PAPER AAS 93- 750

AMERICAN ASTRONAUTICAL SOCIETY

AIAA

Numerical Stability Problem Encountered Modeling Large Liquid Mass in Micro Gravity

Noel H. Hughes

The Boeing Company

AAS/AIAA Astrodynamics Specialist Conference

VICTORIA, B.C., CANADA AUGUST 16-19, 1993

AAS Publications Office, P.O. Box 28130, San Diego, CA 92198

Numerical Stability Problem Encountered Modeling Large Liquid Mass in Micro-Gravity

Noel Hughes

INTRODUCTION

Prior to program cancellation, the high energy upper stage vehicle Shuttle-Centaur was to have made its first flight aboard the Space Transportation System (Space Shuttle) in June of 1986. Typical mission profiles specified durations of a few hours to a few days of on-orbit coast prior to Shuttle/Centaur separation. Prior to separation, the Centaur and its payload were to be elevated to a nominal 45 degree position relative to the Shuttle longitudinal axis. The vehicle restraint was then to be released and Centaur propelled from its support structure by twelve preloaded springs.

Inclusion of an explicit fluid dynamic model of on-board liquids in the Shuttle/Centaur Separation model was necessitated by the dramatic effects of the liquids on vehicle trajectory during separation and the inadequacy of previous analytical techniques.

Galileo, which was to be the first Shuttle/Centaur payload, had a total vehicle mass, including liquids, of approximately 60,000 pounds; of this approximately 38,000 pounds was liquid oxygen (LOX). The ullage, or void space, in the LOX tank was approximately 5% of the tank volume. Thus, the LOX had sufficient void space to move within the tank and sufficient mass to dominate the inertial response of the vehicle. Without inclusion of the effects of motion of the liquid mass within the tank, any dynamic model of the Centaur vehicle would have little validity.

Traditional modeling of liquids in launch vehicles utilizes the pendulum analogy (ref. 1) in which an equivalent pendulum's mass, length and attach point are calculated from the known geometry of the liquids in the tank. This technique depends on the system being subjected to a relatively constant acceleration or gravity field, and thus, is not valid under conditions encountered during separation, ie, micro-gravity to, perhaps, one tenth gravity and back to micro-gravity within the space of approximately one half second. Thus, use of the pendulum analogy was rejected for the separation analysis.

The first attempt at using a computational fluid dynamics code, HYDR-3D, to model on-board liquids during separation was in a stand alone mode. The (previously existing) Shuttle/Centaur Separation Simulation was run with a reduced liquid mass treated as a rigid part of the vehicle, producing a vehicle trajectory. Using HYDR-3D, this trajectory was used to excite a model of the Centaur tank containing the liquid propellants, producing a set of resultant forces. These forces were put back into the separation simulation in another run. This process was iterated until the motions and forces converged to a stable result. This iteration process was typically consuming, among other things, one and one half to two weeks to complete and did not always converge. This was considered unacceptable in view of the large number of scenarios to be analyzed. The logical solution was to combine the two programs, the existing Shuttle/Centaur Separation model and HYDR-3D, into one. This paper details the development of the algorithm that was used to combine the two programs.

HYDR-3D AND PREVIOUS VEHICLE MODELING

HYDR-3D (ref. 2) is a computer program for predicting the behavior of fluids in three dimensions developed by Flow Sciences, Inc. of Los Alamos, New Mexico. It utilizes the finite difference technique, using pressure and velocity as the primary independent variables.

Rocketdyne (ref. 3) had previously incorporated HYDR-3D into its coast phase control dynamics model of the Peacekeeper upper stage. In their approach, both HYDR-3D and the vehicle dynamic model integrated their respective state variables forward in time, passing hydrodynamic forces and moments from HYDR-3D to the vehicle model and passing vehicle accelerations and rotational velocities from the vehicle model back to HYDR-3D. Significantly, the on board liquid mass was less than 25 per cent of the total vehicle mass.

SHUTTLE/CENTAUR SEPARATION RIGID BODY MODELING

Separation of the Centaur/payload upper stage vehicle from the Space Shuttle cargo bay, at an elevation angle of 45 degrees from the Shuttle center line, presented a unique modeling problem. The upper stage represented as much as ten percent of the combined mass of the upper stage-orbiter combined vehicle. A two mass rigid body separation model, SEPAR, was developed early on in the Shuttle/Centaur program and was used for initial engineering studies.

To merge the rigid body model with HYDR-3D, the main unit, program SEPAR, was downgraded to a subroutine. The fixed time integration step size, previously' used by SEPAR, was replaced by a variable step size, dictated by the model time between calls from HYDR-3D. Two subroutines were added to convert HYDR-3D calculated forces and moments to the Centaur vehicle coordinate system; the other calculated the linear and rotational motions that the propellant tank experienced due to the rigid body motion of the vehicle.

INITIAL INTERFACE TECHNIQUE

The first attempt at interfacing HYDR-3D to the Separation Model used the technique that had been successfully employed by Rocketdyne when it incorporated HYDR-3D into its Peacekeeper simulation. This technique merely fed forces from HYDR-3D into the Centaur dynamic model to be used in calculating accelerations at any time step. The accelerations thus calculated would be integrated forward in time and then handed back to HYDR-3D, along with the resultant rotational velocities, to produce another set of forces. This technique was expected to produce a start up oscillatory transient and so various attempts were made at calculating realistic initial conditions for the hydrodynamic forces.

Testing of this technique revealed that it was unstable; the forces, moments and accelerations oscillated with growing amplitude until values were calculated that overflowed the sixty bit word length of the host computer. No set of initial conditions significantly affected this rather prompt oscillatory divergence. Moreover, reduction of time integration step size, the traditional method of stabilizing integration instability, was found to be useless in correcting the divergence. In fact, the frequency of oscillation and the rapidity of divergence increased with decreasing step size; the divergence appeared to be a function of the number of integration steps, rather than of the time integrated through.

An attempt was made to stabilize the system by reducing the time integration step size to zero, ie, preventing time integration at any iteration. The problem was thereby essentially recast in the form

$$\mathbf{F}_{n+1} = f(\mathbf{F}_n) \qquad (1)$$

where **F** is the vector of calculated forces and moments, and the function f is the process of executing first the Separation portion of the code to produce accelerations and then the HYDR-3D portion with these accelerations to produce a new set of forces. Fixed point iteration (ref 4), as shown in fig 1, was then used in an attempt to converge on the required value of forces. Convergence would be reached when the quantity $\mathbf{F}_{n+1} - \mathbf{F}_n$ became smaller than some specified error limit; the integration routines would then be given a non-zero time step and allowed to move forward in time. This fixed point iteration process performed no better than the first time integration technique, diverging about as rapidly as the first technique.

After investigation by the author and Flow Sciences, Jim Sicillian of Flow Sciences (ref 5) pointed out that, indeed, the techniques used thus far were inherently unstable for this application. A very simple analysis, detailed in appendix A, showed why the technique that had been used successfully by Rocketdyne was unstable for the present application. The critical difference between Rocketdyne's and the current analysis was the difference in the masses of on board liquid relative to the mass of the dry vehicle. Sicilian's analysis showed that when the liquid mass was greater than the mass of the dry vehicle, fixed point iteration, or any similar convergence technique would be unstable.

STABILIZATION OF INTERFACE

With an understanding of the cause of the instability (and the attendant confidence that the fault lay in the algorithm itself, not in its implementation), various techniques for stabilizing the interface were investigated.

The first technique involved under relaxation of successive calculations of forces at each time step. Fixed point iteration, similar to that described above, was used with successive estimates of force calculated from

$$\mathbf{F}_{n+1} = f(\mathbf{F}_n) + k(\mathbf{F}_n - f(\mathbf{F}_n)) \qquad (2)$$

where $0. < k < 1.0$ and f is the same function described in eq (1). This technique was also found to be unstable, although the envelope of the divergence appeared to be somewhat sensitive to the value of k used.

The next technique tried was successful in stabilizing the interface and, with minor modifications, was the one finally implemented. This technique defined a new function,

$$G(\mathbf{F}_n) = \mathbf{F}_{n+1} - \mathbf{F}_n \qquad (3)$$

where $\mathbf{F}_n$ was calculated from eq (2), and sought an approximation to the zero of this function. When a value of F_n was found which produced a value of $G(\mathbf{F}_n)$ less than a specified error tolerance, the solution had converged. Several methods of finding the zeroes were investigated; the one finally used was the "Regula Falsi" technique, taken from (ref 4). In this technique, three successive estimates of forces were calculated using eq (2) with $k = .5$. $G(\mathbf{F})$ was tested for convergence after each new estimate was calculated and the procedure terminated if successful convergence was achieved. If convergence was not achieved after three estimates, a new estimate was made from

$$\mathbf{F}_{new} = \mathbf{F}_n - G(\mathbf{F}_n) * (\mathbf{F}_{n+1} - \mathbf{F}_n)/(G(\mathbf{F}_{n+1}) - G(\mathbf{F}_n)) \qquad (4)$$

where

$$(G(\mathbf{F}_{n+1}) - G(\mathbf{F}_n))/(\mathbf{F}_{n+1} - \mathbf{F}_n)$$

was effectively the slope of the secant of function $G(\mathbf{F})$ from $\mathbf{F}_n$ to $\mathbf{F}_{n+1}$ (see fig 2). The value $\mathbf{F}_{new}$ was then tested for convergence against $\mathbf{F}_n$. If convergence had been achieved, the forces, $\mathbf{F}_{new}$ were handed to the rigid body separation model which calculated a set of motions with these forces; HYDR-3D then was allowed to integrate forward one time frame with these motions, producing a set of forces. These forces were handed to the rigid body model which integrated forward in time to "catch up with" HYDR-3D. If convergence had not been reached, the process was begun again with Fnew used for the latest value of $\mathbf{F}_n$.

IMPLEMENTATION

In implementing the interface algorithm and using the combined program on various computer systems, a number of special features were added to the interface to improve performance or resolve problems.

Symmetry considerations within HYDR-3D required that the tank/fluid model be generated with z the axis of tank symmetry. All communications between HYDR-3D and the rigid body model were in the HYDR-3D coordinate system axis orientation; units of communicated variables were those of the rigid body model. Conversion factors were included in the HYDR-3D input file. The rigid body model performed the necessary coordinate system rotation. Included in the coordinate transformation was the capability to rotate the liquid/tank model around the Centaur x, or roll axis (ie, the tank z axis). This allowed the use of a single axially asymmetric tank/fluid model in the analysis of multiple initial ullage position scenarios. This rotation is controlled by the input parameter AROT. With AROT = 0.0, the HYDR-3D y axis was transformed to the Centaur z axis, the HYDR-3D x axis to the Centaur y axis. The rotation controlled by AROT followed the right hand rule around the Centaur x axis, ie, a positive AROT value rotated the HYDR-3D x axis from the Centaur y axis toward the Centaur z axis. Figure 3 shows the relationship between the two coordinate systems.

Because of the nature of the finite difference technique used in HYDR-3D, running the combined program consumed large amounts of computer memory and time. To reduce this consumption, the interface was designed to run in either a three or six degree of freedom mode. In the six degree of freedom mode all six components (three rotational and three translational) of forces, moments and motions are transmitted between HYDR-3D and the rigid body model. In the three degree of freedom model, only two translational and one rotational degree of freedom were communicated, x and z translation and y rotation. X and z moments, rotational velocities and rotational accelerations, y force and linear acceleration were all set to zero by HYDR-3D. This was deemed acceptable because of the minimal coupling between pitch and yaw motions of the Centaur vehicle. Use of the three degree mode required that the tank/fluid model be cut symmetrically in half with the cutting plane normal to the HYDR-3D y axis. This reduced by half the number of fluid cells in the model, and thus the amount of memory consumed, and significantly reduced the computation time. Since forces and moments were calculated using half of the actual fluid, they were doubled before being transmitted to the rigid body model.

A number of features were built into the interface, necessitated by the less than perfect performance of the Regula Falsi convergence technique. It was found that this technique performed very poorly when the slope of the secant calculated in equation (4) was small in absolute value. When this condition is encountered, new values of **F** were derived, away from the area of small slope, and the Regula Falsi process was begun again.

Two types of convergence error tolerance were used, one based on a percentage of **F**, the other based on the absolute value of **F**. Testing based on absolute value of **F** was required because, during part of the Shuttle/Centaur separation process, the vehicle was, essentially, unaccelerated and the correct value of **F** would be very close to zero. Testing for convergence based on a percentage of the calculated value would, thus, be unworkable.

Because HYDR-3D would occasionally produce anomalous spikes in forces calculated, it was deemed acceptable to loosen up convergence tolerances at a particular time frame. After a certain number of attempts at convergence had been made in any time frame, the convergence criterion was progressively loosened until either convergence or a maximum number of iterations was reached. Following convergence and integration forward in time, the original convergence criterion was restored.

VALIDATION

As is necessary in the development of any piece of software, the problem of validating the results of the combined HYDR-3D - Shuttle Centaur model had to be addressed. Ideally, results of a model are compared with the real world performance of the system modeled; if real world data are not available, comparison of model results with results derived from an alternative analysis can be used.

Prior to the present analysis, HYDR-3D and the rigid body Shuttle/Centaur model had been validated separately (ref 6) and (ref 7). The combined program was validated against the two separate programs, ie, the rigid body model and stand alone HYDRA-3D: a run of the combined program was made, saving the forces calculated by HYDR-3D and the motions from the rigid body model. Both programs were then run separately, HYDR-3D being driven by these motions and the rigid body model driven by the forces and moments. The results of this process were compared with the results from the original combined run; agreement was very good. Results from several combined runs were also compared with two runs made using the stand alone iteration process described above. Agreement was also very good in this case.

RESULTS

Several parametric and failure mode studies were conducted using the combined Shuttle/Centaur - HYDR-3D program. These studies examined the effects on Centaur trajectory of parameters such as initial ullage location, variation of disconnect forces and separation guide gap. Failure modes analyzed included, among others, single/multiple separation spring failures, over pressurized fuel dump lines and less than nominal pre-separation Centaur elevation. These studies were aimed at establishing an envelope of conditions under which the Centaur could separate safely from the Shuttle.

Several hundred runs of the combined program were made in the course of these studies. With the computing resources available at the time, as many as three to five runs could be accomplished in a twenty four hour period.

DISCUSSION AND RECOMMENDATIONS

The current convergence algorithm assumes that the relationship between vehicle acceleration and hydrodynamic forces is linear at any point in time. This is a good assumption except in so far as linear and rotational degrees of freedom are coupled. This coupling arises because the on board liquid is not located at the center of gravity of the vehicle, and linear hydrodynamic forces generate both linear and rotational accelerations and vice-versa. The current algorithm ignores this coupling and essentially "brute forces" its way to a converged solution at each time step.

Experience using the combined program has suggested several approaches to improvement of the interface algorithm. These approaches fall generally into two categories: a) accelerating convergence of the current algorithm and b) alternate approaches.

One possible approach to convergence acceleration is to use old information in the calculation of Regula Falsi solution estimates. As noted, above, when a Regula Falsi estimate, $\mathbf{F}_{new}$, has failed to pass the convergence test, it is used as a seed to calculate two more new values of **F** to produce two new values of G(**F**) and all previously calculated values of **F** and G(**F**) are discarded. If, instead, a decision were made to determine which of the old **F** and corresponding G(**F**) values was better, only the other could be discarded and replaced by $\mathbf{F}_{new}$ and $G(\mathbf{F}_{new})$. This would save one **F** calculation for each Regula Falsi estimate. If this did not, in general, increase the number of Regula Falsi estimates required to reach convergence, significant computation would be saved.

An alternate approach to interfacing the two codes would be to calculate an effective mass matrix from HYDR-3D. This would involve calculating a 6 x 6 partial derivative matrix, **J**, where

$$j_{ij} = \partial f_i/\partial a_j$$

and

f_i is the ith component of **F** and a_j is the jth component of the six component vector of linear and rotational accelerations, **A**. This could be calculated by, starting with an estimate of forces at any time step, perturbing each component of **A**, one at a time, and calculating the change in the six components of hydrodynamic force, **F**, caused by each of these perturbations. This effective mass matrix would then be handed to the rigid body model to be added into the real inertial characteristics of the dry vehicle to calculate the resultant accelerations. This process would require at least six passes through HYDR-3D for each time frame calculation which might increase the total computation time required. However, it might be possible to use the same matrix for several time frames, thereby reducing the amount of computation required.

CONCLUSION

The computational fluid dynamics code, HYDR-3D, was merged with the rigid body Shuttle/Centaur Separation model. This combined program was used to simulate vehicle behavior, including the effects of on board liquid propellants, during separation of the Centaur from the STS orbiter. Unique problems concerning the numerical stability of the interface between the two programs arose, due to the large mass of the liquids relative to the mass of the dry Centaur. Several unsuccessful approaches to solving this instability were tried. The eventual solution involved defining a new function which was the difference between successive calculations of forces and moments and approximating the zero value of this function. Various failure mode and parametric studies were performed utilizing the combined program.

REFERENCES

1. Lukens, D.R., Schmitt, A.F., Broucek, G.T.: Approximate Transfer Functions for Flexible Booster and Autopilot Analysis, Convair Report no. AE61-0198.

2. Sicilian,J.M., Hirt,C.W.: "HYDR-3D - A Solution Algorithm for Transient 3D Flows", Flow Sciences, Inc. August, 1984.

3. Eastes, Theodore, Personal Communication, GD/DSD - Rocketdyne, June,l 985

4. Conte, S.D., de Boor, C.: "Elementary Numerical Analysis, an Algorithmic Approach", McGraw-Hill Book Company, New York, 1972.

5. Sicilian, J.M., Personal Communication, GD/DSD - FlowSciences, June, 1985.

6. Steuber, G.D.: Simulation of LERC Drop Tower Tests of Centaur L02 Tank Using Hydr-3D, GD/SSD TM 696-85-754, Dec 17,1985

7. Engbrenghof,L.C.: Shuttle/Centaur Separation System 3-D Simulation, GD/SSD Report P5930

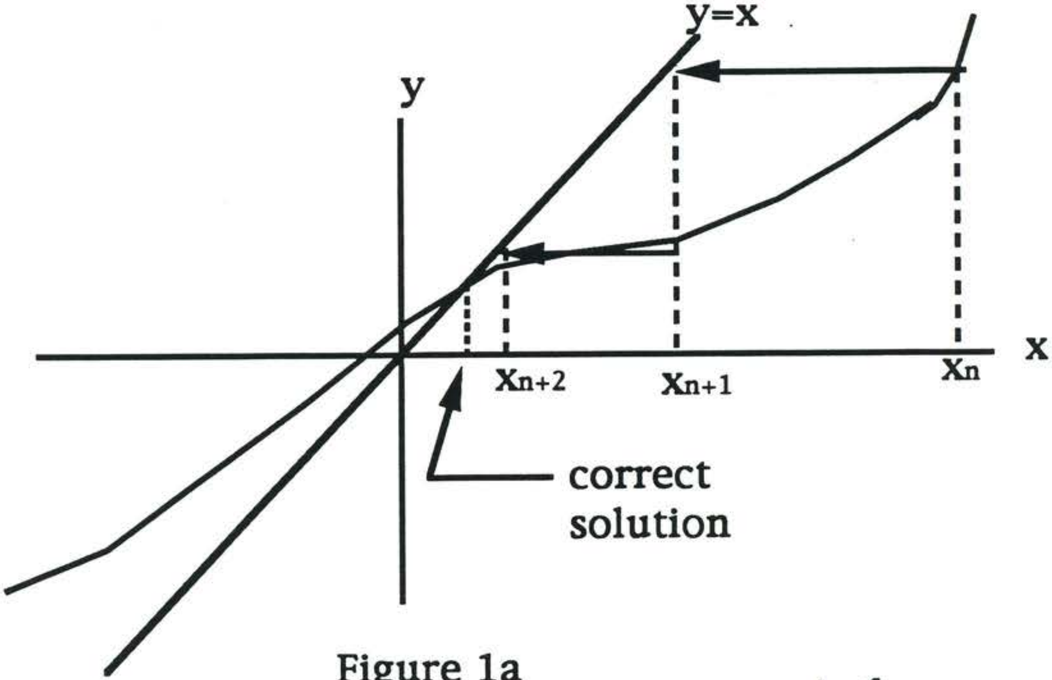


Figure 1a
(stable)

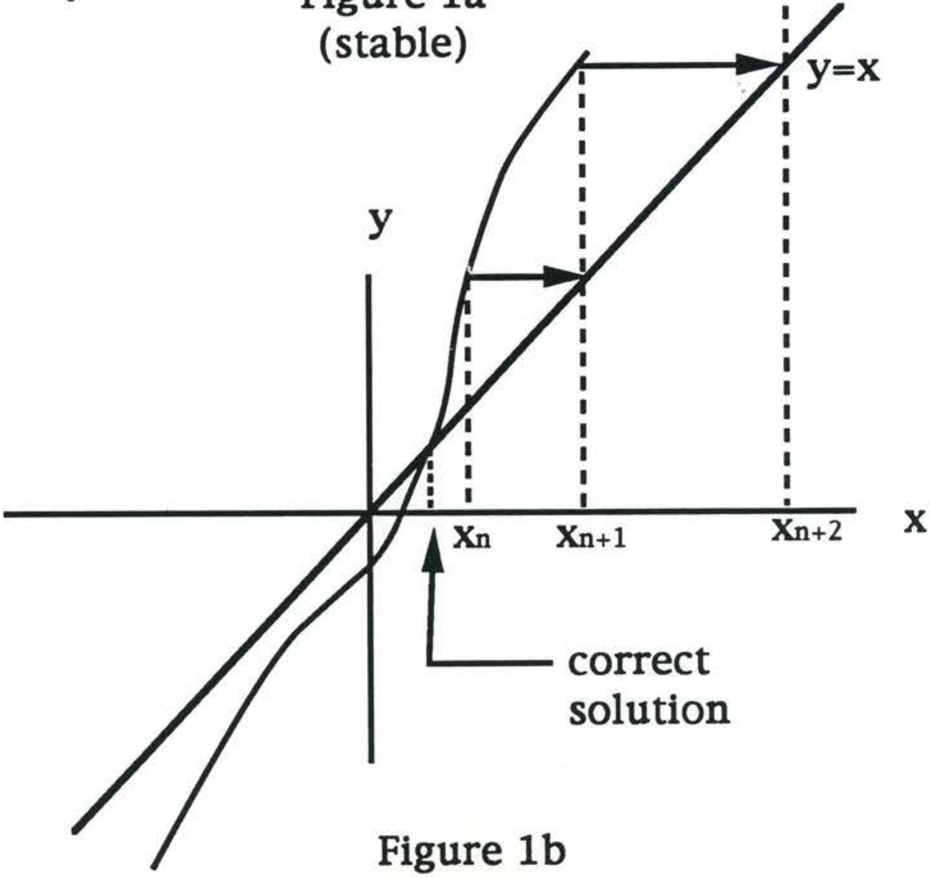


Figure 1b
(unstable)

Stable and unstable behavior of Fixed Point Iteration

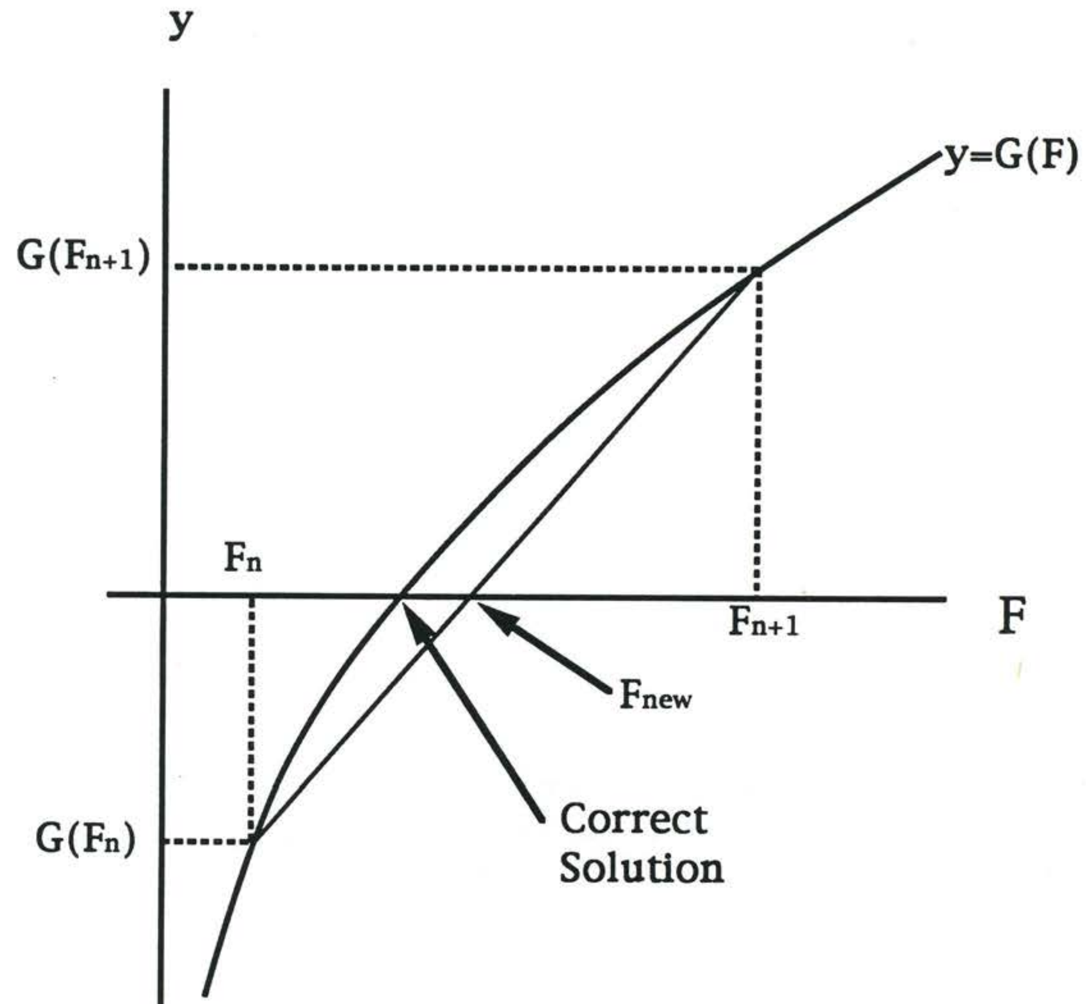
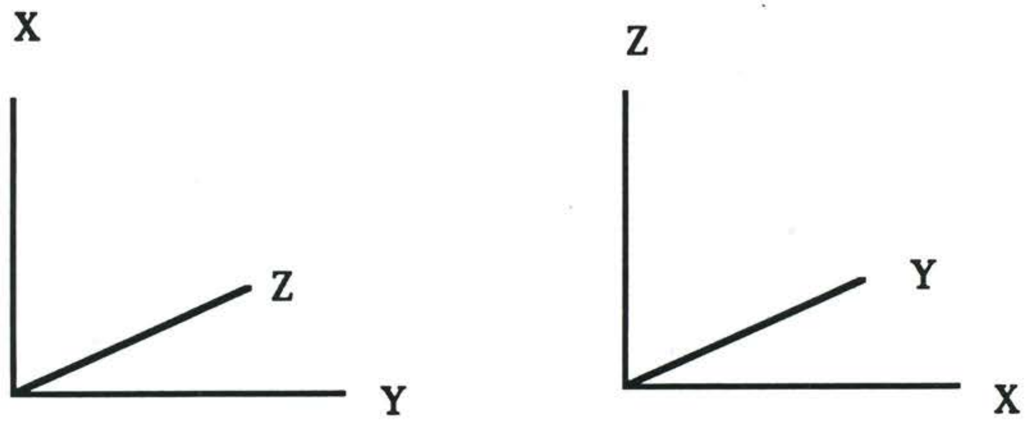


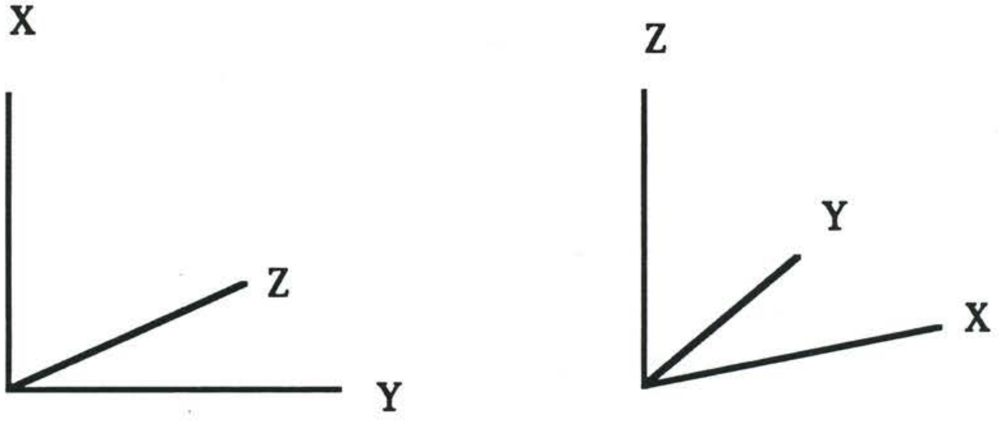
Figure 2
Convergence Process for
Regula Falsi



Centaur

HDR-3D
Tank/Fluid Model

Figure 3a
Coordinate System Orientation
(AROT=0.0)



Centaur

HDR-3D
Tank/Fluid Model

Figure 3b
Coordinate System Orientation
(AROT > 0.0)

APPENDIX A:
Stability Analysis for the HYDR-3D/Rigid Body Model

Refering to fig A-1 (**n**) refers to quantity calculated during the nth iteration

$$\ddot{Z}(n+1) = \frac{F_{ex}(n) - F_h(n)}{M_v}$$

using: $$F_h(n) = \ddot{Z}(n)\, M_L$$

$$\ddot{Z}(n+1) = \frac{F_{ex}(n) - \ddot{Z}(n)\, M_L}{M_v}$$

$$= \frac{F_{ex}(n)}{M_v} - \frac{M_L}{M_v}\ddot{Z}(n)$$

rearranging:

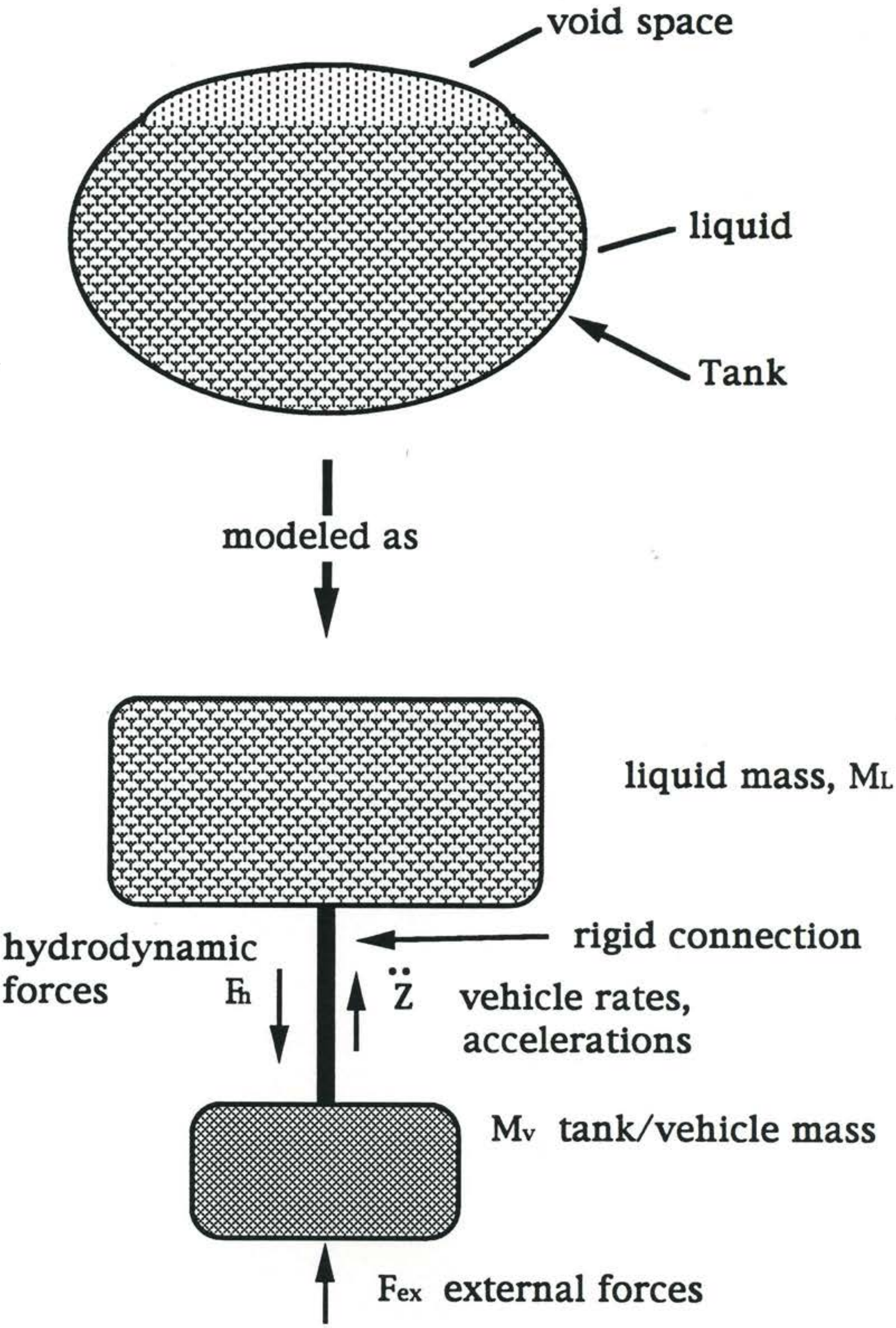
$$\ddot{Z}(n+1) + \frac{M_L}{M_v}\ddot{Z}(n) = \frac{F_{ex}(n)}{M_v}$$

or, using Z transform notation:

$$\ddot{Z}\left(1 + \frac{M_L}{M_v} z^{-1}\right) = \frac{F_{ex}}{M_v}$$

$$\ddot{Z} = \frac{\frac{F_{ex}}{M_v}}{1 + \frac{M_L}{M_v} z^{-1}}$$

fig A-1 Conceptual HYDR-3D/Vehicle Model



poles of this function occur when

$$1 + \frac{M_L}{M_v} z^{-1} = 0$$

solving for z:

$$z = \frac{M_L}{M_v}$$

z lies outside the unit circle (ie, system is unstable) when:

$$\frac{M_L}{M_v} > 1.0$$