Send Orders for Reprints to reprints@benthamscience.ae

The Open Mathematics Journal, 2015, *8*, 1-6

Open Access

Mobile Position Tracking in Three Dimensions using Kalman and Lainiotis Filters

Nicholas Assimakis[1] and Maria Adam[2,*]

[1]*Department of Electronic Engineering, Technological Educational Institute of Central Greece, 3rd km Old National Road Lamia-Athens, P.O. 35100 Lamia, Greece,* [2]*Department of Computer Science and Biomedical Informatics, University of Thessaly, 2-4 Papasiopoulou str., P.O. 35100 Lamia, Greece*

Abstract: In this paper we present two time invariant models mobile position tracking in three dimensions, which describe the movement in x-axis, y-axis and z-axis simultaneously or separately, provided that there exist measurements for the three axes. We present the time invariant filters as well as the steady state filters: the classical Kalman Filter and Lainiotis Filter and the Join Kalman Lainiotis Filter, which consists of the parallel usage of the two classical filters. Various implementations are proposed and compared with respect to their behavior and to their computational burden: all time invariant and steady state filters have the same behavior using both the proposed models but have different computational burden.

Keywords: GPS, Kalman Filter, Lainiotis Filter, mobile position tracking, MPT.

1. INTRODUCTION

The Global Positioning System (GPS) is the most popular positioning technique in navigation providing reliable mobile location estimates in many applications [1-4]. Thus wireless location systems offering reliable mobile location estimates have been studied by researchers and engineers over the past few years. Various techniques require one base station or at least two base stations or more than three base stations in order to determine the location of the user. The accuracy of the positioning results is affected by many interference sources as the signals propagate in the atmosphere. So, techniques were developed using filters to estimate the location of the user through the location information exchanged between the handset and the base station. Kalman Filter has been used in the localization process [4-7], due to the following advantages mentioned in [6]: (a) Kalman Filter [5, 8-10] processes noisy measurements and so it can smooth out the effects of noise in the estimated state variables by integrating more information from reliable data more than unreliable data and (b) Kalman Filter allows the combination of measurements from different sources (locomotion data) and different times. Kalman Filter was implemented for Global Systems for Mobile (GSM) position tracking in [6]: Kalman Filter was used for tracking in two dimensions and it was stated that Kalman Filter is very powerful due to its reliable performance, because it yielded enhanced position tracking results. Also, in [11] two models for GSM position tracking were used in order to describe the movement in x-axis and y-axis simultaneously or separately.

*Address correspondence to this author at the Department of Computer Science and Biomedical Informatics, University of Thessaly, 2-4 Papasiopoulou str., P.O. 35100 Lamia, Greece; Tel: +30 2231066901;
Fax: +30 2231066939; E-mail: madam@dib.uth.gr

In this paper we extend the ideas in [6] and [11] by using two models for Mobile Position Tracking in three dimensions (3D-MPT), which describe the movement in x-axis, y-axis and z-axis simultaneously or separately and by using the Kalman Filter and the Lainiotis Filter [9, 12]. This approach holds under the condition that there exist measurements for the three axes. The paper is organized as follows: In Section 2, we present two time invariant models for Mobile Position Tracking (MPT), which describe the movement in x-axis, y-axis and z-axis. In Section 3, we present the time invariant filters: Kalman Filter, Lainiotis Filter and Join Kalman Lainiotis Filter. In Section 4, we present the corresponding steady state filters. In Section 5, various implementations are proposed. In Section 6, we compare the filters with respect to their behavior and to their computational burden. Finally, Section 7 summarizes the conclusions.

2. TIME INVARIANT MODELS

Linear estimation is associated with time invariant systems described by the following state space equations:

$$x(k+1) = Fx(k) + Gw(k)$$

$$z(k) = Hx(k) + v(k)$$

for $k \geq 0$, where $x(k)$ is the n-dimensional state vector at time k, $z(k)$ is the m-dimensional measurement vector at time k, F is the $n \times n$ system transition matrix, H is the $m \times n$ output matrix, $w(k)$ is the plant noise at time k, $v(k)$ is the measurement noise at time k. Also, $\{w(k)\}$ and $\{v(k)\}$ are Gaussian zero-mean white random processes with covariance matrices Q and R, respectively. The initial state $x(0)$ is a Gaussian random variable with mean x_0 and covariance P_0 and is assumed to be independent of $w(k)$ and $v(k)$.

In this paper we consider two models:

Model A

The first model (model A) describes the movement in x-axis, y-axis and z-axis simultaneously and follows the ideas in [6]. The state vector is of dimension $n=6$ and contains the position and the velocity in x-axis, y-axis and z-axis: $x(k)=\begin{bmatrix} s_x(k) & \upsilon_x(k) & s_y(k) & \upsilon_y(k) & s_z(k) & \upsilon_z(k) \end{bmatrix}^T$. The measurement vector is of dimension $m=3$ and contains the measured position in x-axis, y-axis and z-axis: $z(k)=\begin{bmatrix} z_x(k) & z_y(k) & z_z(k) \end{bmatrix}^T$. Then we take:

$$F=\begin{bmatrix} 1 & \Delta t & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & \Delta t & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$G=\begin{bmatrix} \frac{1}{2}\Delta t & 0 & 0 \\ 1 & 0 & 0 \\ 0 & \frac{1}{2}\Delta t & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{2}\Delta t \\ 0 & 0 & 1 \end{bmatrix},$$

$$\text{and } \mathrm{H}=\begin{bmatrix} 1 & \Delta t & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & \Delta t & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \Delta t \end{bmatrix}.$$

The plant noise $w(k)=\begin{bmatrix} w_x(k) & w_y(k) & w_z(k) \end{bmatrix}^T$ is Gaussian zero-mean with covariance matrix

$$Q=\begin{bmatrix} \sigma_{xq}^2 & 0 & 0 \\ 0 & \sigma_{yq}^2 & 0 \\ 0 & 0 & \sigma_{zq}^2 \end{bmatrix}.$$

The measurement noise $v(k)=\begin{bmatrix} v_x(k) & v_y(k) & v_z(k) \end{bmatrix}^T$ is Gaussian zero-mean with covariance matrix

$$R=\begin{bmatrix} \sigma_{xr}^2 & 0 & 0 \\ 0 & \sigma_{yr}^2 & 0 \\ 0 & 0 & \sigma_{zr}^2 \end{bmatrix}.$$

Model B

The second model (model B) describes the movement in x-axis, y-axis and z-axis separately. In each axis, the state vector is of dimension $n=2$ and contains the position and the velocity: $x(k)=\begin{bmatrix} s(k) & \upsilon(k) \end{bmatrix}^T$. The measurement vector is of dimension $m=1$ and contains the measured position $z(k)$. Then we take:

$$F=\begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix},\ G=\begin{bmatrix} \frac{1}{2}\Delta t \\ 1 \end{bmatrix}, \text{ and } \mathrm{H}=\begin{bmatrix} 1 & \Delta t \end{bmatrix}.$$

The plant noise $w(k)$ is Gaussian zero-mean with covariance matrix $Q=\sigma_q^2$.

The measurement noise $v(k)$ is Gaussian zero-mean with covariance matrix $R=\sigma_r^2$.

It is obvious that we are able to describe the movement in three axis using three separate state vectors: $x_x(k)=\begin{bmatrix} s_x(k) & \upsilon_x(k) \end{bmatrix}^T$ for the x-axis, $x_y(k)=\begin{bmatrix} s_y(k) & \upsilon_y(k) \end{bmatrix}^T$ for the y-axis and $x_z(k)=\begin{bmatrix} s_z(k) & \upsilon_z(k) \end{bmatrix}^T$ for the z-axis. If we merge these three state vectors, we take the state vector $x(k)=\begin{bmatrix} s_x(k) & \upsilon_x(k) & s_y(k) & \upsilon_y(k) & s_z(k) & \upsilon_z(k) \end{bmatrix}^T$ of model A.

3. TIME INVARIANT KALMAN AND LAINIOTIS FILTERS

In this section, we present the classical time invariant Kalman Filter [8-10] and Lainiotis Filter [9, 12] which are the most well-known algorithms that solves the filtering problem. Both algorithms compute the estimation $x(k/k)$ and the corresponding estimation error covariance $P(k/k)$. We also propose the Join Kalman-Lainiotis Filter, which consists of the parallel usage of *two* filters (one Kalman Filter and one Lainiotis Filter) with the same measurements and combination of the results (weight 50% for each filter).

Kalman Filter (KF)

The following equations constitute the KF:

$$\begin{aligned} &x(k+1/k)=Fx(k/k) \\ &P(k+1/k)=GQG^T+FP(k/k)F^T \\ &K(k+1)=P(k+1/k)H^T[HP(k+1/k)H^T+R]^{-1} \\ &x(k+1/k+1)=[I-K(k+1)H]x(k+1/k)+K(k+1)z(k+1) \\ &P(k+1/k+1)=[I-K(k+1)H]P(k+1/k) \end{aligned} \quad (1)$$

for $k\geq 0$, with initial conditions $x(0/0)=x_0$ and $P(0/0)=P_0$.

The Kalman Filter computes the estimation $x(k/k)$ and the estimation error covariance $P(k/k)$ through the prediction $x(k+1/k)$ and the corresponding prediction error covariance $P(k+1/k)$ using the Kalman Filter gain $K(k)$.

Lainiotis Filter (LF)

The following equations constitute the LF:

$$\begin{aligned} x(k+1/k+1)=&K_n z(k+1)+F_n[I+P(k/k)O_n]^{-1} \\ &[P(k/k)K_m z(k+1)+x(k/k)] \end{aligned} \quad (2)$$
$$P(k+1/k+1)=P_n+F_n[I+P(k/k)O_n]^{-1}P(k/k)F_n^T,$$

for $k\geq 0$, with initial conditions $x(0/0)=x_0$ and $P(0/0)=P_0$, where

$$
\begin{aligned}
A &= [HGQG^T H^T + R]^{-1} \\
K_n &= GQG^T H^T A \\
K_m &= F^T H^T A \\
P_n &= GQG^T - K_n HGQG^T \\
F_n &= F - K_n HF \\
O_n &= F^T H^T AHF.
\end{aligned} \tag{3}
$$

Join Kalman-Lainiotis Filter (JKLF)

The filter consists of the parallel usage of *two* filters (one Kalman Filter and one Lainiotis Filter) with the same measurements and combination of the results (weight 50% for each filter). In fact,

$$x(k/k) = \frac{1}{2}[x_{KF}(k/k) + x_{LF}(k/k)] \tag{4}$$

and

$$P(k/k) = \frac{1}{2}[P_{KF}(k/k) + P_{LF}(k/k)] \ , \tag{5}$$

for $k \geq 0$, where the estimation $x_{KF}(k/k)$ and the estimation error covariance $P_{KF}(k/k)$ are computed by the associated equations in (1) and $x_{LF}(k/k)$, $P_{LF}(k/k)$ by (2).

4. STEADY STATE KALMAN AND LAINIOTIS FILTERS

For time invariant systems, it is well known [8] that there exists a steady state value P_p of the prediction error covariance matrix, if the signal process model is asymptotically stable, or if the signal process model is not necessarily asymptotically stable, but the pair $[F, H]$ is completely detectable and the pair $[F, GG_1]$ is completely stabilizable for any G_1 with $G_1 G_1^T = Q$. Then, there also exist a steady state value P_e of the estimation error covariance matrix and a steady state value K of the Kalman Filter gain.

In this section we present the steady state Kalman Filter and Lainiotis Filter. Both algorithms compute the estimation $x(k/k)$ using the previous estimation and the current measurement. We also propose the Join Steady State Kalman-Lainiotis Filter, which consists of the parallel usage of *two* filters (one Steady State Kalman Filter and one Steady State Lainiotis Filter) with the same measurements and combination of the results (weight 50% for each filter).

Steady State Kalman Filter (SSKF)

The following equation constitutes the SSKF:

$$x(k+1/k+1) = A_{KF} x(k/k) + B_{KF} z(k+1) \tag{6}$$

for $k \geq 0$, with initial condition $x(0/0) = x_0$, where

$$A_{KF} = [I - KH]F, \text{ and } B_{KF} = K. \tag{7}$$

The steady state Kalman Filter gain K is computed by

$$K = P_p H^T [HP_p H^T + R]^{-1}, \tag{8}$$

where P_p is the steady state prediction error covariance computed by solving the *Riccati Equation emanating from Kalman Filter (REKF)*

$$P_p = (GQG^T) + FP_p F^T - FP_p H^T [HP_p H^T + R]^{-1} HP_p F^T. \tag{9}$$

In view of the importance of the Riccati equation emanating from Kalman Filter, there exists considerable literature on its algebraic solutions [8, 13] or iterative solutions [8, 14-17] concerning per step or doubling algorithms.

Steady State Lainiotis Filter (SSLF)

The following equation constitutes the SSLF:

$$x(k+1/k+1) = A_{LF} x(k/k) + B_{LF} z(k+1) \tag{10}$$

for $k \geq 0$, with initial condition $x(0/0) = x_0$, where

$$A_{LF} = F_n [I + P_e O_n]^{-1}, \ B_{LF} = K_n + F_n [I + P_e O_n]^{-1} P_e K_m, \tag{11}$$

and P_e is the steady state estimation error covariance computed by solving the *Riccati Equation emanating from Lainiotis Filter (RELF)*

$$P_e = P_n + F_n [I + P_e O_n]^{-1} P_e F_n^T. \tag{12}$$

In view of the importance of the Riccati equation emanating from Lainiotis Filter, there exists considerable literature on its algebraic or iterative solutions [14, 16-18] concerning per step or doubling algorithms.

Note that in [9], it is shown that SSKF is equivalent to SSLF, since

$A_{KF} = A_{LF}$ and $B_{KF} = B_{LF}$.

Join Steady State Kalman-Lainiotis Filter (JSSKLF)

The filter consists of the parallel usage of *two* steady state filters (one Steady State Kalman Filter and one Steady State Lainiotis Filter) with the same measurements and combination of the results (weight 50% for each filter). In fact,

$$x(k/k) = \frac{1}{2}[x_{KF}(k/k) + x_{LF}(k/k)], \tag{13}$$

for $k \geq 0$, where the estimations $x_{KF}(k/k)$ and $x_{LF}(k/k)$ are given by the equations (6), (10), respectively.

5. IMPLEMENTATIONS

In this section, we propose various implementations.

The use of model A, which describes the movement in x-axis, y-axis and z-axis simultaneously requires the use one filter; we are able to use KF/LF/SSKF/SSLF/JKLF in order to compute the estimation and the corresponding estimation error covariance.

The use of model B, which describes the movement in x-axis, y-axis and z-axis separately requires the use of two filters KF/LF/SSKF/SSLF/JSSKLF in order to compute the estimation and the corresponding estimation error covariance for each movement. It is obvious that, if we merge the estimation $x_x(k/k) = \begin{bmatrix} s_x(k/k) & \upsilon_x(k/k) \end{bmatrix}^T$ for the movement in x-axis, the estimation

$x_y(k/k) = \begin{bmatrix} s_y(k/k) & \upsilon_y(k/k) \end{bmatrix}^T$ for the movement in y-axis and the estimation

$x_z(k/k) = \begin{bmatrix} s_z(k/k) & \upsilon_z(k/k) \end{bmatrix}^T$ for the movement in z-axis, we take the state vector of model A:

$$x(k/k) = \begin{bmatrix} s_x(k/k) & \upsilon_x(k/k) & s_y(k/k) & \upsilon_y(k/k) & s_z(k/k) & \upsilon_z(k/k) \end{bmatrix}^T$$
$$= \begin{bmatrix} x_x(k/k) & x_y(k/k) & x_z(k/k) \end{bmatrix}^T.$$

Also, the estimation error covariance matrices $P_x(k/k)$, $P_y(k/k)$ and $P_z(k/k)$ for each movement can be merged to the estimation error covariance of model A:

$$P(k/k) = \begin{bmatrix} P_x(k/k) & 0 & 0 \\ 0 & P_y(k/k) & 0 \\ 0 & 0 & P_z(k/k) \end{bmatrix}.$$

Thus, we propose various implementations for Mobile Position Tracking in three dimensions (3D- MPT), as it is appeared in Table **1**.

Table 1. Implementations for 3D- MPT.

Implementation	Model	System	Filter
1	Model A	Time invariant	KF
2			LF
3			JKLF
4		Steady state	SSKF
5			SSLF
6			JSSKLF
7	Model B	Time invariant	KF
8			LF
9			JKLF
10		Steady state	SSKF
11			SSLF
12			JSSKLF

6. COMPARISON OF THE FILTERS

In this section, we compare the filters with respect to their behaviour and to their computational burden.

Example.

We implemented the filters with the following parametes:

- discretization factor: $\Delta t = 1$,
- movement reliability: $\sigma_{xq}^2 = \sigma_{yq}^2 = \sigma_{zq}^2 = 0.01$,
- measurements reliability: $\sigma_{xr}^2 = \sigma_{yr}^2 = \sigma_{zr}^2 = 0.1$,
- initial conditions: $x_0 = 0$ and $P_0 = 0$.

Concerning the behaviour of the filters, we found that:

- the time invariant filters KF, LF and JKLF are equivalent, since they compute the same outputs (estimation and estimation error covariance), using model A or model B
- the steady state filters SSKF, SSLF and JSSKLF are equivalent, since they compute the same outputs (estimation and estimation error covariance), using model A or model B,
- the steady state filters and the time invariant filters compute outputs very close to each other,
- model A is equivalent to model B, since they produce the same outputs.

These results are depicted in Fig. (**1**).

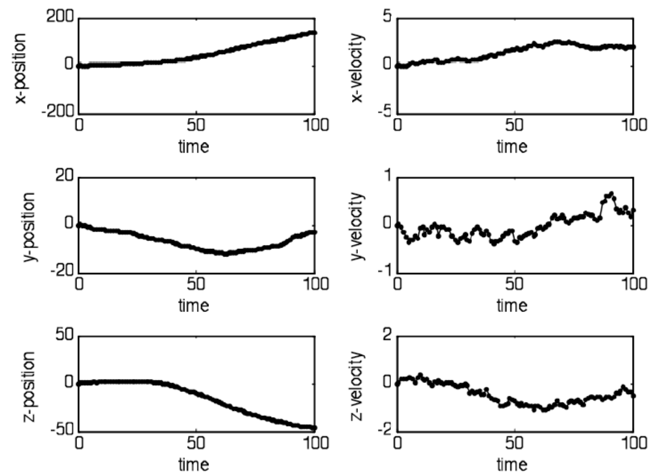

Fig. (1). Position and velocity estimation solid line: KF/LF/JKLF, dashed line: SSKF/SSLF/JSSKLF.

Concerning the computational burden of the filters, we cmpared the filters with respect to their per-iteration calculation burdens, computed using the ideas in [9], as shown in Table **2**.

Table 2. Per-iteration calculation burden of filters.

KF	$4n^3 + 3.5n^2 - 1.5n + 4n^2m + nm + 3nm^2 + (16m^3 - 3m^2 - m)/6$
LF	$4nm + (58n^3 + 9n^2 - 7n)/6$
JKLF	$n^2 + 3n$ (join procedure)
SSKF	$2n^2 + 2nm - n$
SSLF	$2n^2 + 2nm - n$
JSSKLF	$2n$ (join procedure)

Table **3** summarizes the per-iteration calculation burden of all implementations, using model A and model B.

We observe that:

- KF is faster than LF

  speedup(LF model A to KF model A)=1.330

  speedup(LF model B to KF model B)=1.290
- model B is faster than model A

Table 3. Per-iteration calculation burden of implementations.

Implementation	Model	System	Filter	Calculation Burden
1	Model A	Time invariant	KF	1660
2			LF	2207
3			JKLF	3921
4		Steady state	SSKF	102
5			SSLF	102
6			JSSKLF	216
7	Model B	Time invariant	KF	207
8			LF	267
9			JKLF	484
10		Steady state	SSKF	30
11			SSLF	30
12			JSSKLF	64

speedup(KF model A to SSKF model B)=55.333

speedup(LF model A to SSLF model B)=73.567

CONCLUSION

In this paper we presented two time invariant models for Mobile Position Tracking in three dimensions (3D- MPT), which describe the movement in x -axis, y -axis and z -axis simultaneously or separately, provided that there exist measurements for the three axes. We presented the time invariant filters as well as the steady state filters: the classical Kalman Filter and Lainiotis Filter and the Join Kalman Lainiotis Filter, which consists of the parallel usage of the two classical filters. Various implementations are proposed and compared with respect to their behavior and to their computational burden. We found that all time invariant and steady state filters have the same behavior using both the proposed models. We found that: (a) Kalman Filter is faster than Lainiotis Filter, (b) Join Kalman Lainiotis Filter is slower than both Kalman Filter and Lainiotis Filter, (c) steady state filters are faster than time invariant filters and (d) the filters using the model, which handles the movement in x -axis, y -axis and z -axis separately, are faster than the same filters using the model, which handles the movement in x -axis, y -axis and z -axis simultaneously.

CONFLICT OF INTEREST

The authors confirm that this article content has no conflicts of interest.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

[1] Drane C, Macnaughtan M, Scott C. Positioning GSM telephones. IEEE Communications Magazine 1998; 36(4): 46-54.

[2] Küpper A. Location-based services. Fundamentals and operation. John Wiley and Sons 2005.

[3] Mouly M, Pautet M. The GSM system for mobile communications. Washington DC: Telecom Publishing 1992.

[4] Rooney S, Chippendale P, Choony R, Le Roux C, Honary B. Accurate vehicular positioning using DAB-GSM hybrid system, IEEE 51st Conference Proceedings of VTC 2000-Spring. Tokyo: 2000; vol. 1: pp. 97-101.

[5] Andreu JB. Tracking a mobile receiver using the unscented kalman filter. Master Thesis: University of Freiburg 2013.

[6] Dubois JP, Daba JS, Nader M, El Ferkh C. GSM position tracking using a Kalman Filter. World Academy of Science, Engineering and Technology 2012; 68: pp. 1610-19.

[7] Kos T, Grgic M, Sisul G. Mobile user positioning in GSM/UMTS cellular networks, in multimedia signal processing and communications, Conference Proceedings 48th International Symposium ELMAR-2006: 2006; pp. 185-8.

[8] Anderson BDO, Moore JB. Optimal Filtering. New York: Dover Publications 2005.

[9] Assimakis N, Adam M. Discrete time Kalman and Lainiotis filters comparison. IJMA 2007; 1(13): 635-59.

[10] Kalman RE. A new approach to linear filtering and prediction problems. J Basic Eng-T ASME, (series D) 1960; 82(1): 34-45.

[11] Assimakis N, Adam M. Global systems for mobile position tracking using Kalman and Lainiotis filters. Scientific World Journal 2014, Article ID 130512: 1-8. [Available from: http://dx.doi.org/10.1155/2014/130512]

[12] Lainiotis DG. Partitioned linear estimation algorithms: discrete case. IEEE Trans Automat Contr 1975; AC-20: 255-7.

[13] Vaughan DR. A nonrecursive algebraic solution for the discrete time Riccati equation. IEEE Trans Automat Contr 1970; 15(5): 597-9.

[14] Assimakis N, Lainiotis D, Katsikas S, Sanida F. A survey of recursive algorithms for the solution of the discrete time Riccati equation. Nonlinear Anal Theor 1997; 30(4): 2409-20.

[15] Assimakis N, Roulis S, Lainiotis D. Recursive solutions of the discrete time Riccati equation. Neural Parallel Sci Comput 2003; 11: 343-50.

[16] Lainiotis DG, Assimakis ND, Katsikas SK. A new computationally effective algorithm for solving the discrete Riccati equation. J Math Anal Appl 1994; 186(3): 868-95.

[17] Lainiotis DG, Assimakis ND, Katsikas SK. Fast and numerically robust recursive algorithms for solving the discrete time Riccati equation: the case of nonsingular plant noise covariance matrix. Neural Parallel Sci Comput 1995; 3(4): 565-84.

[18] Lainiotis DG. Discrete Riccati equation solutions: Partitioned algorithms. IEEE Trans Autom Control 1975; AC-20: 555-6.

Received: March 01, 2014 Revised: November 18, 2014 Accepted: December 10, 2014

© Assimakis and Adam; Licensee *Bentham Open.*

This is an open access article licensed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/-licenses/by-nc/3.0/) which permits unrestricted, non-commercial use, distribution and reproduction in any medium, provided the work is properly cited.