N90-20672

Prototype Part Task Trainer a Remote Manipulator System Simulator

David Shores
Barrios Technology Inc.
1331 Gemini
Houston, Texas 77058

Abstract

The Part Task Trainer program (PTT) is a kinematic simulation of the Remote Manipulator System (RMS) for the orbiter. The purpose of PTT is to supply a low cost man-in-the-loop simulator, allowing the student to learn operational procedures which then can be used in the more expensive full scale simulators. PTT will allow the crew members to work on their arm operation skills with out the need for other people running the simulation. The controlling algorithms for the arm were coded out of the Functional Subsystem Requirements Document to ensure realistic operation of the simulation. Relying on the hardware of the workstation to provide fast refresh rates for full shaded images allows the simulation to be run on small low cost stand alone work stations, removing the need to be tied into a multi-million dollar computer for the simulation. PTT is not intended to replace the full scale simulators but to augment the training process and reduce the load of the full scale simulators, especially when the student is learning new procedures and is error prone. PTT will allow the student to make errors which in the full scale mock up simulators might cause failures or damage hardware. On the screen the user is shown a graphical representation of the RMS control panel in the aft cockpit of the orbiter, along with a main view window and up to six trunion and guide windows. The dials drawn on the panel maybe turned using the dials on the dial box to select the desired mode of operation. The inputs controlling the arm are read from a chair with a Translational Hand Controller (THC) and a Rotational Hand Controller (RHC) attached to it.

INTRODUCTION

Part Task Trainer (PTT) is a kinematic simulator for the shuttle remote manipulator system(RMS). This paper will discuss what PTT does, it's history, uses, operation, design and the future of the program.

The controlling algorithms for the arm are coded from the functional subsystem software requirements document (FSSR) to ensure operation as close to the real arm as possible. Five of the computer supported modes and one of the non-computer supported modes are modeled. These modes supply the student with training in the major RMS modes of operation.

HISTORY

PTT started out as two separate programs on two separate machinès. The graphics were done on an IMI500 in wire frame and the simulation on an HP9000. When the Silicon Graphics 4D/60 was announced it was decided these two programs could be merged and provide better functionality. The controlling algorithms were coded from the FSSR and merged with already existing display code. This allowed us to deliver a limited working version in two months.

USES

PTT will be used in the training cycle for the crew members. It will provide inexpensive hands on training in an environment were mistakes can cause no damage to hardware. In the full scale simulator if the student makes a mistake damage to the equipment could be costly. But with PTT the worst damage only means restarting the simulator not rebuilding the hardware. PTT is not meant to replace the large scale simulators, but to augment them. The large scale simulators are expensive to run (computer time, support personnel), but PTT needs no support personnel, it is all self contained. It will allow the crew members more time to work with the arm and learn the different modes of operation. It will be used to maintain proficiency of operation, warm up for the integrated simulations and flight specific training. It is also used for engineering studies of reach limits and space station assembly.

OPERATION

PTT started out originally to be only a single joint operation simulator. But with the capability of the machine for floating point operations it was decided to include the computer supported modes. In single joint mode the operator is working with only one joint at a time. Therefor, the movements of the end effector will be an arc rather than a straight line as in the computer supported modes

In single joint mode the user selects a joint with the joint knob and inputs a positive or negative rotation with a toggle switch on the chair. There is no joint software reach limit checking done since this mode is used to drive the arm out of reach limits. In four of the computer supported modes (orbiter unloaded, end effector, orbiter loaded and payload) the translational hand controller (THC) and the rotational hand controller (RHC) are used to control the point of resolution (POR). The POR is the point about which the rotations are calculated, generally this is the tip of the end effector or a point located inside the payload. The translations translate the POR in a straight line along the axis of the coordinate system and the rotations are done about the POR. If the arm is in orbiter unloaded mode the coordinate system used is the orbiter's. In orbiter loaded mode the coordinate system is the orbiter's plus any offset added by the user for the POR. In end effector mode the coordinate system is the tip of the end effector. In payload mode the offset is added to the end effector position for the final POR. The THC provides positive and negative input on all three axis. The RHC provides positive and negative rotations for pitch, yaw and roll.

The last computer supported mode, operator commanded (OCAS), deals with the POR the same way as the other four. The difference is in the input for movement. In OCAS the user enters the position and attitude desired for the end effector. If it is a valid position and attitude, meaning the arm can reach it, the software attempts to drive the POR to this position and attitude in a straight line. The software does no checking for reach limits, singularities or interference when checking the final position and attitude. But reach limits and singularities are checked when the arm is being driven to the new position and if one occurs the user must deal with it. Interference between models is left up to the user just like the real RMS.

During the simulation the user has the option of practicing the grapple and release operation. When the grapple trigger is activated a list of possible grapple figures is checked to determine which one should be grappled. The grapple fixture must be within the constraints of the real arm, these are $-4 < [x,y,z] < 4$ and $-15 < [pitch, yaw, roll] < 15$. If the grapple is determined to be valid the arm is drawn to the grapple fixture and the payload is relinked dynamically to the arm. In other words the software determines the new position and attitude of the payload relative to the arm for the drawing hierarchy. This procedure can be reversed when a release is done. The new position and attitude relative to NULL is calculated for the payload and the hierarchy is changed to reflect this change. When the arm is going through a grapple or release sequence it takes approximimantly the same amount of time as the real arm does to help reduce negative training.

These are the basic modes of operation for the RMS. Now we will discuss the link to the graphics interface.

DESIGN

The interface to the simulator is a graphic representation of the control panel for the RMS in the aft cabin of the orbiter, an alpha-numeric terminal, a buttons and dials box, the mouse and a specially designed chair with an RHC and a THC attached to it. The lower left quarter of the screen contains the panel. This is used to indicate which mode of operation is active. The actual panel has three dials for the mode control. These dials have been mapped to the dials box. Movement of the dials is reflected by the dials on the screen. The other buttons and toggle switches on the panel which are needed for the simulator are mapped to the buttons box. The alpha-numeric terminal is used to simulate the auxiliary display in the aft cabin. Two of the possible screens have been modeled. The DISP94 and SPEC96 screens. These are useed for input for operator commanded mode and position and attitude information display. The buttons and dials box is used for moving the camera around in X, Y, and Z, manipulating the dials on the panel, changing the active camera, and the switch input for the panel. The mouse is used for the toggling between wire frame and shaded views, and enlarging any of the view windows up to full screen or back. Also, in setup mode it is used to adjust models, cameras, lights and other operational information. The chair with the RHC and THC is used for arm control input. The chair communicates with the simulation over an RS-232 line.

The top left quarter of the screen is the main view window. This window contains the view from the active camera. The active camera can be changed to preset camera positions with the buttons or moved around in X, Y, and Z with the dials. The right side of the screen is used for special purpose windows. These windows are views of the trunions on the payload and the associated guides in the payload bay. These windows provide an unseeable view in the real world. They assist the student when doing single joint operations.

There are two types of models used in PTT. Those predefined by the software, the panel, and those built by another program and read in, the orbiter. 98% of the models read into PTT were created with the in house model building package Solid Surface Modeler (SSM).

The predefined models are the models used to draw the panel. These were hand designed and placed on the screen. Only the parts of the panel which change are updated. If the parameter dial is turned the parameter dial on the display will change as well as the number readout, but nothing else is updated. The models read into PTT are drawn in the view window. All of the models are update in the main view every time. Each of the special purpose windows has a list of models associated with them, if any of the models in the list are moved then the window is updated. Otherwise it is left unchanged.

The models read in are linked together in an hierarchy which tells the program where to draw each model. Each node in the hierarchy has a flag which is set if the position and attitude change. If so, every node that is a child to this node must be redrawn.

One of the design goals deals with the speed of updating the screen. Only drawing the models which move allows the graphics engine to do as little work as possible when updating the screen. Another design goal was to minimize negative training. Since mistakes on orbit can be costly or even dangerous all training is done as close to the actual procedure used in flight as possible for consistency. Some examples of this are labeling the dials and buttons on top instead of underneath. All the switches and knobs in the orbiter are labeled on top. Also the length of time is takes the grapple/release sequence. Since the arm can be moved with this operation is taking place the time in the simulator is approximantly the same amount used for the real arm so the user does not get in the habit of moving the arm to soon.

The justification for PTT is simple. Most of the code was already written but used in different programs. By using this code the maintenance of the code is relatively easy. It also means enhancements to the code are just as easy. The cost of operation is minimal. Once the student has an introduction course there should be no more need for instructors. Also the cost of the machine is small to the cost of the large simulators.

The future of PTT looks promising. It will go into the training cycle in April. So far everybody dealing with training who has seen PTT has liked it and are anxious to get it into the training cycle. As for program enhancements another view window and dynamics have been discussed. We are hoping to get a Silicon Graphics 240GTX which is a 4 processor parallel machine. We feel these enhancements would greatly improve the ability of the simulator. We also have several different versions of PTT. One allows the student to work with a two arm configuration. Another version of PTT is being developed for the Space Station Freedom arm.

With the ease of use, ease of modifications and speed of the simulator PTT should be very useful for training, maintaining proficiency and engineering studies.