DETERMINISTIC STUDIES OF DEBRIS HAZARDS WITH PARALLEL PROCESSORS

Liam M. Healy and Shannon L. Coffey
Code 8233
Naval Research Laboratory
Washington, DC 20375-5355 USA

Abstract

A new generation of parallel processing computers makes possible the ability to propagate all objects in the space surveillance catalog with simulated objects, and detect close approaches. With this capability, it is possible to test deterministically debris scenarios, without resorting to statistical models. To compare the positions of objects we have developed two methods, an all-to-all comparison and a one-to-all comparison. For the former, a sieve significantly reduces computation time; for the latter, direct comparison is possible in parallel. We show results from several simulations, including simulated multiple sources of debris, hazard to the space station, and close contacts amongst the catalog itself, to show potential for debris studies. The techniques described here have potential application to the general problem of catalog maintenance.

1 Introduction

Much of the current literature on debris studies is based on statistical models of debris and their evolution. We propose a different attack: with a parallel processor it is possible to conceive of propagating and detecting close conjunctions not only simulated debris but the actual catalog of orbiting objects kept by the space surveillance centers. The propagation would be with one of the accepted analytical models, numerical integration, or a new propagator.

By deterministic studies, we mean studies in which propagation is performed analytically or numerically for each object based on that object's phase space coordinates, as opposed to statistically based on some debris model. That is, assuming an initial position and velocity of an object, its future position and velocity are determined by application of exact or approximate equations of motion. For actual objects, the observed positions and velocities may be used. For simulated debris, we may pick some model to represent the initial velocities, and the actual position of the disintegrating object at the time of fragmentation. From this point, all objects positions and velocities are computed using whatever propagator is appropriate.

Ordinarily, this would be too difficult and time consuming to contemplate doing for any realistic debris simulation. With a catalog of about 7000 objects currently, and perhaps many more in the near future, deterministic propagation, even without any simulated debris is a daunting task on a serial computer. It is a natural, however, for a "data-parallel" model of computing. In this case, for n satellites, there are n independent virtual processors, each doing a serial calculation for its own satellite or fragment. This paper describes one such computer program, on the Connection Machine (CM), and some results with debris simulations.

As it stands now, there is a program COMBO (Calculation of Miss distance Between Objects) run by the surveillance centers on serial machines that, in conjunction with a propagator or propagators, will find all potential hazards from the catalog to a particular object; this we call the "one-to-all" problem. What we describe here goes further: for a large catalog, what is the mutual threat, that is, what are the close objects in an "all-to-all" comparison? Such a task would not be contemplated on current serial processors. Alternatively, we might want to know the close contacts between two fairly large sets, such as the existing catalog and a simulated debris cloud. Again, this is effectively impossible on serial machines.

For want of a better name we coined our program COMBO also. However, it represents but one experiment in using data-parallel processors for space surveillance. The ultimate goal of this investigation is much more ambitious. We seek to determine the extent that parallel processors may perform other space surveillance functions. It is clear from debris studies and the coming of permanent manned space stations like Freedom, that better computing performance will be necessary to catalog very small objects of debris.

2 Propagation

There are several so-called "analytic" propagators currently in use on serial computers by the space surveillance centers. Prominent among these are PPT2 and SGP4. Both of these are based on Brouwer's [1] theory for averaging of the dynamics induced by an object orbiting a planet with a non-spherical mass distribution, such as the earth. These propagators give quite good results, typically a few kilometers over twenty-four hours. However, for very close encounters, this may not be enough. In that case, one might resort to numerical integration, but only for the very pair that is potentially on a collision course. With parallel processing, we can contemplate doing much more numerical integration than we might on a serial machine.

3 COMBO

The computation of close miss-distances involves finding all pairs of satellites in a set that are within some specified critical distance c. To find close contacts at a particular time, a procedure is as follows. One may take the Cartesian coordinates $\mathbf{x} = \{x, y, z\}$ in arrays stored

$\mathbf{x}_{p(1)}$	$\mathbf{x}_{p(2)}$	$\cdots$	$\mathbf{x}_{p(n)}$	
	$\mathbf{x}_{p(1)}$	$\cdots$	$\mathbf{x}_{p(n-1)}$	$\mathbf{x}_{p(n)}$
	↓	Compute distances	↓	
	$\lvert\mathbf{x}_{p(2)} - \mathbf{x}_{p(1)}\rvert$	$\cdots$	$\lvert\mathbf{x}_{p(n)} - \mathbf{x}_{p(n-1)}\rvert$	

Figure 1: Comparison of two arrays to detect close contacts at a particular time.

$\mathbf{x}_{l(1)}$	$\mathbf{x}_{l(2)}$	$\cdots$	$\mathbf{x}_{l(n)}$	
	$\mathbf{x}_{h(1)}$	$\cdots$	$\mathbf{x}_{h(n-1)}$	$\mathbf{x}_{h(n)}$
	↓	Compute distances over range of t	↓	
	$\lvert\mathbf{x}_{l(2)}(t) - \mathbf{x}_{h(1)}(t)\rvert$	$\cdots$	$\lvert\mathbf{x}_{l(n)}(t) - \mathbf{x}_{h(n-1)}(t)\rvert$	

Figure 2: Comparison of two arrays to detect close contacts over a range of times.

on the CM, and duplicate the set. By sorting the elements in advance, we can use a sieve to shorten computation time. The position vectors are rearranged so that one Cartesian coordinate, say z, is sorted in ascending order. First observe that

$$|z_i - z_j| > c \Rightarrow |\mathbf{x}_i - \mathbf{x}_j| > c, \tag{1}$$

and as a consequence of the sorted coordinates, if $k > j > i$, then

$$|z_j - z_i| > c \Rightarrow |z_k - z_i| > c. \tag{2}$$

These together imply that if, at a certain stage of the computation, the minimum of the differences of the z values is greater than c, no further shifts will produce an aligned pair whose Cartesian distance is less than c. Thus if we let p be the permutation that puts sets in z-order, i.e.

$$z_{p(1)} \leq z_{p(2)} \leq \cdots \leq z_{p(n)}, \tag{3}$$

shifting and finding distances as in Figure 1 After each shift the program checks the minimum z differences in parallel of all aligned pairs. If this minimum is greater than c, we terminate the computation, confident that all pairs that are within a distance c of each other have been found. In this way, the actual number of computations is greatly reduced. For a typical catalog of 7000 or so objects, the number of shifts is reduced to about 40 or 50 for a critical distance of 50 kilometers instead of 7000 shifts, with a corresponding speedup.

In order to find all contacts over a range of times and not just those at the step times, this method must be modified. Using the position and velocity from the propagator and higher derivatives from the two-body gravitation equation, we can extrapolate over a time interval the positions of all objects to find the closest approach. For any two aligned objects being compared, the closest approach in the time interval may be found. The sieve described above will not work, however, because the computation of the minimum z differences does not take into account motion during the time interval.

As an alternate, consider the following. If we know, or have an estimate, of the minimum and maximum z values in the time interval, we can make two arrays (Figure 2): the top one containing the Cartesian positions arranged by maximum z over the interval (permutation $h(i)$) and the other with them arranged by the minimum z over the interval (permutation $l(i)$). We align as before but compare over a time interval $[t_0 - T/2, t_0 + T/2]$ determining the closest distance by a power series expansion of the motion.

Because the two arrays are arranged differently, the stopping point of the shifting, has a different criterion than before. Since the upper sorting reflects the minimum values of z possible over an interval and the lower one reflects the maximum, if the difference of the two is greater than the critical distance, those two objects cannot approach within the critical distance during that interval. If all the aligned pairs have that property, no further shifting need be done. The starting point of comparison is also different, though more radically. In the fixed time case, we start with the arrays shifted by one. This insures that we get all contacts and never any duplicates. In the continuous time case, with the arrays in two different arrangements, we must start with the upper array displaced to the right of the lower array, and we will possibly obtain duplicates. How far to the right is determined as follows. Each object has two positions, one in the upper array and one in the lower. The object whose position in the lower array is farthest to the right of its position in the upper array determines the alignment: the upper array should start one shift to the left of the point where this object is aligned, i.e. $\max_i[h(i) - l(i)] - 1$ shifts to the right. The net effect is that significantly more serial shifts are required than for the instantaneous case, though still far fewer than all possible comparisons, and more shifts are needed the longer the time interval.

The actual comparison of Cartesian distances will no longer be a simple cartesian distance, but must involve an extrapolation and solution for the time of closest approach. Typical propagators give us the position and velocity. Using a simple Keplerian model, the acceleration and higher derivatives are easy to compute. Thus, with a power series expansion for distance as a function of time, we solve for the time of closest approach in a time interval. If it proves to be less than the critical distance, the time and distance can be refined later with a Newton search procedure using the actual propagator.

For a one-to-all comparison of a single object against the catalog, the computation is much simpler and faster, and is also suited to the data-parallel model of computation. In this case, the cartesian positions and velocities

Ident	a (km)	e	I (deg)	l (deg)	g (deg)	h (deg)
118	7280.58	.009158	66.78	-24.3	49.2	159.5
211	7101.36	.010261	66.58	-158.4	166.1	56.2
4724	7431.89	.065031	62.81	51.1	-105.9	133.4
11671	6962.73	.003230	82.51	172.3	-306.5	61.4
11694	7847.57	.001765	74.03	-2.2	-12.1	11.0
14801	7146.72	.000728	74.07	155.0	-208.3	83.2
14814	7072.02	.001526	74.03	11.6	154.2	-44.4
17911	7006.59	.002355	82.51	-128.7	134.4	8.8
19396	7053.58	.005222	58.50	-96.8	95.6	-152.5
19440	7730.94	.018462	73.75	178.2	-262.1	127.8

Table 1: Satellites broken up at 1 April 1992 00:00 in simulation.

for the given object can be directly compared with the entire catalog in parallel.

4 Processing

Over a simulation of specified time, propagating by specified time steps, there will be many contacts obtained. This raw contact data will be processed so that all contacts are divided into two categories: traveling pairs and coincidental contacts. The traveling pairs are those objects that are within the critical distance and stay within that distance for most of the simulation. Presumably, it is intended that these objects travel together. The coincidental contacts are therefore more interesting: they represent objects that come within the critical distance at some point in the simulation, and then pass out of contact.

The summary of close contact data is divided into the two categories; For both kinds, the identification numbers for each member of the pair is given. For the traveling pairs, the mean and standard deviation distance and contact percentage are given. For coincidental contacts, the time and closest distance are given. More details, such as Cartesian positions and velocities, latitude, longitude and altitude, can be obtained if desired.

5 Debris Models

Close conjunction detection can be performed within one set, such as the space surveillance catalog, or it can be between two sets only. Either of these sets can come from a variety of sources; a surveillance catalog, objects extracted from a catalog, a simulated constellation of objects, or a simulation of debris.

In order to simulate the effect of debris, we take a very simple model of an exploding satellite. From the center of mass of the source object, the fragments will have velocity vectors equally distributed in all directions spherically, and magnitude distributed with a Maxwell-gas distribution:

$$p(v) = \frac{4}{\sqrt{\pi}} \mu^{3/2} v^2 e^{-\mu v^2} \qquad (4)$$

with

$$\mu = \frac{1}{v_p^2} \qquad (5)$$

v_p = peak speed of distribution. This distribution compares well qualitatively with published data on debris [3].

Multiple breakups will be computed in parallel. These fragments are then propagated as ordinary objects and compared with the original catalog, minus the objects that have disintegrated.

6 Simulations

With this Connection Machine program and the actual catalog from the Naval Space Surveillance Center for April 1, 1992, we simulated various orbital configurations. For one debris study, we simulated the explosions of ten low-earth orbit objects, each into 400 fragments, with peak distribution speeds of 100 meters/second. A simulation with 4 minute time steps for a period of six hours after the explosions comparing the debris against the actual remaining catalog showed that 297 different satellites from the catalog would come within 5.0 kilometers of a fragment, with the closest approach being 0.412 kilometers. The run time on the Connection Machine CM-200 for this simulation was less than an hour.

The satellites whose breakup was simulated, with their orbital elements, are given in Table 1, and the contacts within one kilometer are given in Table 2.

Id #1	Id #2	km	date/time
2741	1007314	0.412	01-Apr-1992 05:24:47.905
12224	1009446	0.562	01-Apr-1992 05:35:42.801
17392	1009011	0.591	01-Apr-1992 03:54:51.836
13027	1009185	0.605	01-Apr-1992 01:29:47.949
12675	1007427	0.612	01-Apr-1992 04:38:37.732
12743	1006921	0.726	01-Apr-1992 01:22:51.916
3393	1007172	0.731	01-Apr-1992 03:45:35.045
17090	1007434	0.985	01-Apr-1992 04:08:04.753

Table 2: Close contacts within 1 km for simulated breakup.

In order to get a feel for the hazard to the proposed space station from the present catalog and from as yet unrecorded debris, we expanded the catalog approximately three times. The space station orbit was taken as 450 km circular at 28.5° inclination for this simulation. A 27,244 piece catalog was constructed by making three copies of the catalog existing on April 1, 1992 with the orbital elements of each new piece randomly perturbed by up to 3.0% of their original values. The space station was run against this approximate catalog for seven days and 46 close approaches were found under 25.0 kilometers, with the closest being 1.415 km. The run time on the Connection Machine CM-200 for this simulation was about 25 minutes.

The contacts of this simulated space station orbit with the expanded catalog of debris under 15 km are given in Table 3.

Ident	km	date/time
21868	1.415	05-APR-1992 01:24:11.323
208940	9.179	08-APR-1992 04:59:41.483
214420	9.380	02-APR-1992 22:56:39.767
105533	10.185	05-APR-1992 16:04:35.928
181105	10.200	02-APR-1992 16:25:36.872
215778	10.366	08-APR-1992 21:36:56.092
314448	11.227	03-APR-1992 05:48:35.748
107967	12.139	06-APR-1992 13:14:57.196
201844	12.463	03-APR-1992 04:22:44.019
16084	12.487	02-APR-1992 08:33:42.560
2179	12.742	03-APR-1992 11:16:22.560
305197	13.722	02-APR-1992 07:56:41.374
305077	14.160	06-APR-1992 19:57:13.399
314417	14.600	05-APR-1992 08:52:03.844

Table 3: Close contacts within 15 km for simulated space station orbit.

Finally, we compared the catalog all-to-all against itself. That is, for a twenty-four hour simulation period of the catalog starting 17 January 1993 at 14:00:00, we found all objects that came within 2.0 km of another object in the catalog. In this period, there were 140 coincidental contacts. There were 57 that came within 1 km; Figure 3 shows the cumulative number of contacts for each distance. The run time on the Connection Machine CM-200 with 8K processors (Paris) for this simulation was about one and a half hours.

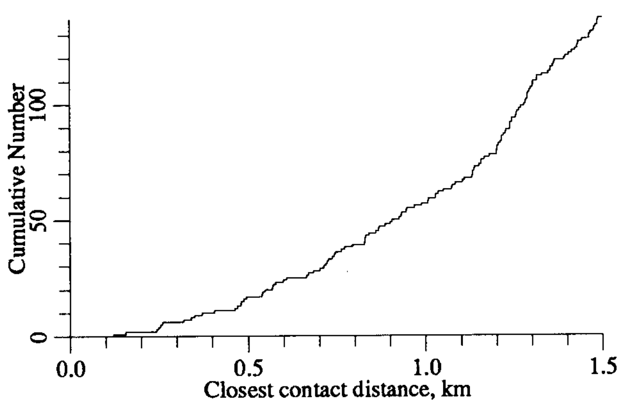

Figure 3: Number of close contacts in 24 hour period less than or equal to given distance.

7 Other applications

Debris poses significant dangers for missions such as the space station. The station's large cross sectional area coupled with occupancy for more than 30 years increases significantly the danger of catastrophic failure as compared to other manned missions. It will be essential to provide accurate warning of collisions, yet necessary to minimize maneuvers and their disruptions to science experiments. With the addition of debris down to 1 cm in size the current catalog of 7,000 objects will grow to an estimated 25,000 objects. This brings up four significant problems that need to be solved; 1) development of better ground based sensors to detect smaller objects 2) increased observation accuracy to improve the elements 3) increased prediction accuracy 4) improvement in cataloging techniques. The first two are radar issues, while the last two may be improved with better computing resources.

Our experiments with COMBO have provided insight into how many of the procedures used in maintaining the catalog can be mapped to a data parallel processor. Detecting 3 times more objects will result in an explosion in the number of Uncorrelated Tracks (UCT), for at first, all of the newly detected objects will not correlate to any known object. Sorting through UCTs to find which ones actually belong to the same object consumes enormous amounts of computer time. It amounts to performing an all-to-all comparison of the set of UCTs to itself. We are now developing a variation of the COMBO algorithm to apply to the UCT problem.

Verification of observations requires propagating the element set to the time of the observation and checking that the observations are close to the propagated object. By duplicating this one-to-one comparison between an observation and its element set to all observations under consideration one has an ideal application for parallel processing.

Another operation is called RETAG, where it is necessary to determine the best element set to associate an observation with. This requires propagating all element sets to the time of the observation and measuring the "closeness" between the observation and the object. This is analogous to the one-to-all COMBO run of a single object against the catalog. We summarize the different comparisons in Table 4.

Improving prediction accuracy requires an improvement in drag modeling. But even the current drag models are not fully incorporated in the analytic prediction programs used at the space surveillance centers. To improve prediction accuracy it will be necessary to make greater use of numerical integrators that can accommodate non-conservative forces. This is quite possible with parallel processors. We are currently investigating the use of numerical integrators for every prediction function in the space surveillance arena. As yet we have not found any insurmountable obstacles. In, fact it may well be that numerical integration is more suitable for some operations. For instance, for COMBO the propagation time steps are quite short, on the order of 5 minutes. Thus the synchronization of a numerical integrator for a large number of objects may be preserved naturally. In addition, integration times of a day are not a problem in themselves, but rather the huge number of objects that may have to be integrated at any one time. Parallel processing may solve this problem by integrating large numbers of orbits simultaneously.

Acknowledgment

We are grateful to Bernie Kelm for significant contributions to this effort on the Connection Machine.

Type Comparison	Cataloging	COMBO
one-to-one	verification	Identical operation on n objects
one-to-all	RETAG	Shuttle conjunctions with the catalog
all-to-all	UCT	All possible collisions in space

Table 4: Analogy between COMBO and cataloging operations

References

[1] D. Brouwer "Solution of the problem of artificial satellite theory without drag," *Astron. J.*, **64** 378–397 (1959).

[2] R. Jehn, *Fragmentation Models*, MAS Working Paper Nr. 312, European Space Operations Centre, December 1990.