ISSN: 2692-5079

III. Computational Requirements

It is then established that the Lainiotis Information filter equations are derived by the Lainiotis filter equations. Thus the two filters are equivalent with respect to their behavior, since they calculate theoretically the same estimates. The two filters are iterative algorithms; then, it is reasonable to assume that both the Lainiotis filter and the Information filter compute the estimation x(k/k) and the estimation error covariance P(k/k) executing the same number of iterations. Thus, in order to compare the algorithms with respect to their computational time, we have to compare their per step (iteration) calculation burden (CB) required for the on-line calculations; the calculation burden of the off-line calculations (initialization process for time invariant filters) is not taken into account.

Scalar operations are involved in matrix manipulation operations, which are needed for the implementation of the filtering algorithms. Table I summarizes the calculation burden of needed matrix operations. Note that the identity matrix is denoted by I and a symmetric matrix by S. The details are given in [2].

The per iteration calculation burdens of Lainiotis filter and Lainiotis Information filter are calculated using Table I and summarized in Table II.

From Table II, it is clear that the algorithms' calculation burdens depend on the state vector dimension n and the measurement vector dimension m.

From table II we conclude that the selection of the faster implementation depends on the relationship between n and m.

TABLE I. Calculation Burden Of Matrix Operations

Matrix Operation	**Matrix Dimensions**	***Calculation Burden***
$C = A + B$	$(nxm)+(nxm)$	nm
$S = A + B$	$(nxn)+(nxn)$	$(n^2+n)/2$
$B = I + A$	$(nxn)+(nxn)$	n
$C = A \cdot B$	$(nxm)\cdot(mxl)$	$2nml–nl$
$S = A \cdot B$	$(nxm)\cdot(mxn)$	$n^2m+nm–(n^2+n)/2$
$B = A^{-1}$	nxn	$(16n^3–3n^2–n)/6$

TABLE II. Per Iteration Calculation Burden Of Lainiotis Filter (LF) and Lainiotis Information Filter (LIF)

System	***Filter***	***Calculation Burden***
Time Varying	LF	$CB_{TVLF} = (41n^3–12n^2+n)/3 +(16m^3–3m^2–m)/6 +15n^2m–2nm+3nm^2$
	LIF	$CB_{TVLIF} = (56n^3–6n^2–3n)/3 +(16m^3–3m^2–m)/3 +13n^2m–3nm+5nm^2$
Time Invariant	LF	$CB_{TILF} = (58n^3–3n^2–n)/6 +2n^2m+2n$
	LIF	$CB_{TILIF} = (74n^3+12n^2–8n)/6 +2nm$

For time varying systems, we have:

$$CB_{TVLIF}–CB_{TVLF} = (15n^3+6n^2–4n)/3+(16m^3–3m^2–m)/6 –(2n^2m+nm–2nm^2)$$

Then, for time varying systems, the areas (depending on the model dimensions) where the Lainiotis Information Filter or the traditional Lainiotis Filter is faster, are shown in Fig. 1. The following Rule of Thumb is derived: Lainiotis Information Filter is faster than Lainiotis Filter, when $m/n<1.12$.

For time invariant systems, we have

$$CB_{TILIF}–CB_{TILF} = (16n^3+15n^2–7n)/6 –2n^2m$$

Then, for time invariant systems, the areas (depending on the model dimensions) where the Lainiotis Information Filter or the traditional Lainiotis Filter is faster, are shown in Fig. 2. The following Rule of Thumb is derived: Lainiotis Information Filter is faster than Lainiotis Filter, when $m/n>1.35$.

Finally, from Table II and Table II in [3] we conclude that Lainiotis Information filter may be faster than (Kalman) Information filter, depending on the state vector dimension n and the measurement vector dimension m.

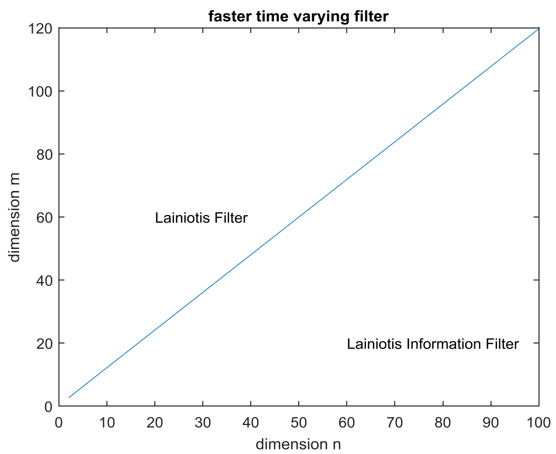

Fig. 1. The faster time varying filter

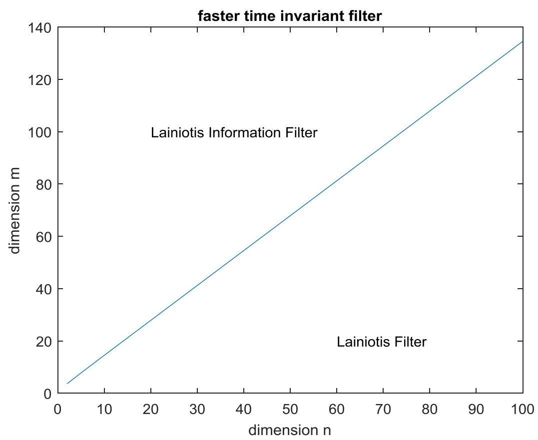

Fig. 2. The faster time invariant filter

ISSN: 2692-5079

IV. Application to Riccati Equation

We consider the **time invariant case**. Concerning the estimation error covariance matrix, the ***algebraic*** **Riccati equation emanating from the Time Invariant Lainiotis Filter** becomes:

$$P = P_n + F_n\,[I+PO_n]^{-1}PF_n^{\mathrm{T}} \quad (23)$$

The classical algorithm to solve the algebraic Riccati equation emanating from the Time Invariant Lainiotis Filter equation [9]-[10] is to implement iteratively the **Riccati equation emanating from the Time Invariant Lainiotis Filter (RETILF)**:

$$P(k+1/k+1) = P_n + F_n[I+P(k/k)O_n]^{-1}P(k/k)F_n^{\mathrm{T}} \quad (24)$$

for k=0,1,… with initial condition $P(0/0)=P_0$ until $||P(k+1/k+1)-P(k/k)||<\varepsilon$ (convergence criterion).

The ***algebraic*** **Riccati equation emanating from the Time Invariant Lainiotis Information Filter** becomes:

$$S = \mathrm{B} - \Gamma[S+\mathrm{E}]^{-1}\Gamma^{\mathrm{T}} \quad (25)$$

The algorithm to solve the Riccati equation emanating from the Time Invariant Lainiotis Information Filter equation is to implement iteratively the **Riccati equation emanating from the Time Invariant Information Lainiotis Filter (RETILIF)**:

$$S(k+1/k+1) = \mathrm{B} - \Gamma[S(k/k)+\mathrm{E}]^{-1}\Gamma^{\mathrm{T}} \quad (26)$$

for k=0,1,… with initial condition $S(0/0)=S_0=P_0^{-1}$ until $||S(k+1/k+1)-S(k/k)||<\varepsilon$ (convergence criterion).

The two algorithms are iterative algorithms; then, it is reasonable to assume that the first algorithm computes the steady state estimation error covariance P and the second algorithm computes the inverse $S=P^{-1}$ of the steady state estimation error covariance, executing the same number of iterations. Thus, in order to compare the algorithms with respect to their computational time, we have to compare their per iteration calculation burden (CB) required for the on-line calculations; the calculation burden of the off-line calculations is not taken into account. The per iteration calculation burdens of the algorithms that solve the Riccati equation emanating from Lainiotis filter and Lainiotis Information filter are calculated using Table I and summarized in Table III.

From Table III, it is clear that the algorithms' calculation burdens depend on the state vector dimension n. From table II we conclude that the Lainiotis Information Filter provides a faster method than the classical one to solve the Riccati equation emanating from Lainiotis filter, since

$$CB_{RETILF}-CB_{RETILIF} = (2n^3-3n^2+3n)/6 > 0$$

TABLE III. Per Iteration Calculation Burden Of Riccati Equation Solution Algorithms

Filter	***Calculation Burden***
LF	$CB_{RETILF} = (52n^3-6n^2+2n)/6$
LIF	$CB_{RETILIF} = (50\,n^3-3n^2-n)/6$

V. Conclusions

Kalman filter and Lainiotis filter are well known algorithms that solve the filtering problem, producing the state estimation as well as the corresponding estimation error covariance matrix. Kalman filter and Lainiotis filter are equivalent with respect to their behavior, since they produce the same estimations [2].

Using the Information estimation error covariance matrix, which is the inverse of the estimation error covariance matrix, the (Kalman) Information filter has been derived from Kalman filter. Also, Kalman filter and (Kalman) Information Filter are equivalent [1].

In this paper, using the Information estimation error covariance matrix, the Lainiotis Information filter is introduced. The Lainiotis filter and the Lainiotis Information filter are equivalent with respect to their behavior, since they produce the same estimations.

The computational requirements of the Lainiotis filter and the Lainiotis Information filter are determined and a method is proposed to a-priori (before the filters' implementation) decide which filter is the faster one.

In the time invariant systems case, the Lainiotis Information filter provides a faster method than the classical one to solve the Riccati equation emanating from Lainiotis filter.

References

[1] B. D. O. Anderson and J. B. Moore, Optimal Filtering, Dover Publications, New York, 2005.

[2] N. Assimakis and M. Adam, "Discrete time Kalman and Lainiotis filters comparison", Int. Journal of Mathematical Analysis (IJMA), vol. 1(13), pp. 635-659, 2007.

[3] N. Assimakis, M. Adam, A. Douladiris, "Information Filter and Kalman Filter Comparison: Selection of the Faster Filter", International Journal of Information Engineering, vol. 2(1), pp. 1-5, 2012.

[4] M. E. Khan, Matrix Inversion Lemma and Information Filter, [Online]. Available: https://emtiyaz.github.io/Writings/MILandIF.pdf

[5] T.G. Phillips, M.E. Green and P.R. McAree, "An Adaptive Structure Filter for Sensor Registration from Unstructured Terrain", Journal of Field Robotics, vol. 32(5), pp. 748-774, 2015.

[6] Behrouz Safarinejadian and Mojtaba Asad, "Fractional Order State Space Canonical Model Identification Using Fractional Order Information Filter", 2015 International Symposium on Artificial Intelligence and Signal Processing (AISP), IEEE, pp. 65-70, 2015. doi:10.1109/AISP.2015.7123479

[7] M.C.P. Santos, L.V. Santana, A.S. Brandão , M. Sarcinelli-Filho, R. Carelli, "Indoor low-cost localization system for controlling aerial robots", Control Engineering Practice, vol. 61, pp. 93-111, 2017. doi.org/10.1016/j.conengprac.2017.01.011

[8] A. M. Shahri and R. Rasoulzadeh, "Implementation of a Low- Cost Multi- IMU by Using Information Form of a Steady State Kalman Filter", AUT Journal of Elerctrical Engineering, vol. 49(2), pp. 195-204, 2017. doi: 10.22060/eej.2017.12045.5028.

[9] D. G. Lainiotis, N. D. Assimakis, S. K. Katsikas, "A new computationally effective algorithm for solving the discrete Riccati equation", Journal of Mathematical Analysis and Applications, vol. 186(3), pp. 868-895, 1994.

[10] D. G. Lainiotis, N. D. Assimakis, S. K. Katsikas, "Fast and numerically robust recursive algorithms for solving the discrete time Riccati equation: The case of nonsingular plant noise covariance matrix", Neural, Parallel and Scientific Computations, vol. 3(4), pp. 565-584, 1995.