NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Technical Memorandum 33-771

A Numerical Comparison of Discrete Kalman Filtering Algorithms: An Orbit Determination Case Study

(NASA-CR-148278) A NUMERICAL COMPARISON CF DISCRETE KALMAN FILTEFING ALGORITHMS: AN OFBIT DETERMINATION CASE STUDY (Jet Propulsion Lab.) 44 p HC $4.00 CSCL 09B

N76-26930

Unclas

G3/65 42348

JET PROPULSION LABORATORY
CALIFORNIA INSTITUTE OF TECHNOLOGY
PASADENA, CALIFORNIA

June 15, 1976

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No. 33-771	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle A NUMERICAL COMPARISON OF DISCRETE KALMAN FILTERING ALGORITHMS: AN ORBIT DETERMINATION CASE STUDY		5. Report Date June 15, 1976
		6. Performing Organization Code
7. Author(s) Catherine L. Thornton, Gerald J. Bierman		8. Performing Organization Report No.
9. Performing Organization Name and Address JET PROPULSION LABORATORY California Institute of Technology 4800 Oak Grove Drive Pasadena, California 91103		10. Work Unit No.
		11. Contract or Grant No. NAS 7-100
		13. Type of Report and Period Covered Technical Memorandum
12. Sponsoring Agency Name and Address NATIONAL AERONAUTICS AND SPACE ADMINISTRATION Washington, D.C. 20546		14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

The numerical stability and accuracy of various Kalman filter algorithms are thoroughly studied. Numerical results and conclusions are based on a realistic planetary approach orbit determination study. The case study results of this report highlight the numerical instability of the conventional and stabilized Kalman algorithms. Numerical errors associated with these algorithms can be so large as to obscure important mismodeling effects and thus give misleading estimates of filter accuracy. The positive result of this study is that the Bierman-Thornton U-D covariance factorization algorithm is computationally efficient, with CPU costs that differ negligibly from the conventional Kalman costs. In addition, accuracy of the U-D filter using single-precision arithmetic consistently matches the double-precision reference results. Numerical stability of the U-D filter is further demonstrated by its insensitivity to variations in the a priori statistics.

ORIGINAL PAGE IS OF POOR QUALITY

17. Key Words (Selected by Author(s)) Aircraft Communications and Navigation Computer Programming and Software Numerical Analysis Statistics and Probability	18. Distribution Statement Unclassified -- Unlimited

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 39	22. Price

HOW TO FILL OUT THE TECHNICAL REPORT STANDARD TITLE PAGE

Make items 1, 4, 5, 9, 12, and 13 agree with the corresponding information on the report cover. Use all capital letters for title (item 4). Leave items 2, 6, and 14 blank. Complete the remaining items as follows:

3. Recipient's Catalog No. Reserved for use by report recipients.
7. Author(s). Include corresponding information from the report cover. In addition, list the affiliation of an author if it differs from that of the performing organization.
8. Performing Organization Report No. Insert if performing organization wishes to assign this number.
10. Work Unit No. Use the agency-wide code (for example, 923-50-10-06-72), which uniquely identifies the work unit under which the work was authorized. Non-NASA performing organizations will leave this blank.
11. Insert the number of the contract or grant under which the report was prepared.
15. Supplementary Notes. Enter information not included elsewhere but useful, such as: Prepared in cooperation with... Translation of (or by)... Presented at conference of... To be published in...
16. Abstract. Include a brief (not to exceed 200 words) factual summary of the most significant information contained in the report. If possible, the abstract of a classified report should be unclassified. If the report contains a significant bibliography or literature survey, mention it here.
17. Key Words. Insert terms or short phrases selected by the author that identify the principal subjects covered in the report, and that are sufficiently specific and precise to be used for cataloging.
18. Distribution Statement. Enter one of the authorized statements used to denote releasability to the public or a limitation on dissemination for reasons other than security of defense information. Authorized statements are "Unclassified—Unlimited," "U.S. Government and Contractors only," "U.S. Government Agencies only," and "NASA and NASA Contractors only."
19. Security Classification (of report). NOTE: Reports carrying a security classification will require additional markings giving security and downgrading information as specified by the Security Requirements Checklist and the DoD Industrial Security Manual (DoD 5220.22-M).
20. Security Classification (of this page). NOTE: Because this page may be used in preparing announcements, bibliographies, and data banks, it should be unclassified if possible. If a classification is required, indicate separately the classification of the title and the abstract by following these items with either "(U)" for unclassified, or "(C)" or "(S)" as applicable for classified items.
21. No. of Pages. Insert the number of pages.
22. Price. Insert the price set by the Clearinghouse for Federal Scientific and Technical Information or the Government Printing Office, if known.

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Technical Memorandum 33-771

A Numerical Comparison of Discrete Kalman Filtering Algorithms: An Orbit Determination Case Study

Catherine L. Thornton
Gerald J. Bierman

JET PROPULSION LABORATORY
CALIFORNIA INSTITUTE OF TECHNOLOGY
PASADENA, CALIFORNIA

June 15, 1976

PREFACE

The work described in this report was performed by the Mission Analysis Division of the Jet Propulsion Laboratory.

ABSTRACT

The numerical stability and accuracy of various Kalman filter algorithms are thoroughly studied. Numerical results and conclusions are based on a realistic planetary approach orbit determination study. The case study results of this report highlight the numerical instability of the conventional and stabilized Kalman algorithms. Numerical errors associated with these algorithms can be so large as to obscure important mismodeling effects and thus give misleading estimates of filter accuracy. The positive result of this study is that the Bierman-Thornton U-D covariance factorization algorithm is computationally efficient, with CPU costs that differ negligibly from the conventional Kalman costs. In addition, accuracy of the U-D filter using single-precision arithmetic consistently matches the double-precision reference results. Numerical stability of the U-D filter is further demonstrated by its insensitivity to variations in the a priori statistics.

CONTENTS

I. INTRODUCTION

In this report attention is focused on the nonstationary linear discrete estimation problem. Not all algorithms applicable to this problem are included in our study. Two important omissions are the continuous-time algorithms [1] and [2] and the Chandrasekhar-type algorithms recently reported by Morf and Kailath [3] and Lindquist [4]. Our main reason for omitting continuous-time algorithms is that such algorithms are heavily dependent upon integration methods for their accuracy and numerical stability. We thought it best not to try, in this report, to compare the continuous and discrete algorithms in terms of numerical accuracy. The Chandrasekhar-type algorithms were omitted because they do not seem to be computationally competitive with our other algorithms for this class of problems. A perhaps more cogent reason for these omissions is that restrictions of time and computer budget prevented an exhaustive all-inclusive study.

The algorithms selected for study include the familiar conventional and stabilized (Joseph form) Kalman filters [5] and [6], the Potter-Schmidt square root filter [6], and the Bierman-Thornton factorization filter [7] and [8].

Examples of numerical failure reported by Bellantoni and Dodge [9], Schmidt, et al. [10], Dyer and McReynolds [11], and others have alerted the estimation applications community to the numerical pitfalls of the familiar Kalman algorithms. Our experience with estimation and control applications engineers, however, indicates that they generally prefer the seemingly simpler Kalman filter algorithms for computer implementation, and they dismiss reported instances of numerical failure. Indeed, the attitude often displayed is that when numerical problems present themselves, more sophisticated algorithms can be used. The implication is, of course, that sophisticated in this context implies complexity, cumbersomeness, and inefficiency. In Refs. [7] and [8], we demonstrated that our factorization algorithms are easy to mechanize and are neither cumbersome nor inefficient. Furthermore, the case study reported here shows in a very dramatic way that the numerical shortcomings of the standard Kalman algorithms contrast markedly with the reliability of the factorization methods. It is important to note that the computational stability of the U-D filter does not rest on this simulation study. Gentlemen's work [12] and [13] relates the U-D measurement update to the numerically stable square root free Givens rotations; and the results of Björck [14] show that our modified Gram-Schmidt time updating algorithm is numerically reliable. Finally, the work by Gill et al. [15] establishes the numerical integrity of our efficient colored noise updating algorithm.

The Potter-Schmidt square root filter also performed very reliably in our study, and the quality of the numerical results differed negligibly from those of the U-D filter. Potter's algorithm, reported in Householder's book [16], is related to Householder orthogonal transformations (cf. Bierman, [6]). Schmidt's time updating of the Potter square root matrix is also accomplished using Householder orthogonal transformations. Thus numeric reliability of the Potter-Schmidt filter rests on the use of orthogonal transformations. Storage and computation requirements for the Potter-Schmidt filter are nearly twice that for the U-D factorization, and because of this, our preference is toward the latter formulation.

The Kalman measurement updating algorithms contrast sharply with the numerically stable factorization algorithms because they have no basis of numerical soundness, and they are held in ill-repute by members of the numerical analysis community. The poor performance of the covariance algorithms exhibited in our case study is thus no surprise to numerical analysts. The abundance of estimation and control literature touting Kalman filter-type algorithms indicates, however, that this information is not sufficiently well known.

As noted earlier, an attitude often encountered among estimation practitioners is that they will switch to the more accurate and stable algorithms if and when numerical problems occur. An analogy comes to mind of a smoker who promises to stop when cancer or heart ailment symptoms are detected. To expand on this analogy, one may note the following:

- Most smokers do not get cancer or heart disease. (Most applications of the Kalman algorithms work.)
- Even when catastrophic illness does not occur, there is diminished health. (Even when algorithms work, performance may be degraded.)
- Smokers can take precautions to lessen the danger, such as smoking low tar or filtered cigarettes. (Engineers can scale their variables to reduce the dynamic range or use double-precision arithmetic.)
- Lung cancer may not be diagnosed until it is too advanced for treatment. (Numerical problems may not be detected in time to be remedied.)

The orbit determination case study reported here highlights these points. We hope that this report will convince the engineering community to alter their "smoking" habits.

Our main goals in this report are:

(a) To emphasize the importance of numerics in determining system performance. Considerable effort has been devoted to modeling, to asymptotic stability, and to the identification of a priori filter statistics; but comparatively little has been done to stress the impact of algorithm selection on system performance.

(b) To show via simulation results that the computer numeric effects mentioned in (a) can cause erroneous predictions based on linear estimation theory.

(c) To show that both the conventional and stabilized Kalman filters are numerically unreliable.

(d) To demonstrate that the Bierman-Thornton U-D factorization filter is computationally efficient and numerically stable.

Items (a) and (b) are intended to show that numeric effects are important both in predicting system performance (i.e., accuracy analysis results) and in computing estimates. The stabilized Kalman filter is often taken as a reference against which other algorithms are compared, and the point of item (c) is to show that this is not a reliable yardstick.

A portion of the forthcoming Mariner Jupiter Saturn 1977 (MJS'77) deep space mission was chosen for our filter comparison study. Problems of this nature are generally solved at the Jet Propulsion Laboratory using the square root information filter ([6], [11], and [17]), a method which has proven to be an efficient, stable, and accurate means of solving orbit determination problems. Our reason for experimenting with other filter algorithms is our interest in future missions involving on-board autonomous navigation. Algorithms of the type compared in this study are more appropriate to problems of this nature because estimates are required frequently and data is processed pointwise.

The reason that our study should be of interest to the entire estimation and control community is that our results do not correspond to a contrived, unrealistic situation. On the contrary, this estimation problem is well posed in an engineering sense; the problem is observable, the transition matrix is not ill-conditioned, the measurement coefficient matrices are not unusually large, and the a priori state error variances were chosen small enough to avoid obvious initial ill-conditioning. Thus, the numerical failures and performance degradations that are documented here should be of general interest.

The outline of this report is as follows. In Section II, the orbit determination problem used in our study is stated, and details of the simulation that are of general relevance are discussed. In Section III, results of the simulation study are presented and discussed; and Section IV contains our conclusions.

II. PROBLEM FORMULATION AND RELEVANT SIMULATION MINUTIAE

A. The Trajectory

The problem chosen for this study is a portion of the forthcoming MJS'77 deep space mission, which involves the approach to Saturn. The period of our interest extends from 30 days before Saturn encounter (point of closest approach) to the encounter. For the initial 20 days, the spacecraft (S/C) trajectory is very nearly rectilinear, a situation that is characteristic of the major portion of most deep space missions. The last portion of the trajectory has a hyperbolic bend due to the effect of Saturn's gravity. Hence, the portion of the trajectory up to encounter is especially useful for accurate determination of planetary mass and S/C position and velocity. This trajectory is thus characteristic of a large number of orbit determination situations.

The nominal S/C trajectory and transition matrices were obtained by integrating the equations of motion and variational equations (cf. [18]) and were donated by MJS navigation team personnel. Because this study is intended to assess only the effects of filter numerical errors, the simulation was constructed from a linear model. The actual trajectory, $x(t)$, is defined by

$$x(t) = x_{nom}(t) + \Delta x(t) \tag{1}$$

where

$$\Delta x(t_{j+1}) = \Phi(t_{j+1} t_j)\Delta x(t_j) + G(t_{j+1}, t_j)\, w(t_j) \tag{2}$$

The components of $x_{nom}(t)$ are the earth-centered S/C cartesian coordinates of position, velocity, and acceleration. The acceleration components of the perturbation $\Delta x(t)$ are modeled as colored noise with time constants of 12 hours and standard deviations of 10^{-11} km/sec^2; and these define variances of the white noise, $w(\cdot)$, appearing in (2). The S/C model used for the orbit determination problem has a piecewise constant acceleration model, with $t_{j+1} - t_j = \Delta t$ taken as 2 hours.

Kalman filtering algorithms with no process noise are notoriously unstable. They frequently give inaccurate but not disastrous results and sometimes give unmistakeable signs of failure, such as negative diagonal entries in the computed covariance matrix and entries of excessive magnitude. A high level of process noise was included (by an order of magnitude) because it was believed that such a model would be less sensitive to numerical errors. Previous experience with Kalman filter algorithms has shown that they have better numerical stability in situations with high process noise levels. It turned out that adding process noise to the filter model did improve the performance of the Kalman filter algorithms, but not enough to regard the results as accurate or reliable. More about this will be discussed in Section III.

The effect of Saturn's mass on the S/C trajectory is very significant near encounter, and because of this, our model includes the GM of Saturn with a 0.1% uncertainty 1σ.

B. The Measurements

Three earth-based tracking stations are involved with monitoring the S/C. Their locations are such that there is coverage at all times. In our simulation, we include two to three doppler points every 2 hours with 1 mm/sec accuracy (for 1 minute averaging time) and occasional range points with an accuracy of 3 meters. There were a total of 535 doppler points and 72 range points in the 30-day arc preceding the Saturn encounter. Since this study was intended to include the significant error sources, the station location position uncertainties were also included in our model (cf. Table 1).

Data for our linear simulation analysis was generated as follows. Doppler and range partial derivatives were evaluated analytically about the nominal trajectory, using JPL's orbit determination software [18]. Pseudo-observables, z, were computed from

$$z = H\,\Delta X + \nu \tag{3}$$

where the elements of H are the partial derivative coefficients; ΔX is the state perturbation (cf. Eq. 2) augmented with the GM_6 (gravitational constant of Saturn) error and the station location errors. Thus, ΔX has a total of 19 components; 9 dynamic and 10 bias parameters. They are position (3), velocity (3), acceleration (3), GM_6 and station locations (9); and ν is white data noise obtained from a Gaussian random number generator.

The statistics used to define our nominal trajectory and data sequence are collected in Table 1.

C. The Filter Algorithms

The five covariance-type filter algorithms compared in this study were the conventional Kalman filter, Joseph's stabilized Kalman filter, a conventional Kalman filter with lower bounding, the Bierman-Thornton U-D factorization filter, and the Potter-Schmidt square root filter. Details of these

Variable	Std. Dev.
Position	1000 km
Velocity	100 m/s
Acceleration	10^{-11} km^2/sec (τ = 12 hr)
	Spin axis - 1 meter
Stn. loc. error	Longitude - 2 meter
	Latitude - 5 meter
GM (Saturn)	.1%
Range	3 meters
Doppler	1 mm/sec (for 1 min count time)

TABLE 1

Summary of A Priori Statistics Used to Generate Nominal Data algorithms, especially those critical to computer implementation, are discussed in Refs. [6], [8] and [17]. For reference, the algorithms are briefly discussed here.

1. Conventional Kalman Filter

$$K = \tilde{P}H^T (H\tilde{P}H^T + r)^{-1} \text{ (Kalman gain)} \tag{4}$$

$$P = \tilde{P} - K(\tilde{P}H^T)^T \quad \text{(conventional measurement update)} \tag{5}$$

where $\tilde{P}$ and P are the a priori and a posteriori covariance matrices, respectively.

$$\tilde{P} = \Phi P \Phi^T + GQG^T \quad \text{(covariance time update)} \tag{6}$$

Here $\tilde{P}$ is the one-step predicted error covariance.

Remark: All of our matrices are time-dependent and should be subscripted; subscripts are omitted, however, for notational simplicity.

Remark: Whenever possible, vector outer products are used to reduce computation. Symmetry of the covariance matrix is preserved by computing only the upper triangular elements. (An exception is our mechanization of the Joseph stabilized algorithm noted below.)

2. Joseph's Stabilized Measurement Update

$$P_1 = \tilde{P} - K(\tilde{P}H^T)^T \tag{7a}$$

$$P = (P_1 - (P_1H^T)K^T) + (Kr)K^T \tag{7b}$$

Symmetry was exploited in (7a) although this does not seem to be important when K is computed using (4), and P is symmetric. P is obtained from (7b) by arranging the computations as indicated by the parentheses.

Remark: Significantly improved results were obtained when all of P was computed in (7b) and the off-diagonal elements were averaged. The fact that numerical results are sensitive to such mechanization details is indicative of the algorithm's instability.

Remark: An alternative arrangement of Joseph's algorithm is

$$W = I - KH \tag{8}$$

$$P = (W\tilde{P})W^T + (Kr)K^T$$

Here too all the elements of P are computed and the off-diagonal elements averaged. This mechanization was not included in our comparisons because it is considerably more wasteful of computer storage and requires far more computations than do any of the algorithms included in our study.

3. Conventional Kalman Filter with Lower Bounding

Here P_1 is computed using (7a), and the filter updated covariance is defined by (9):

$$P(j,j) = \max(P_1(j,j),\ \sigma^2_{min}(j)); \qquad j = 1,\ldots,n \tag{9a}$$

$$P(i,j) = \begin{cases} P_1(i,j) & \text{if } P_1^2(i,j) < M(i,j) \\ \mathrm{SGN}\left(P_1(i,j)\right)\sqrt{M(i,j)} & \text{otherwise} \end{cases} \tag{9b}$$

where $M(i,j) = \rho^2_{min}\ P(i,i)\ P(j,j)$ and $i = 1,\ldots,j-1$. The n components of σ_{min} and the correlation ρ_{min} are chosen a priori.

This mechanization is typical of the techniques that are used to prevent the computed covariance from having diagonals (variances) that are too small, or negative, and correlations that are too large. Such mechanizations are, to be sure, not optimal and the computed P is generally not the actual estimate error covariance. Choosing the bounds σ_{min} and ρ_{min} is something of an art, and appropriate values are generally determined from lengthy simulation studies.

Our purpose for including this lower bound filter algorithm is merely to illustrate that ad hoc "patching" techniques can compensate to some extent for numerical inadequacies of the covariance filter algorithms. Introducing numerical safeguards of this nature is not necessary when factorization algorithms are used.

4. The U-D Factorization Filter

The error covariance matrix is uniquely factored as $P = UDU^T$, with U unit upper triangular and D diagonal. Measurement and time updating algorithms for the U and D factors are given in Refs. [7]and[8].

5. The Potter-Schmidt Square Root Filter

Here the error covariance matrix is factored as $P = SS^T$ with S square. (The factorization is not unique, but that is no problem.) Measurement updating is accomplished by updating S using Potter's algorithm, and time updating is accomplished by triangularizing the augmented array $[\Phi S \; GQ^{\frac{1}{2}}]$ by applying an orthogonal transformation from the right. Algorithm details may be found in Refs. [5], [6] and [17].

Formulae for factorization updating are not as compactly represented as are their covariance counterparts. This should not, however, detract from their utility. Detailed comparisons [7]and[8] have shown that factorization algorithms require no more computer storage, are no harder to mechanize, and are competitive computationally* with their counterparts. Unfortunately, space limitations force us to omit explicit algorithm descriptions.

All the algorithms discussed here propagate estimates using

$$\Delta X = \Delta\tilde{X} + K(z - H\Delta\tilde{X}) \qquad \text{(measurement update)} \qquad (10a)$$

where K is the filter computed gain, and

$$\Delta\tilde{X} = \Phi\Delta X \qquad \text{(time update)} \qquad (10b)$$

---

*The U-D algorithms are in certain circumstances even more efficient that are the covariance algorithms.

Remark: There is a significant accuracy deterioration in the estimate error when single-precision arithmetic is used to compute (10); because of this, the error estimates are retained in double-precision (regardless of whether the filter algorithm is single- or double-precision).

D. Numerical Accuracy

The complexity of our case study problem prohibits closed-form solutions, and consequently the numerical solution computed using double-precision* arithmetic is used as a reference. Estimates and sigmas, computed using the Bierman-Thornton and Potter-Schmidt factorization algorithms, agreed to 10 or more digits when computed using double-precision arithmetic. The conventional and stabilized Kalman filter algorithms, computed using double-precision, agreed to eight or more digits with the other results. These comparisons established:

- Confidence that our computer implementation of the various filter algorithms was correct.
- Assurance that when double-precision arithmetic is used, numerical errors due to roundoff and cancellation are of no major consequence (to the orbit determination filtering problem); all four of the algorithms were sufficiently accurate for this problem.
- Limitations on computable accuracy. Even when all filter computations were in (18-digit) double-precision, the results could not be trusted to more than 10 digits.

One might surmise from our double-precision comparisons that we could expect filtering accuracy to be about half of the arithmetic precision used in the computation. With a few exceptions, the single-precision factorization

*Our simulations were carried out on a UNIVAC 1108 having a 27-bit characteristic (8-9 decimal digits) in single-precision and a 60-bit characteristic (18 decimal digits) in double-precision.

algorithms satisfied this rule. On the other hand, the two covariance algorithms, when operated in single-precision, exhibited unmistakable numerical deterioration (cf. Section III) and could not be relied upon at all. The results obtained show that accuracy of the covariance algorithms deteriorates rapidly as computer word length decreases.

Remark: The carefully checked double-precision programs were converted to single-precision* by removing the FORTRAN IV "implicit double-precision" statement. In addition, the filter programs were arranged so that both single- and double-precision versions used the very same (single-precision) inputs. These precautions guaranteed that the sometimes marked differences in the single- and double-precision estimation results was due solely to the numerics of the filtering algorithms.

E. Simulation Philosophy

A single nominal trajectory, one proposed for the MJS mission, was chosen for our case study. Transition and observation matrices were constructed corresponding to this nominal. The various filter algorithms, computed in single-precision and operating from these inputs, were compared. Numerical effects were evidenced by the differences in computed variances and gain profiles of the various algorithms. Especially prominent was the frequent appearance of negative variances arising from both the conventional and stabilized covariance filter algorithms.

One might surmise from these results that since the gains and sigmas computed using the factorization algorithms stayed close to the correct values, the estimates based on these gains could be trusted. On the other hand, the covariance filters produced negative variances and markedly different gain

---

*In the single-precision programs, however, estimates were computed in double-precision and inner products were accumulated in double-precision before rounding.

profiles. Thus one might expect that estimates based on these algorithms would be inaccurate. These speculations were easily corroborated using a double-precision covariance error analysis program which evaluates the effects of nonoptimal gains by computing actual error covariances (cf. Refs. [1] and [17]).

<u>Remark</u>: Since each filter operates on the same data and state transition matrices and computes estimates in double-precision, only the gain calculations differ. Hence, it is the gain algorithms that we are comparing.

Two principal results of the gain evaluations were:

(1) The U-D and square root covariance algorithms performed as anticipated; i.e., the gain profiles were nearly optimal in that the actual and the (single-precision) computed covariances were close to each other; and close to the optimal.

(2) Actual covariances corresponding to the covariance filters were considerably larger than the optimal covariances. The magnitudes of the actual variances, however, indicated that the filter estimates would at least track the actual trajectory.

To illustrate the results predicted by the evaluation program, an actual trajectory (a perturbation to the nominal that was consistent with our assumed filter statistics) and a data noise sequence (consistent with the range and doppler accuracies) were included in our study problem. The gain profiles were applied to this simulated problem, and estimate errors consistent with those predicted by the actual variances resulted. Variations were introduced into the simulation model to assure that the results were not coincidental. The consistency of the results convinced us that the sample estimate results, to be described in the next section, are not happenstance but can truly be regarded as typical.

III. SIMULATION RESULTS

Of the five filter algorithms mechanized in this study, we had the most difficulty with the Kalman stabilized formulation. This is somewhat surprising because the equations appear so simple (and we have previous experience coding Kalman filters). Our difficulty can be traced to numerical inconsistencies between the single- and double-precision mechanizations.* It turned out that there were no programming errors, only that the single-precision results were sensitive to the a priori statistics and to the grouping of terms in the computer code. By contrast, the single-precision factorization results were always consistent with the double-precision reference. These findings and other results of interest are related by describing the following aspects of our study:

- Results for the basic 19-state filtering problem
- The effects of scaling the a priori and data noise variances
- Phenomena related to lower dimensional models

A. CASE 1: The Complete 19-State Model

The first case we study in detail is the 19-parameter model described in Section II. The a priori statistics given in Table 1 are typical assumptions for this kind of estimation problem. In orbit determination, it is standard practice to begin filtering with large a priori uncertainties in position and velocity. However, to avoid the initialization numerical instability associated with the Kalman algorithms, we chose to use relatively small a priori variances.

---

*Wampler [19] points out that these are sufficient reasons to declare an algorithm numerically unstable and to abandon it. Our findings are consistent with his conclusion.

For this case and the others to follow, the double-precision filters agree to at least 8 digits (and generally to 10 or more). The single-precision programs, however, produce a variety of results. Actual filtering performance* for this case is illustrated in Figures 1 and 2.

Remark: We chose the root-sum-square of position and velocity errors as a measure of estimation accuracy because these parameters are of primary interest in navigation and are representative of the general filtering results recorded in this study.

In Figures 1 and 2, the position and velocity uncertainties of the factored single-precision algorithms are shown to agree with the double-precision references. It is important to note that this consistency was observed in all of the cases studied; i.e., the single-precision factorization results always agreed with the double-precision reference cases.

The single-precision Kalman algorithms, on the other hand, exhibit no such numerical stability. Obvious numeric deterioration, in the form of negative computed variances, appear at inexplicable times. Negative variances first appear in the conventional Kalman mechanization after four days of filtering and after ten days when the stabilized mechanization is used. Several other surprising phenomena warrant mention.

(1) Both the conventional and stabilized algorithms compute intermittent negative variances. From a total of 607 measurement updates, the conventional algorithm computes negative variances 177 times and the stabilized algorithm produces negative variances 69 times.

(2) Bias parameter variances are also intermittently negative. This violates the theoretic monotonicity of constant parameter variances

---

*Actual accuracies were obtained from the error analysis program which evaluates computed gain profiles from the various filter algorithms.

(i.e., bias parameter variances should always decrease as more data is processed). To illustrate the erratic behavior exhibited, we note that at 9.75 days the stabilized algorithm computes $\sigma^2_{GM} = -1.8 \times 10^9$, and at 10 days, this is adjusted to 1.7×10^4. The correct (double-precision) value is $\sigma^2_{GM} \doteq 5 \times 10^3$.

(3) As the next case will show, the numerical instability discussed here is related to the choice of a priori statistics. However, even in the case of the conventional algorithm (which exhibits numerical failure earlier), it takes more than 48 time and 80 measurement updates before negative computed variances appear.

(4) The appearance of negative diagonal elements in the computed covariance is not necessarily related to filter variances which are tending toward zero. Their appearance in this case acts instead as an indicator of algorithm numeric deterioration.

Perhaps the most surprising result of this example is the fact that the Kalman algorithms, despite their unsatisfactory computed covariances, are able to generate meaningful (but not accurate) state estimates. According to the error analysis results, the gain profiles generated by the Kalman algorithms do lead to estimates which track the actual trajectory. The results, while not acceptable, are better than we anticipated they would be considering the intermittent appearance of negative diagonals.

A simulation was performed to demonstrate the accuracies predicted by the error analysis. A data noise sequence and a trajectory were generated using the same model assumed in the error analysis. This simulated data was filtered by each of the algorithms of our study, and estimate errors were then compared. The results are illustrated in Figures 3 and 4. Notice how closely the gain evaluation results of Figures 1 and 2 predict the error curves of the sample path.

As anticipated, the conventional algorithm in single-precision produces enormous errors in position and velocity at 4 days. We note with interest that the estimates and computed sigmas obtained from the stabilized Kalman filter, when monitored at one-day intervals, show few telltale signs of numeric deterioration. Except for the times when negative variances are printed (only 3), these estimates and sigmas appear reasonable and consistent. Only when the results are compared with the double-precision reference does it become apparent that the computed Kalman estimates are far from optimum.

By comparison, estimates computed using the factorization algorithms agree to about 4 or 5 digits with the double-precision values. This agreement corresponds to better than 1 km in position and 50 mm/sec in velocity. These single-precision accuracies are particularly impressive when it is noted that estimation uncertainties are two orders of magnitude greater than these differences; i.e., the differences in the single- and double-precision results are in the noise level.

In every case studied, the relative position and velocity accuracies displayed the same general agreement illustrated in Figures 1-4. Simulation and error analysis results were also consistently similar. We utilize these observations to restrict our subsequent discussions, for the most part, to the comparison and analysis of position uncertainties. Thus, unnecessary discussions of velocity uncertainties, and simulation results are omitted. To further curtail the length of this report, we omit the conventional Kalman algorithm from our subsequent discussions; the numerical instability of the conventional algorithm is already well documented in Refs. [5] and [9] - [11]. We note in passing that our experience reinforces this point; viz., almost every conventional Kalman (single-precision) test case contained computed covariance matrices with negative diagonal elements.

An Aside: Recall that the stabilized update formula was introduced as a computational improvement to the conventional formula and was supposed to assure nonnegativity of the computed covariance matrix. The results of our study show that the stabilized algorithm does not guarantee nonnegativity of the computed covariance (or even nonnegativity of the diagonal elements).* Indeed our study shows, contrary to popular belief, that one can actually obtain worse results using the stabilized formula in place of the conventional one (worse in the sense that negative diagonal elements appear more often and position errors are at times larger in the case of the stabilized algorithm). Because it does give improved performance in various other applications, we do not suggest that one abandon the stabilized algorithm and return to the simpler conventional formula. Actually we think that both formulae are bad and should not be used as computational algorithms.∎

Numerical divergence of the Kalman filter is often associated with computed covariance matrices that lose their nonnegativity. Hence it is a common practice to attempt to preserve nonnegativity by bounding the diagonals from below (to prevent computed variances from becoming too small) and to limit the correlations between pairs of variables. Trying to stabilize the conventional Kalman algorithm with such patches opens a pandora's box of filtering alternatives; e.g., should the lower bound on the velocity sigmas be 1.0 m/sec or 0.1 m/sec? Should the maximum correlation be .99 or .98? Should the bounds be time-varying? etc. Experimenting with this multitude of alternatives can be frightfully expensive, especially when (as is often the case) the choice of patch factors is problem-dependent.

---

*This numeric instability is not caused by the vector outer product algorithm mechanization; similar results have been observed using the matrix product mechanization.

For this case study, filtering results are indeed sensitive to the choice of bounds, as Figure 5 illustrates. This figure displays the RSS position error profiles produced by the single-precision patched algorithm for various bounding schemes. Comparing Figures 3 and 5, it can be seen that patching gives a marked improvement over the stabilized Kalman results; but all of the patched curves are far above the optimal result. Filtering accuracies are compared in Figure 6, and the poor performance of the patched filter is demonstrated. Continuing the comparison, we note that the patched algorithm is not even efficient. To see this, one has only to include the simulations required to choose an appropriate set of patch factors and the extra computation and logic that the patched algorithm requires.

Our conclusion from the study of this algorithm is that the practice of introducing ad hoc patch factors to combat Kalman filter numerical divergence results in algorithms that are significantly less efficient and accurate than the factorization methods. We omit patching techniques from further consideration but close our discussion of this subject with the observation that results analogous to those of Figs. 5 and 6 were obtained for all the other cases studied.

B. CASE 2: Scaling of the A Priori State and Data Covariances

Numerical ill-conditioning of the Kalman filter can often be attributed to the presence of large initial uncertainties and relatively small data covariances. These effects can be reduced by scaling the filter inputs, but the improved numerical conditioning is somewhat offset by the effects of using incorrect a priori filter statistics (cf. Figure 7). By combining orbit determination intuition and numerical experimentation, we found that reducing the initial velocity uncertainty by an order of magnitude (to 10 m/sec) and increasing the range uncertainty (from 3 meters to 10 meters) resulted in a "stable" stabilized algorithm. For this choice of filter statistics neither the conventional nor the stabilized Kalman algorithm computed negative variances. Moreover, for this example the simulation estimate errors are consistent with the filter formal statistics. Such a situation creates a false sense of security because, while the Kalman algorithms appear to operate well, they are, in fact, woefully suboptimal. Refer to Figure 7 and note that the Kalman filter errors (the middle flagged curve) are much larger than the achievable filter performance (the bottom curve). To appreciate the seriousness of the Kalman algorithm position error, we note that the incremental error due to the use of the Kalman algorithm is larger than mission navigation requirements allow.

The results in Figure 7 also show that the Kalman filter is more accurate when suboptimal ($\sigma_v = 10$ m/sec, $\sigma_R = 10$ meters) rather than optimal ($\sigma_v = 100$ m/sec, $\sigma_R = 3$ meters) covariances are assumed! For the larger part of the filtering period, the suboptimal Kalman estimates, with scaled inputs, are an order of magnitude more accurate than are the "optimal" computed results.

If only one of the a priori uncertainties (σ_v or σ_R) is scaled, the stabilized Kalman algorithm continues to produce negative computed variances. The situation with scaled σ_v is illustrated in Figure 8. When σ_v is scaled down an order of magnitude, the initial velocity variance is scaled down by two orders of magnitude. However, instead of improving filter numerics, the stabilized algorithm with reduced a priori increased the number of times that negative variances were computed (from 69 to 114). Note in Figure 8 how the position errors peak earlier (6 days) than when the larger σ_v a priori was used.

In a filtering problem with observability and significant amounts of process noise, one would expect that estimates should depend, loosely speaking, only on the recent past. Thus, estimate error profiles corresponding to the use of different a priori statistics should, except for initial transient effects, look quite similar. Such is the case with the factorization filters, as the bottom two curves of Figure 8 illustrate. In contrast, the stabilized Kalman algorithm produces error profiles which are quite sensitive to the choice of a priori statistics (cf. the topmost curves of Figures 7 and 8).

The conclusion to be drawn from this discussion is that numerical instability can cause unpredictable results which violate established estimation principles.

C. CASE 3: Reduced-Dimension Problems

The results reported in the previous cases were obtained using the complete 19-state model. In this section, models of smaller dimension are examined. Our results here show, among other things, that the numerical instability of the Kalman algorithm is not caused by the dimensionality of the model; and that the inclusion of process noise improves the appearance of the computed covariance but not the accuracy of the estimate.

The smallest, physically meaningful model corresponding to the planetary approach problem has only the six position and velocity variables. This 6-state system is a parameter estimation problem because, even though the variables are time-dependent, there is no process noise. The Kalman updating algorithms are known to be numerically unstable for parameter estimation problems, and consequently we were only mildly surprised to find that the stabilized algorithm computed 96 covariances with negative diagonal elements. Just as in case 1, the stabilized filter intermittently computes covariance matrices with negative diagonal elements. This 6-state filter was applied to the simulated trajectory (based on the complete 19-state model) and managed to partially track the spacecraft.

A question that immediately comes to mind is whether the stabilized Kalman estimate errors for this case are due primarily to the use of a reduced-order filter model, or to the numerics of the algorithm. The answer becomes obvious when the factorization algorithms are applied to this problem. The factorization filters computed covariances which were, as usual, close to the corresponding reference double-precision results. The actual position uncertainties in Figure 9 show, however, that position errors corresponding to the single-precision stabilized algorithm are orders of magnitude larger than the position errors corresponding to the single-precision factorization algorithms. By comparing the factorization curves of Figures 1 and 9, one can see that the accuracy loss due to mismodeling is considerable. Comparison of the stabilized curves for these two figures suggests that either the stabilized algorithm compounds the effects of mismodeling or the numerical errors are so large that they have become the dominant errors.

To further separate the effects of mismodeling from the numerics, we calculated the actual covariances corresponding to the reduced model (i.e., assuming no mismodeling). Comparing Figures 9 and 10, one finds that position uncertainties corresponding to the stabilized algorithm are very nearly the same. The results indicated in these figures show that the numerical errors associated with the stabilized algorithm are so large that they completely obscure the effects of mismodeling. By contrast, the factorization curve of Figure 10 demonstrates the accuracy of the U-D and Potter-Schmidt algorithms. Because the numerical errors have been removed, the factorization curves of Figures 9 and 10 clearly show how 6-state filtering accuracies are affected by the presence of unmodeled parameters.

We chose as our second model a 9-state system which includes the three colored noise accelerations in addition to position and velocity. The reason for choosing this model is that it includes a significant amount of process noise (cf. Table 1); and process noise is generally assumed to stabilize Kalman filter numerics.

Computed filter results appeared to corroborate this theory. For example, the stabilized Kalman filter produced covariances, gains, and estimates (based on our simulation sample) which looked reasonable. The results differed, however, from those obtained using the U-D and Potter-Schmidt algorithms. Error analysis of the two sets of results (cf. Figure 11) shows that the factorization results are free of numerical errors and that Kalman results are severely degraded. Note how the numerical deterioration of the Kalman algorithm translates into position errors that are orders of magnitude larger than they need be. Our conclusion here is that while the inclusion of process noise improves the performance of the Kalman algorithms, the results still lack the accuracy achievable using factorization methods.

IV. Conclusions

Excellent numeric accuracy and stability were demonstrated throughout this study by both the U-D and the Potter-Schmidt factorization algorithms. Both algorithms mechanized in single-precision gave results that were close to the double-precision references. In every case of our comprehensive study, these algorithms out-performed all of the Kalman algorithms. Accuracy improvements were generally substantial, and often the improvements were orders of magnitude. Numerical stability of the factorization algorithms was evidenced by their lack of sensitivity to the choice of a priori variances and process noise levels.

The Kalman filters were, in contrast, very sensitive to the input statistics. Numerical deterioration was rampant in both the conventional and stabilized algorithms, and computed covariance matrices with negative diagonals were a common occurrence. Even when the input statistics were modified to stabilize the numerics, the Kalman algorithm performed poorly. In these cases, the accuracy degradation was not apparent but had to be identified using a double-precision error analysis program. Our analysis showed numerics to be the dominant error source in the Kalman algorithms, and they completely obscured the effects of mismodeling. This result is of special interest because engineers rarely include the effect of numerical error in their construction of error budgets and mission design requirements. Our results suggest that when factorization algorithms are employed, the engineer can justifiably ignore numeric effects.

Since good things are seldom free, one might surmise that the accuracy and stability associated with the factorization methods must be balanced with additional, and perhaps prohibitive, amounts of computation. References [7] and [8] contain detailed arithmetic operation counts which show that the Potter-Schmidt algorithm is not unreasonably costly (and generally compares with the stabilized Kalman algorithm), while the Bierman-Thornton U-D algorithm is competitive with the conventional Kalman mechanization. For the problem at hand, we have more complete information about computer costs; viz., computer overhead costs associated with indexing, logic, etc., are included in our CPU timing records. Table 2 gives the CPU times for the 19-state model of case 1. The Potter-Schmidt algorithm is the most expensive of the algorithms, and this is our primary objection to it. Indeed, it was this cost problem that triggered our quest for a more efficient factorization algorithm. Our success is evidenced in Table 2; viz., the U-D filter was even faster than the conventional Kalman algorithm. Time propagation costs are influenced by the numbers of bias and colored noise variables (cf. [8]). While the U-D method is not always cheaper than the conventional Kalman algorithm, it is generally competitive.

Demonstrating with a meaningful engineering problem, we have shown that numerical errors can dominate performance of the Kalman algorithms, that the U-D and Potter-Schmidt factorization algorithms dramatically reduce the effects of numerical errors, and finally that the cost of using the U-D algorithm differs insignificantly from the costs of the conventional Kalman filter. Thus our U-D filter offers numerical reliability at an affordable price.

Filter Algorithm	Single-Precision	Double-Precision
Conventional Kalman	39	49
Stabilized Kalman	45	59
U-D	38	46
Potter	63	80

TABLE 2

Comparison of Filter Execution Times*

*CPU time in seconds

References

[1] Jazwinski, A. H., Stochastic Processes and Filtering Theory, Academic Press, New York (1970).

[2] Lindquist, A., "Optimal Filtering of Continuous-Time Stationary Processes by Means of the Backward Innovation Process," SIAM J. Control, Vol. 12, No. 6 (1974), 747-755.

[3] Morf, M., Kailath, R., "Square-Root Algorithms for Least-Squares Estimation," IEEE Trans. Automatic Control, Vol. AC-20, No. 4 (1975), 487-497.

[4] Lindquist, A., "A New Algorithm for Optimal Filtering of Discrete-Time Stationary Processes," SIAM J. Control, Vol. 12, No. 4 (1974), 736-746.

[5] Kaminski, P. G., Bryson, A. E., Schmidt, S. F., "Discrete Square-Root Filtering: A Survey of Current Techniques," IEEE Trans. Automatic Control, Vol. AC-16, No. 6 (1971), 727-735.

[6] Bierman, G. J., "A Comparison of Discrete Linear Filtering Algorithms," IEEE Trans. Aero. Elect. Systems, Vol. AES-9, No. 1 (1973), 28-37.

[7] Bierman, G. J., "Measurement Updating Using the U-D Factorization," Proc. 1975 IEEE Conf. on Decision and Control, 337-346.

[8] Thornton, C. L., Bierman, G. J., "Gram-Schmidt Algorithms For Covariance Propagation," Proc. 1975 IEEE Conf. on Decision and Control, 489-498.

[9] Bellantoni, J. F., Dodge, K. W., "Square-Root Formulation of the Kalman-Schmidt Filter," AIAA J., Vol. 5 (1967), 1309-1314.

[10] Schmidt, S. F., Weinberg, J. D., Lukesh, J. S., "Case Study of Kalman Filtering in the C-5 Aircraft Navigation System," 1968 Joint Automatic Control Conf., Case Studies in System Control.

[11] Dyer, P., McReynolds, S., "Extension of Square-Root Filtering to Include Process Noise, J. Opt. Theory and Applic., Vol. 3, No. 6 (1969), 444-458.

[12] Gentlemen, W. M., "Least Squares Computations by Givens Transformations Without Square Roots," J. Inst. Maths. Applic., Vol. 12 (1973), 329-336.

[13] Gentlemen, W. M., "Error Analysis of QR Decompositions by Givens Transformations," J. Linear Algebra and its Applic. 10 (1975), 189-197.

[14] Björck, A., "Solving Least Squares Problems by Gram-Schmidt Orthogonalization," BIT, Vol. 7 (1967), 1-21.

[15] Gill, P. E., Golub, G. H., Murray, W., Saunders, M. A., "Methods for Modifying Matrix Factorizations," Math. of Comp., Vol. 28, No. 126 (1974), 505-535.

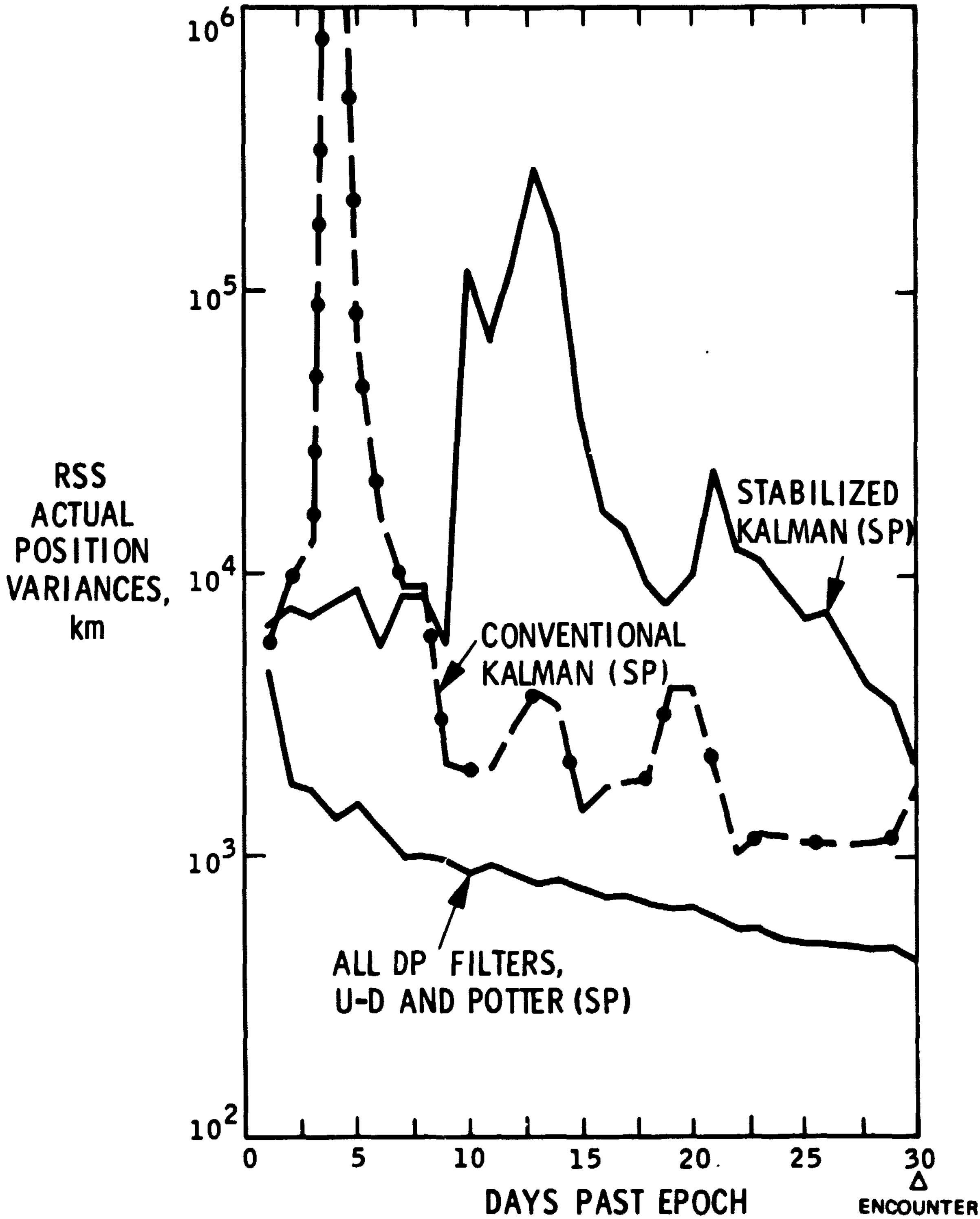
[16] Householder, A. S., The Theory of Matrices in Numerical Analysis, Blaisdell, New York (1964).

[17] Bierman, G. J., Factorization Methods for Discrete Sequential Estimation, Academic Press, New York (1976).

[18] Moyer, T. D., "Mathematical Formulation of the Double Precision Orbit Determination Program (DPODP)," Jet Propulsion Laboratory Technical Report, TR 32-1527 (1971).

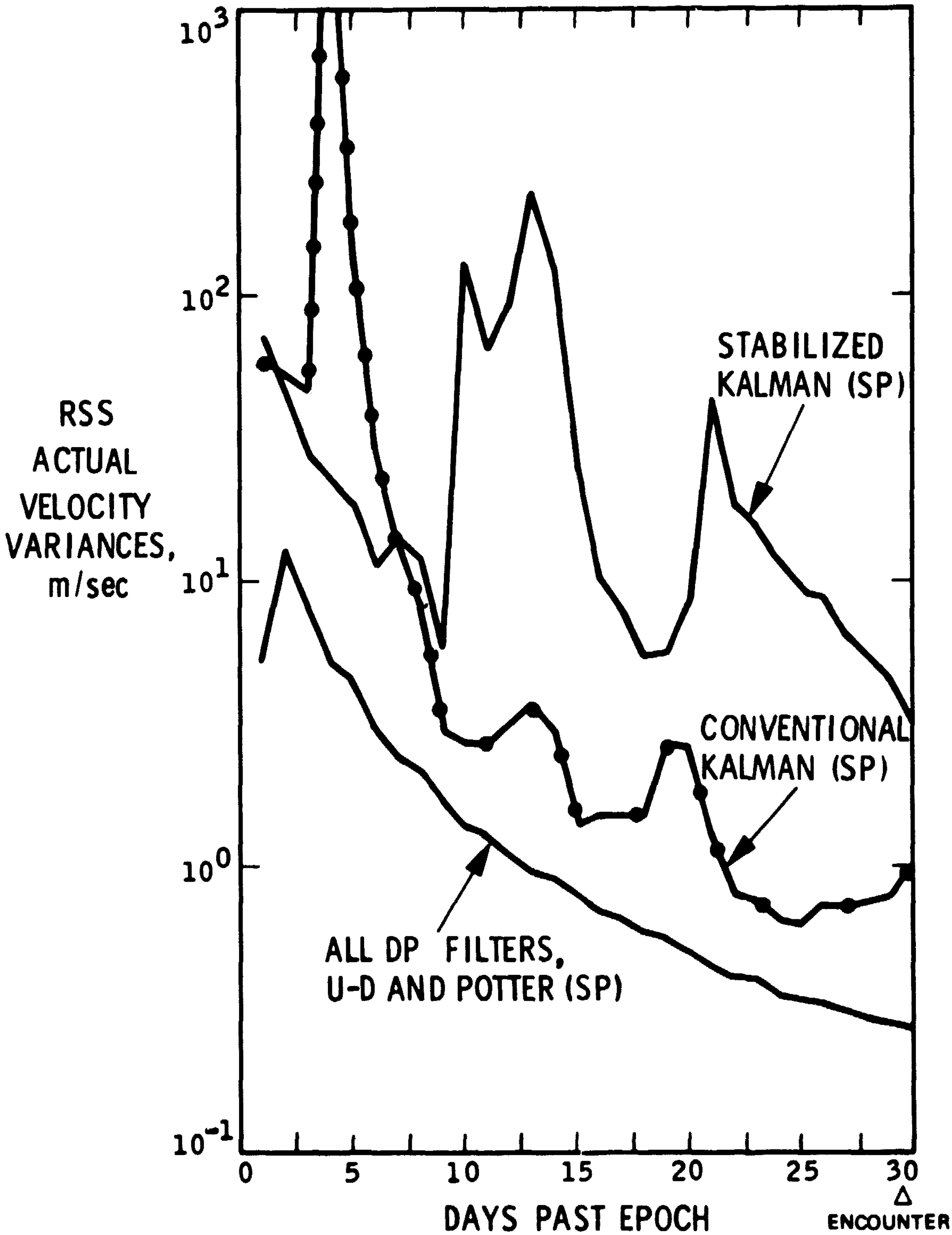
[19] Wampler, R. H., "A Report on the Accuracy of Some Widely Used Least Squares Computer Programs," J. Amer. Stat. Assoc., Vol. 65, No. 330, 1970, 549-565.

1. COMPARISON OF ACTUAL POSITION UNCERTAINTIES *



* CASE 1, 19 STATE MODEL WITH CORRECT A PRIORI

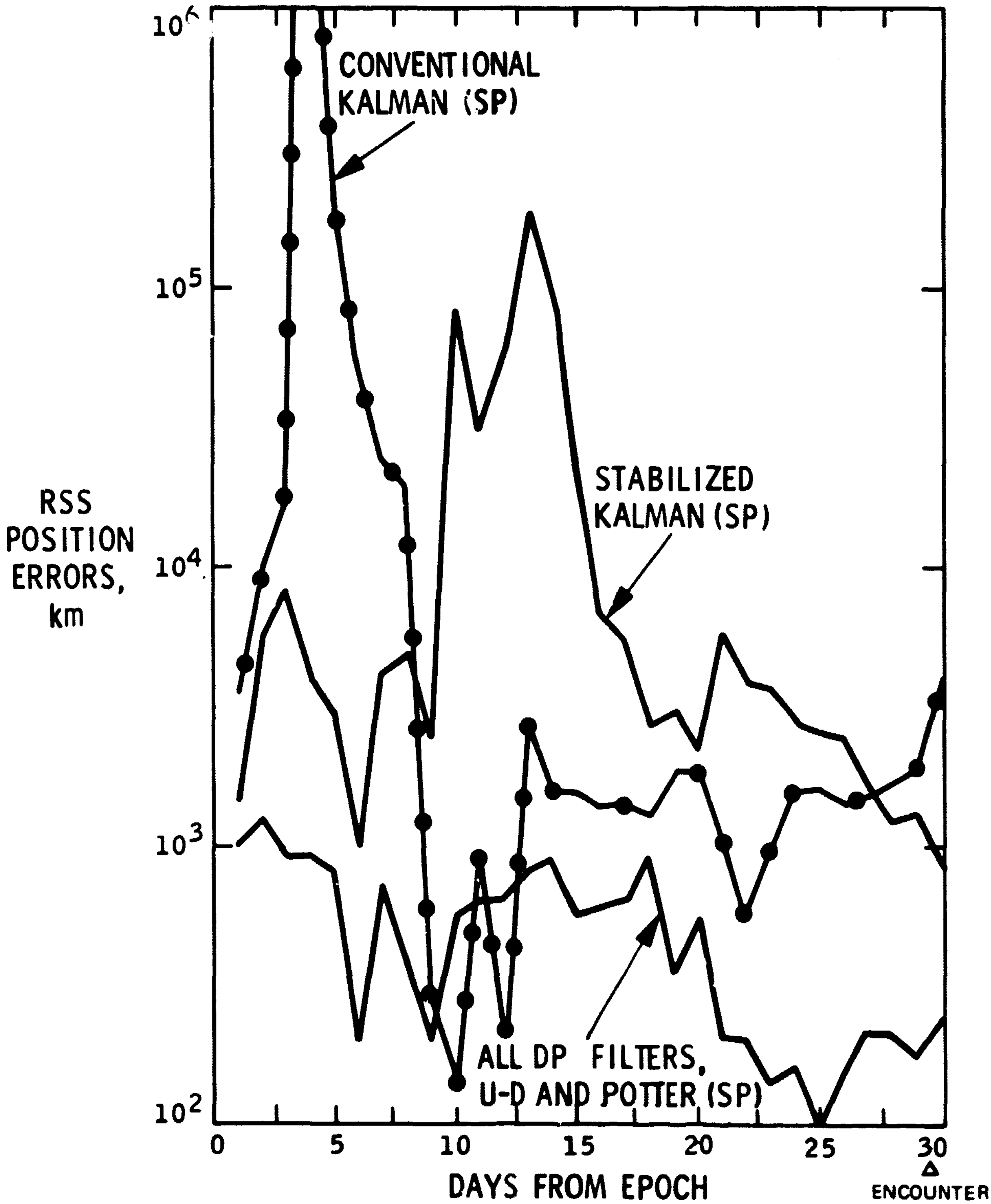
2. COMPARISON OF ACTUAL VELOCITY UNCERTAINTIES*



* CASE 1, 19-STATE MODEL WITH CORRECT A PRIORI

3. COMPARISON OF POSITION ERRORS*

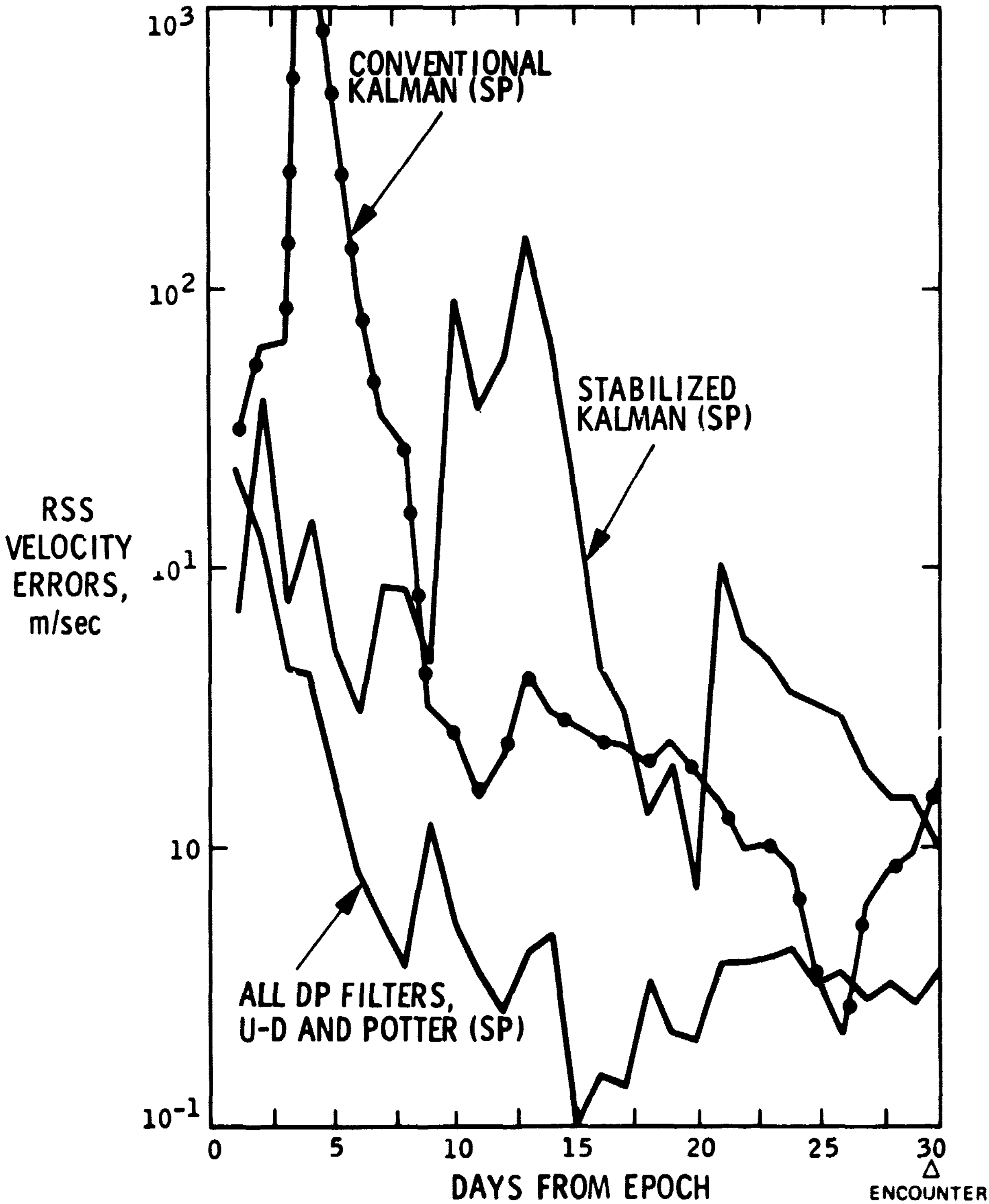
SINGLE SIMULATION RESULT



* CASE 1, 19-STATE MODEL WITH CORRECT A PRIORI

4. COMPARISON OF VELOCITY ERRORS*

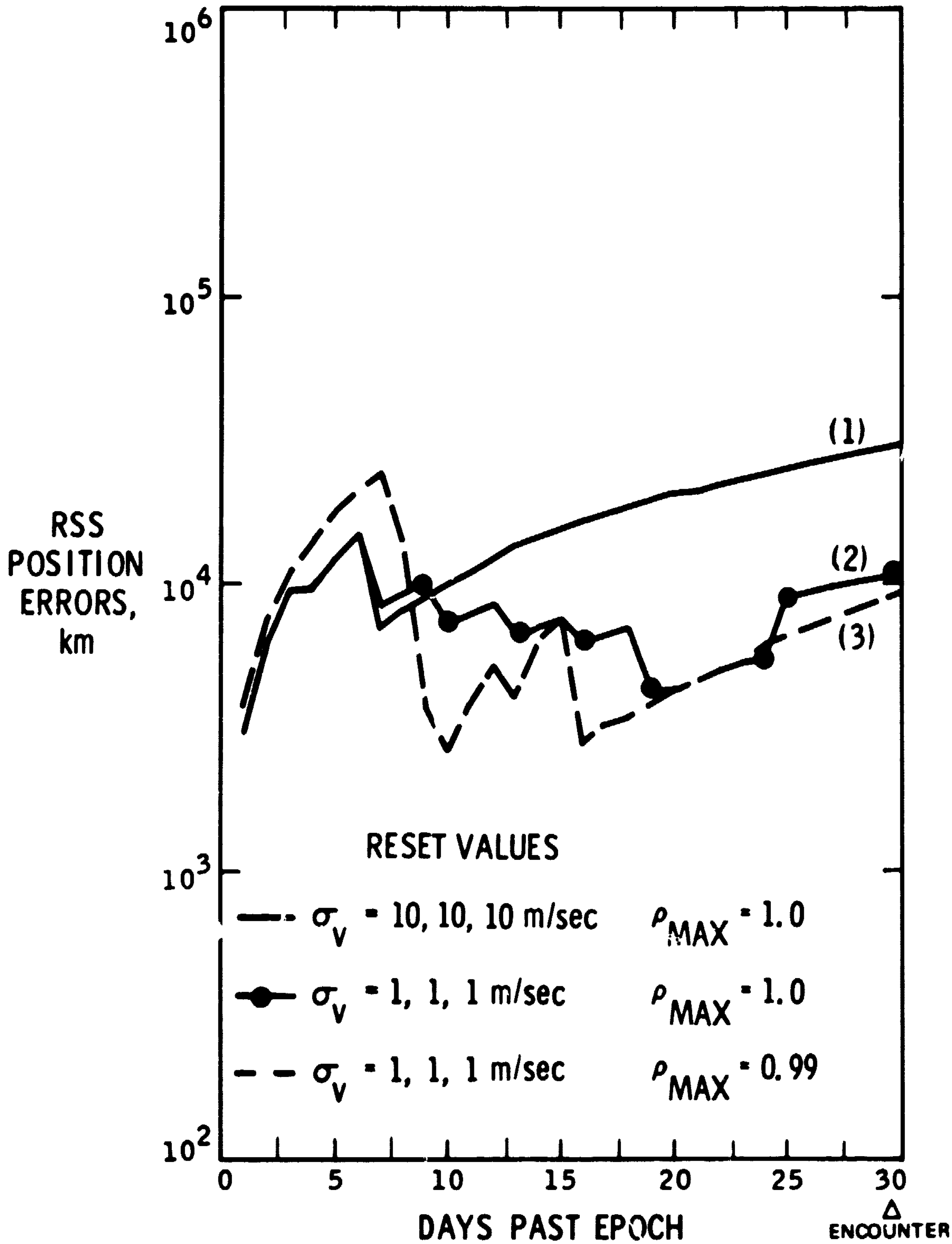
SINGLE SIMULATION RESULT



* CASE 1, 19-STATE MODEL WITH CORRECT A PRIORI

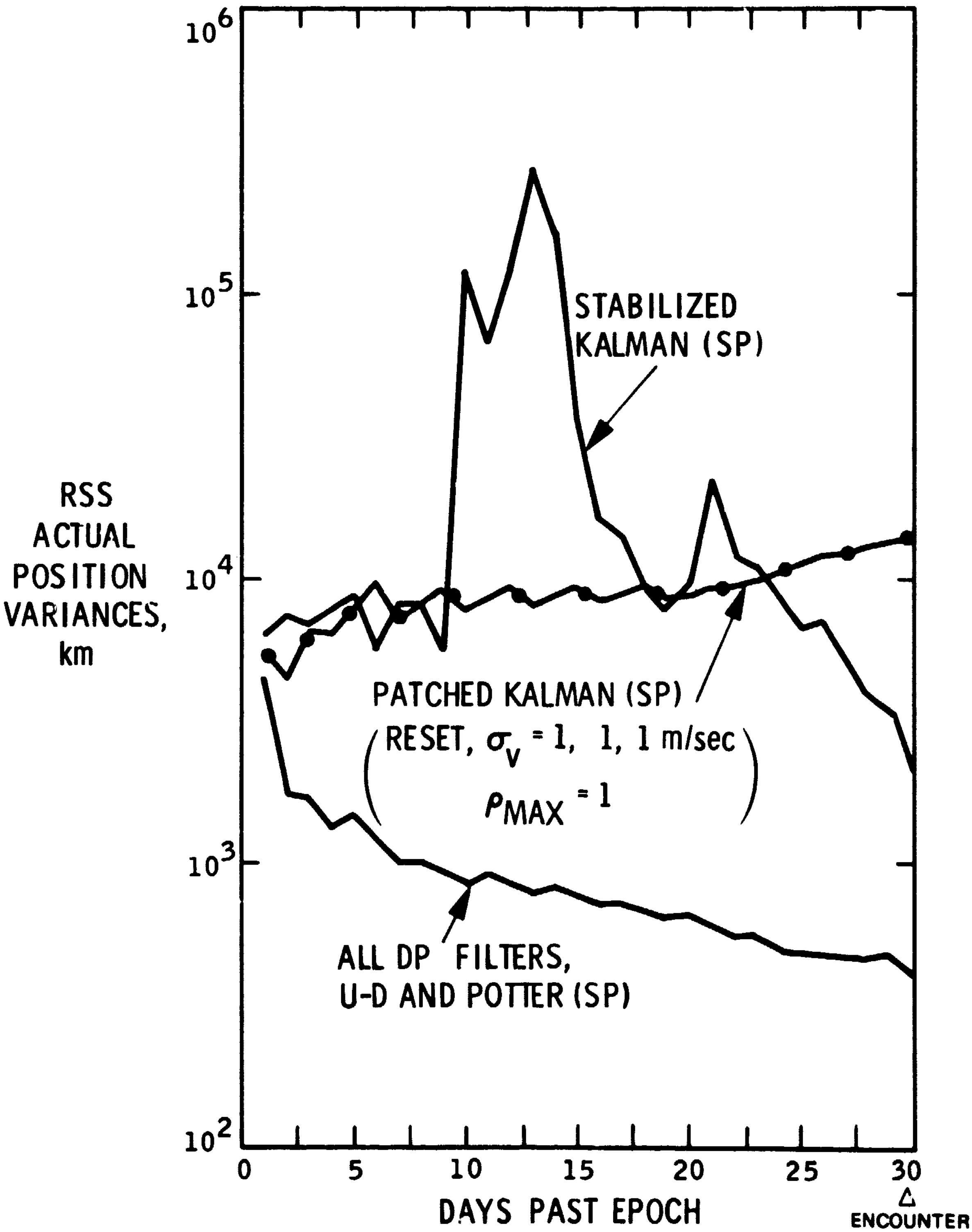
5. COMPARISON OF PATCHED KALMAN ALGORITHM PERFORMANCE* FOR DIFFERENT RESET VALUES

SINGLE SIMULATION RESULT



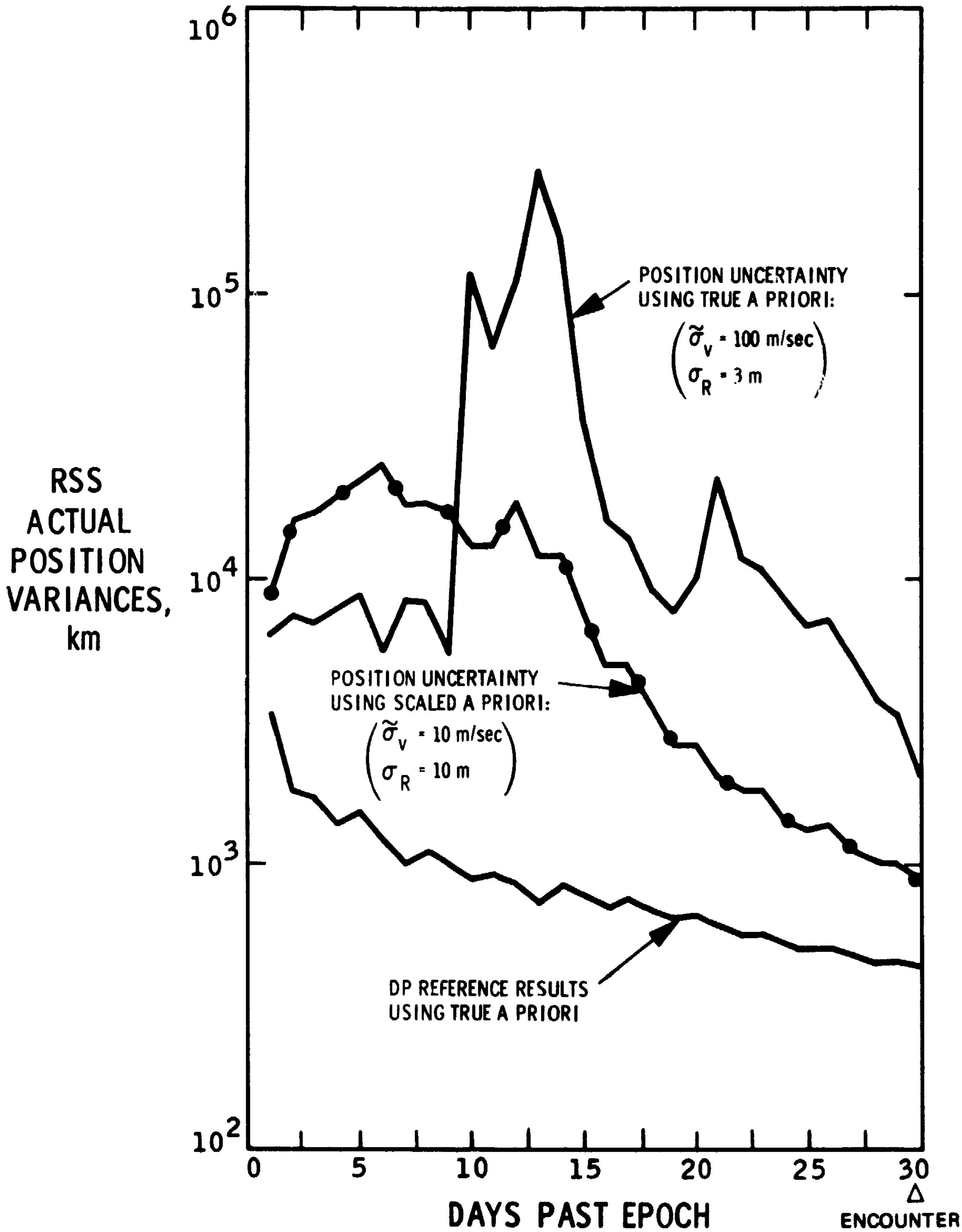
* CASE 1, 19-STATE MODEL WITH CORRECT A PRIORI

6. COMPARISON OF PATCHED AND STABILIZED ACTUAL POSITION UNCERTAINTIES*



* CASE 1, 19-STATE MODEL WITH CORRECT A PRIORI

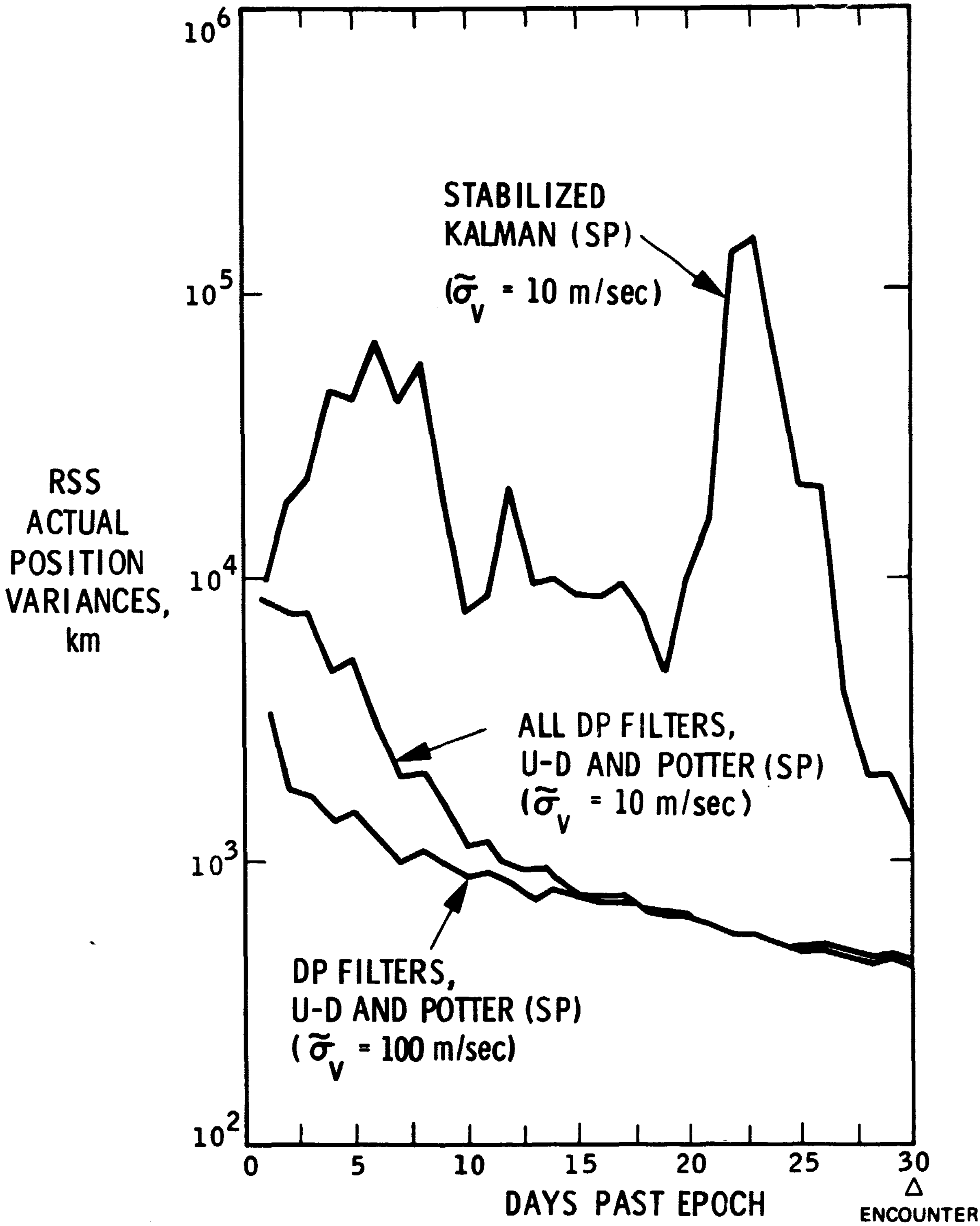
7. PERFORMANCE COMPARISONS* OF SP STABILIZED KALMAN FILTERS USING SCALED A PRIORI STATISTICS



*CASE 1, 19-STATE MODEL WITH CORRECT A PRIORI

8. COMPARISON OF ACTUAL POSITION UNCERTAINTIES

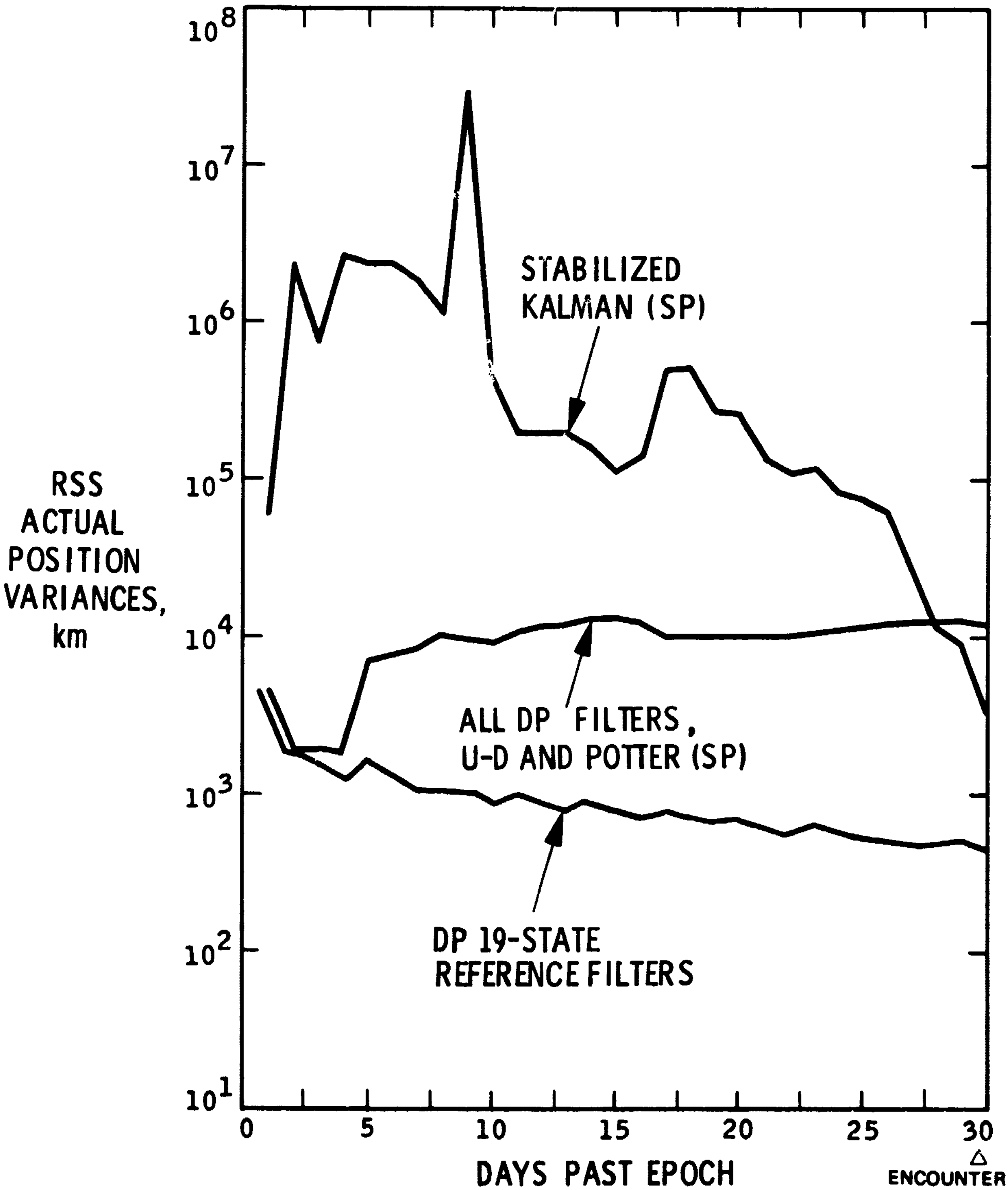
SCALED VELOCITY A PRIORI*



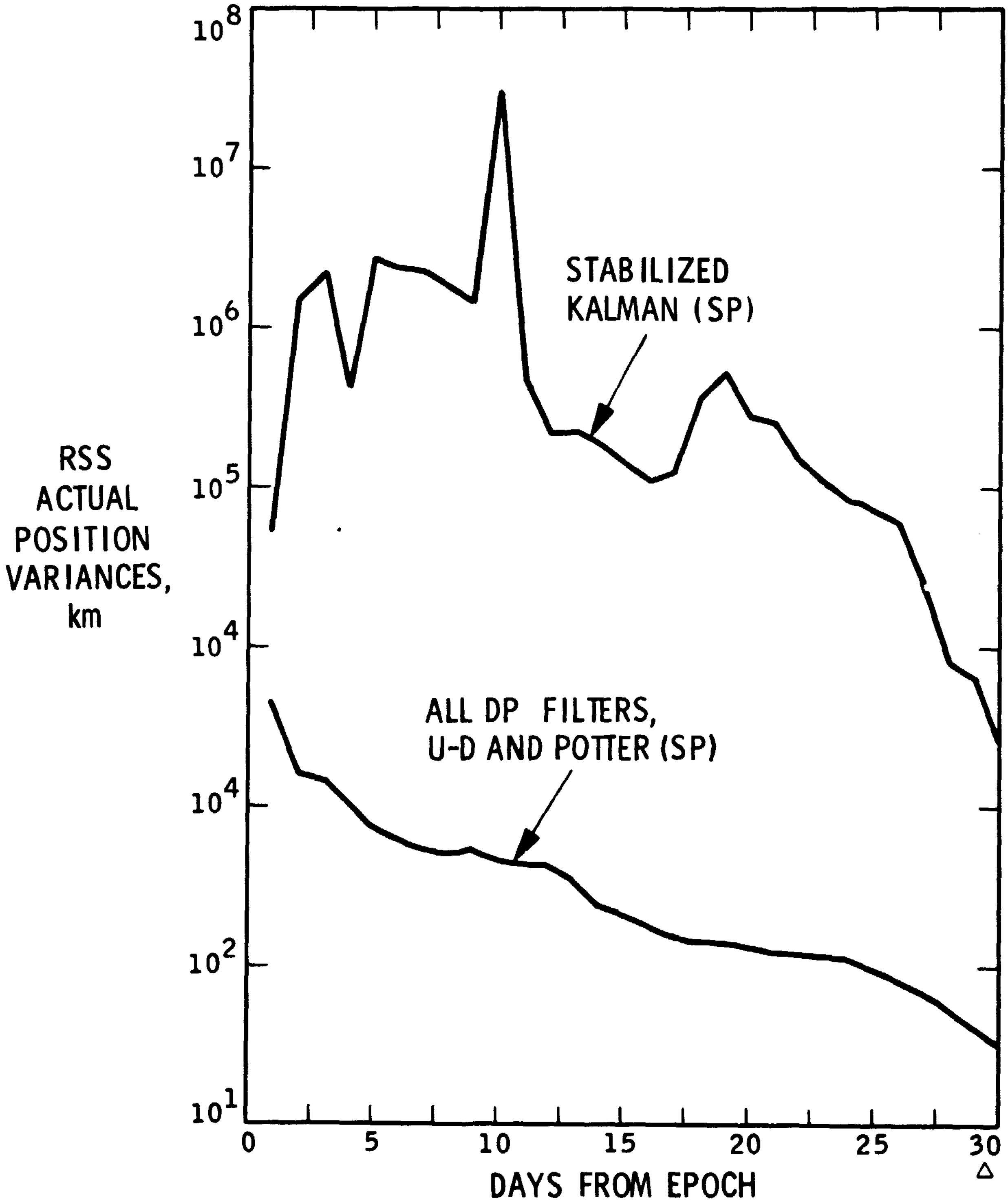
* 19-STATE FILTER MODEL WITH VELOCITY A PRIORI SCALED DOWN FROM 100 m/sec to 10 m/sec

9. COMPARISON OF ACTUAL POSITION UNCERTAINTIES

6-STATE FILTER EVALUATED FOR 19-STATE MODEL

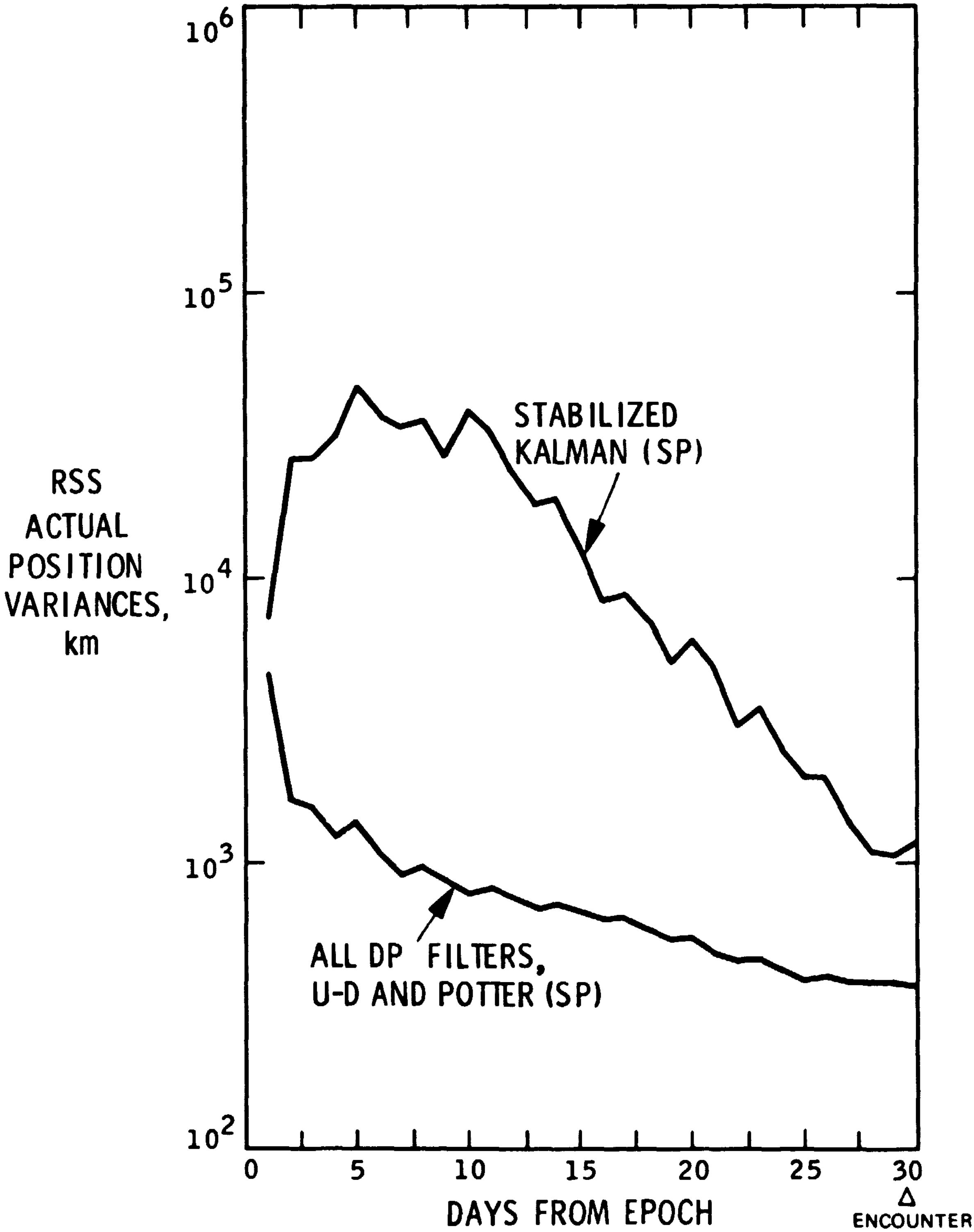


10. COMPARISON OF ACTUAL POSITION UNCERTAINTIES
6-STATE FILTER EVALUATED FOR 6-STATE MODEL



11. COMPARISON OF ACTUAL POSITION UNCERTAINTIES

9-STATE FILTER EVALUATED FOR 9-STATE MODEL



NASA—JPL—Coml., L.A., Calif.