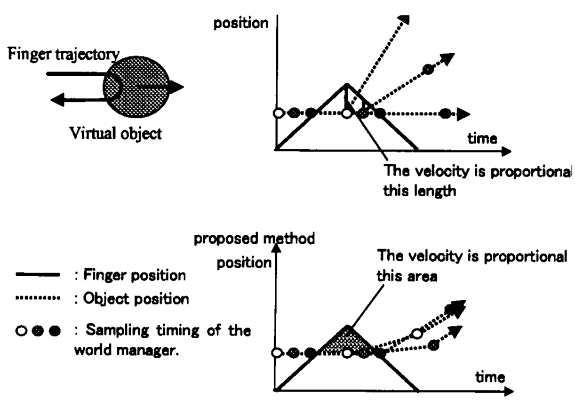

Figure 3: The effect of the proposed method

3 Detail of whole system

3.1 Interprocess communication

The following shows the whole procedure of our V.R. system (see Fig.4).

1. The haptic controller sends the information on the finger position to the world manager.
2. The world manager prepares representing planes (i. e. intermediate representation).
3. The world manager sends the information on the planes to the haptic controller.
4. The haptic controller shows interpolated planes that have spring-damper character to the user and integrates the force given from the user to the planes.
5. The haptic controller sends the information on the integrated force back to the world manager.
6. The world manager updates virtual world with the information on the integrated force.

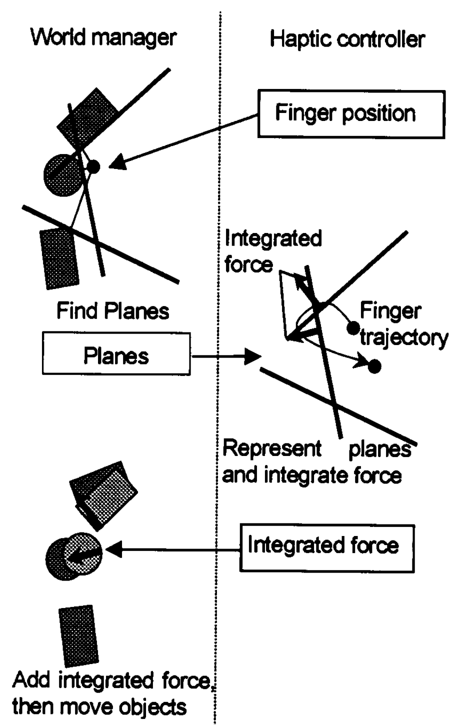

Figure 4: The influence of coarse sampling

3.2 Preparation of planes

The world manager prepares planes to inform the haptic controller. The following is the procedure.

1. Find objects within a certain radius sphere whose center is the finger.

2. On each object found by step 1, find the nearest point from the finger (see Fig.4).

3. For each object found by step 1, find the plane that goes through the point found by step 2 and faces to the finger (see Fig.4).

It is noted that the word "plane" means the pair of the point found at 2 and the normal found at 3.

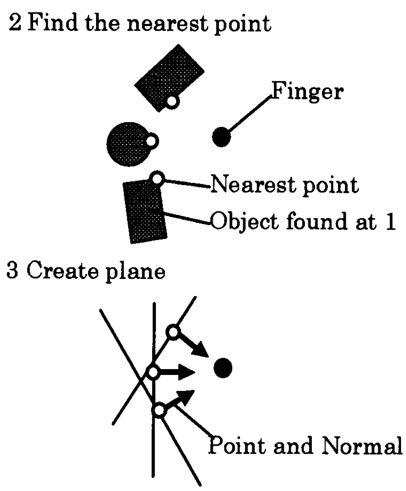

Figure 5: Preparation of planes

3.3 Interpolation of planes

The haptic controller shows user linearly- interpolated planes made from two planes that the world manager updated currently and previously. This makes continuous motion of the planes and gives natural feeling. If the haptic controller renews the plane crudely informed by the world manager, position and normal of the plane can suddenly change on a large scale and the surface can be felt chattered.

3.4 Integration of force

During the cycle of the world manager, the haptic controller integrates the force given from the user to the world. The amount of the force given by the user can be considered equivalent to the amount of the force generated by the haptic controller, and the direction of the forces is opposite. The procedure of the integration is shown in the following.
The haptic controller does:

1. Calculate output force with spring-damper model.

2. Add the current force given by the user onto the force integrated till then.

When the world manager updates, the haptic controller also dose:

1. Send the information on the integrated force to the world manager.

2. Reset the integration of the force to the value of the zero.

This procedure applies to each plane.

3.5 Procedures to update world

Notations:

Δt	: the period of haptic controller updates.
$T = k \cdot \Delta t$	: the period of world manager updates.
v_i	: velocity of the center of the gravity of the object #i.
ω_i	: angular velocity of the object #i.
m	: the mass of the object #i.
I	: the matrix of inertia of the object #i.
$P_i(t)$	: the position of the plane #i.
$N_i(t)$	: the normal of the plane #i.
$C(t)$	: the position of the finger.
F_i	: the integration of the force from user to the plane #i.
$f_i(t))$	: the force from the user to the plane #i.

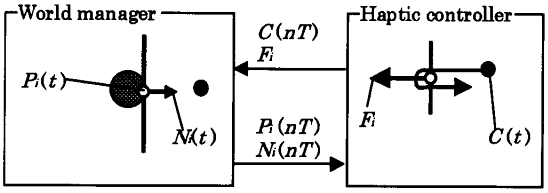

Figure 6: Notations

Procedure of the world manager:
Recieve $F_i, C(nT)$
For each object update velocity:
$v_i(nT) = v_i((n-1)T) + fraclmF_i,$
$\omega_i(nT) = \omega_i((n-1)T)$
$+I^{-1}(C(nT) - P_i((n-1)T)) \times F_i$
Update the position of all objects.
Find closest point of the object #i : $P_i(nT)$
For each object calculat normal:
$Ni(nT) = \frac{C(nT)-P_i(nT)}{|C(nT)-P_i(nT)|}$
Send $P_i(nT)$ and $N_i(nT)$)

Procedure of the haptic controller:
Recieve $P_i(nT), N_i(nT)$
For $t = nT$ to $t = (n+1)T - \Delta t$
Get the finger position : $C(t)$
Interpolate Plane:
$P_i(t) = \frac{t-nT}{T} P_i(nT)$
$+(1 - \frac{t-nT}{T}) P_i((n-1)T)$
$N_i(t) = \frac{t-nT}{T} N_i(nT)$
$+(1 - \frac{t-nT}{T}) N_i((n-1)T)$
Calculate force:
if $((C(t) - P_i(t)), N_i(t))N(t) > 0$ then
$f_i(t) = (((C(t) - P_i(t)), N_i(t))N_i(t)$
else $f_i(t) = 0$
Integrate force:
if $t = nT$ then $F_i = f_i(t)$
else $F_i := F_i + f_i(t)$
$t := t + \Delta t$
Send F_i to the haptic controller.

3.6 Simulation results

Since the previous method transmitted inexact integration of the force with large influences of sampling timing. Therefore, virtual objects accelerated much for the user's push in some cases, but did not move at all by pushing strongly in the other cases. A simulation that the user's finger flicked a virtual object was done to evaluate the difference of the effect of the interprocess communication on removing or existing of the influence of the sampling timing. In Fig.7, we applied only intermediate representation in the case of A, and we apply not only intermediate representation but also integration of the force in B. Both graphs showed five doted lines that represented the motion of the virtual object. The differences of these five dotted lines were the sampling timing (phase) of the world manager. The simulation result showed that our proposed method, in case of B, which transmits exact integration of force to the virtual world, had less influence of sampling timing compared to the previous method, in the case of A.

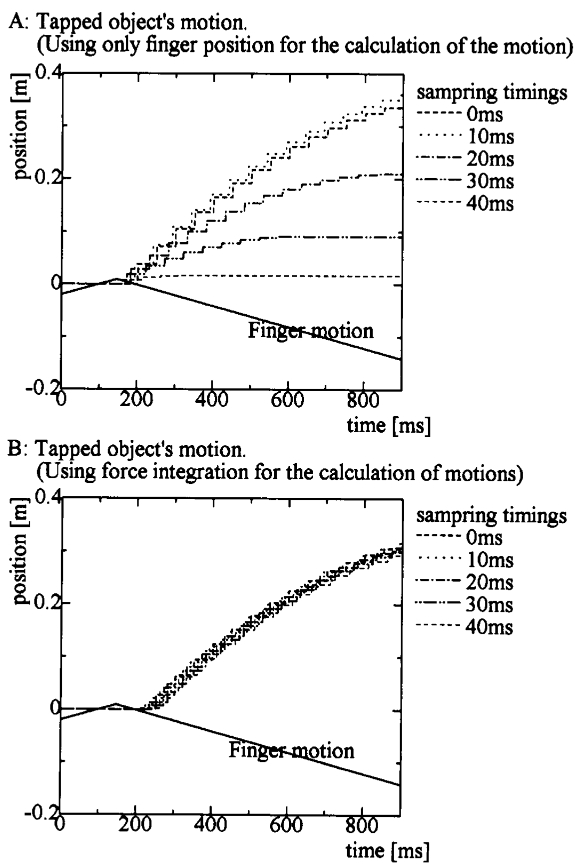

Figure 7: Simulation result

4 Experiment

4.1 Experimental system

An experimental system had been constructed. The haptic interface used here was SPIDAR [6] using tensed strings. A PC with Pentium-II 300MHz was used for managing world, controlling haptic interface, and rendering computer graphics. The update frequency of the world manager was 20Hz and that of the haptic controller was 1000Hz. The haptic controller showed 50 interpolated planes in a period of the world manager. The stiffness of the spring model was 2kN/m. The virtual world was constructed with a three-dimensional physical simulator; the gravity, aerial resistance, and collisions and frictions among the objects were taken in to consider.

4.2 Experiment and results

An experiment had been carried out and showed the effectiveness of the proposed method. The experimental task was as following. In the virtual world, a cube of 1kg was placed on a floor at the distance scale of 0cm (Fig.8). Subjects were asked to tap the cube once to move the cube to the distance scale of 30cm.

The experiment had to be done with two different (proposed and previous) methods of force calculation. Subjects completed the task with a certain method 30 times and with another method 30 times.

Fig.9 showed the position where the cube stopped. It showed that with the proposed method the subject could learn how to move the cube to the correct position. But with the previous method he could not. Table 1 showed the average of the square error of the stop position from the 10th trial to the 30th trial. The result showed that the proposed method obviously improved the correctness of the operation.

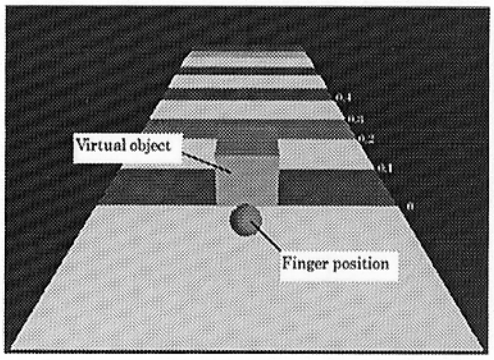

Figure 8: The pitcure of the experimental task.

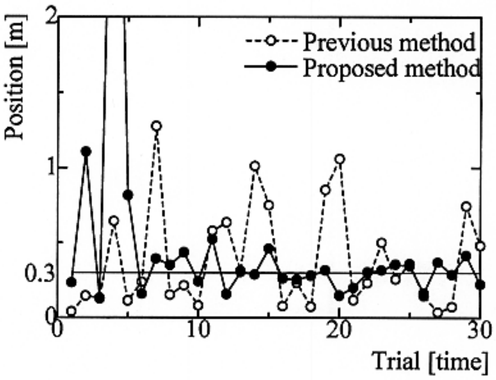

Figure 9: The position where the cube stopped.

5 Conclusion

A new scheme of haptic V.R. system was proposed. In this scheme, the haptic controller integrated the force from user to haptic interface and sent the integration of force to the world manager. It was shown that proposed method removes the influence of sampling timing. An experiment was made to check the effectiveness of the proposed method. Results showed that proposed method improve the correctness of the operation.

6 Future directions

6.1 Find the minimum update rate for the world manager

In the experiment, when the update rate of the world manager was dropped to 5Hz, the floor and spheres were felt larger than their actual size. This phenomenon occurred when world manager used only planes as an intermediate representation [3][4][5]. When the update rate of the world manager was dropped to 10Hz the graphics of the world

method	subject			
	A	B	C	D
previous	0.735	0.342	0.316	0.315
proposed	0.134	0.098	0.172	0.159

Table 1: The deviation error of the stop position [m]. ($\sqrt{\frac{1}{N}\sum(x-0.3)^2}$)

moved shakily, but the sense of the touch and push were tolerable. In this case, the graphics renderer had to interpolate the position of objects temporally like the haptic controller did to show the continuous motion of objects. The tolerable update rate of the world manager will depend on the virtual world. We would like to research the relation of the inertia of the virtual object and tolerable update rate.

6.2 Reduce the influence of the delay

In proposed method, the velocity of the plane in the haptic controller was constant in one period of updates of the world manager. We might need to change the velocity of the plane in the period to reduce the influence of the delay. So that the plane will have its velocity and mass. Then, the haptic controller calculates the velocity of the plane in its update. This method will also reduce the influence of a nonregular update rate of the world manager.

6.3 Representation of friction

The haptic controller interpolates the position of the plane. Because the position of the plane is a point of the collision between the finger and the object, the velocity of the position of the plane is different from the velocity of the object, which is expressed by the plane. Thus, the relative velocity from the object to the finger becomes necessary to calculate the frictional force. To represent friction, the world manager must send also the velocity of the object to the haptic controller. We had already loaded this idea and confirmed that the friction could be expressed.

6.4 Deformation of objects

There are some methods to deform a virtual object using haptic interface. These methods deform an object based on the overlap of the finger and the object[7]. The proposed method could be able to realize subtle deformations of objects by the deformation based on the integrated force. Experiment is to be carried out to support the idea.

References

[1] Lonnie Love and Wayne Book, "Contact Stability Analysis of Virtual Walls", Proc. of Dynamic Systems and Control Division ASME 1995, pp.689-694

[2] Beeling Chang and J. Edward Colgate, "Real-time impulse-based simulation for haptic display", Proc. of the 1997 ASME International Mechanical Engineering Congress and Exhibition

[3] Adachi Y., Kumano T., and Ogino K., "Intermediate Representation for Stiff Virtual Objects", Proc. IEEE Virtual Reality Annual International Symposium '95, pp.203-210

[4] William R. Mark, Scott C. Randolph, Mark Finch, James M. Van Verth, and Russell M. Taylor II, "Adding Force Feedback to Graphics Systems: Issue and Solutions", SIGGRAPH 96 Conference Proc. 1996 ACM

[5] Juli Yamashita, Cai Yi, and Yukio Fukui, "Feature-Based Haptic Rendering and Protocol", Proc. of the Virtual Reality Society of Japan Second Annual Conference, 1997, pp.195-197

[6] Masahiro Ishii and Makoto Sato, "A 3D Spatial pInterface Device Using Tensed Strings", Presence Vol. 3, No. 1, Winter 1994, pp.81-86

[7] Kenji YAMAMOTO, Akio ISHIGURO, and Yoshiki UCHIKAWA, "A Development of Dynamic Deforming Algorithm for 3D Shape Modeling with Generation of Interactive Force Sensation", Proc. 1993 IEEE Virtual Reality Annual International Symposium, pp.505-511