Design of a robust Controller/Observer for TCP/AQM network: First application to intrusion detection systems for drone fleet
Thierry Miquel, Jean-Philippe Condomines, Riad Chemali, Nicolas Larrieu

To cite this version:

HAL Id: hal-01545617
https://hal-enac.archives-ouvertes.fr/hal-01545617
Submitted on 27 Jun 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Design of a robust Controller/Observer for TCP/AQM network: 
First application to intrusion detection systems for drone fleet

Thierry Miquel\textsuperscript{1}, Jean-Philippe Condomines\textsuperscript{1}, Riad Chemali\textsuperscript{2} and Nicolas Larrieu\textsuperscript{2}

Abstract—This paper proposes a robust controller/observer for UAVs network anomaly estimation which is based on both Lyapunov Krasovkii functional and dynamic behavior of TCP (Transmission Control Protocol). Several research works on network anomaly estimation have been led using automatic control techniques and provide methods for designing both observer and command laws dedicated to time delay problem while estimating the anomaly or intrusion in the system. The observer design is based on a linearized fluid-flow model of the TCP behavior and must be associated to an AQM (Active Queue Management) to perform its diagnosis. The developed robust controller/observer in this paper has to be tuned by considering the time delay linear state-space representation of TCP model. As a first result, the designed controller/observer system has been successfully applied to some relevant practical problems such as topology network for aerial vehicles and the effectiveness is illustrated by using real traffic traces including Denial of Service attacks. Our first results show promising perspectives for Intrusion Detection System (IDS) in a fleet of UAVs.

I. INTRODUCTION

The number and diversity of applications involving Unmanned Aerial Vehicles (UAVs) is growing every year. The necessary resort to use a fleet of versatile UAVs has led to an important interest from network community to design algorithms for Anomaly Detection Systems (ADS) or Intrusion Detection Systems (IDS) based on TCP (Transmission Control Protocol) network. Among abnormal patterns in TCP network \cite{1}, \cite{2}, \cite{3} such as overload, flash crowds, worms, port scans, the malicious anomalies have the worst impact on the fleet of UAV by creating congestion and reducing significantly the Quality of Service (QoS) of the whole network. This is the worst case for UAV certification and integration into civil airspace. That is the reason why, nowadays, malicious anomaly detection is an important issue. In \cite{3} an overview is provided spanning multiple research areas and application domains.

Network anomalies and security-related problems (such as Distributed Denial of Service (DDoS) attacks) is an important issue of active security threats detection. A variety of tools for anomaly detection are mainly based on data packet signature. This behavior is fully operational with well known DDoS attacks. However, this mechanism is inefficient when a new type of attack is performed. For such a reason, we are designing in this paper a new type of IDS able to detect different type of DDoS. The detection model is based on a traffic characterization analysis (performed through an automatic control estimation). All types of attacks which are not following the initial automatic control estimation trigger an alarm and consequently, the malicious traffic can be analysed in details. This automatic control based approach has the main advantage not to be associated to a specific type of attacks. Any attacks which are not following the initial model can be analysed, detected and managed. The security and performances of the whole network are then improved.

Exploiting the capabilities of observers or estimators allows, by generating consolidated signals, to extend the way malicious intrusion can be controlled while enhancing fleet of UAVs intrinsic flight handling qualities. Among the non-linear methods \cite{4} the Super-Twisting Algorithm (STA) \cite{6}, \cite{7}, \cite{8} is the most widely used for chattering avoidance while detecting anomalies. Its principles rely firstly on the non linear fluid model applied on TCP dynamics and secondly on sliding modes \cite{5} which are often used to design robust nonlinear observers or control laws. Unfortunately, building upon this peculiar observer provides for bounded input-bounded state (BIBS) finite-time stability only \cite{9} restricting the application of this observer to the class of the systems for which the upper bound of the initial condition might be estimated in advance. Such an approach can be very non-systematic for complex dynamical systems such as TCP model for a fleet of UAVs.

Another relevant method proposed in the literature is based on time delay linear state estimation. Such an approach \cite{10} is based on both Lyapunov Krasovkii functional and dynamic behavior of TCP/AQM (Transmission Control protocol/ Active Queue Management) to use a Luenberger observer to cope with anomaly detection. An Active Queue Management (AQM) consists in adjusting data flow rates sent by the UAV into the network. The principle consists in dropping (or marking when ECN, Explicit Congestion Notification \cite{11} option is enabled) some packets before buffer saturates. Consequently, the estimator must be associated to a robust AQM to perform its diagnosis. The study of congestion control in a time delay system framework is not new and has been successfully \cite{13}, \cite{14}, \cite{15}, \cite{16} exploited. A relevant constructive algorithm \cite{17} has been proposed.

The contribution of this paper is to propose a robust controller/observer algorithm able to detect traffic anomalies (i.e. DDoS). This algorithm has been designed and tested...
in real traffic conditions. Indeed, the simulink design and its theoretical evaluation have been confronted to real traffic traces. These traces have been generated thanks to an hybrid UAV network simulator. Consequently, the validation of the theoretical estimator is improved by testing its behavior with real DDoS attacks, real UAV trajectories, real UAV background traffic and real UAV fleet topology. Guided by the Lyapunov theory, we propose to develop a design to determine all correction terms associated with time delay linear state-space representation of TCP model. In the sequel, §II presents the basics of the modelling adopted to tackle the time delay linear estimation problem of determining the state vector components of a fluid-flow model fitted out with a TCP model. §III presents the theoretical background of our proposed controller/observer system. Finally, §IV gathers all the results obtained after solving the time delay linear estimation problem in real conditions.

II. DYNAMICAL SYSTEM MODELING

In order to tackle a wide range of applications, various implementations of TCP models in terms of assumptions and numerical techniques [18], [19], [20] exist. TCP network is commonly represented using a linearized fluid-flow model [18] associated with our network topology. As shown in Figure 1 this topology consists of $N$ TCP sources, with the same propagation delay connected to a destination node through a router. This simple topology is due to : 1) the high complexity behavior of a fleet of UAVs in which each UAV can be sender, receiver and router; 2) the difficulty for such systems to derive a reliable and representative network modelling from scratch.

![Diagram of Sources, Bottleneck router, and Receivers](image)

Fig. 1. Sources/receivers connection in a fleet of UAVs

The bottleneck link is shared by $N$ flows and TCP applies the congestion avoidance algorithm described in [21] for example. To recover an Intrusion Detection System (IDS) accuracy using the network topology presented before, it is necessary to use, if possible, additional instruments (e.g. probability of packet, queue of the router buffer) and a linear/non linear estimation algorithms. The estimation algorithm makes use of the queue at the router buffer which delivers a scalar $q$. Assuming a continuous flow, the behavior of our topology network can be mathematically described as follows:

$$\mathcal{M}_s \left\{ \begin{array}{l}
\dot{W}(t) = \frac{1}{\tau(t)} - \frac{W(t)W(t-\tau(t))}{2\tau(t)} p(t-\tau(t)) \\
\dot{q}(t) = \frac{W(t)}{\tau(t)} N - C + d(t) \\
y(t) = q(t)
\end{array} \right. \quad \text{(process)} \quad (1)
$$

In the first differential equation, $W(t)$ represents the TCP window size, $\tau(t)$ the round trip time (RTT) which can be modelled by using parameters associated to the network configuration $C$, $T_p$ as $\tau = q/C + T_p$. The latter quantity $C$ represents the transmission capacity of the router, $T_p$ the propagation delay and $N$ the number of TCP sessions. The variable $p(t)$ is the marking/dropping probability of a packet and can be seen as known measured input. This quantity depends whether the explicit congestion notification to regulate the queue size of the router buffer. In the second differential equation, $q(t)$ is the queue length of the router. The malicious anomalies are modelled by an additional signal $d(t)$ mixed with the regular traffic passing through the router and filling the buffer.

The nonlinear state space representation corresponding to $\mathcal{M}_s$ can be described in a compact form such as: $\dot{x} = f(x, u, d)$ and $y = h(x, u)$ where: $x = [W^T, q^T]^T$, $u = p$ and $y = q$ are the state, input and output vectors respectively. Moreover, a linearization of $\mathcal{M}_s$ was carried out in [24] to allow the use of traditional control theory approaches. The fluid-flow model of TCP now becomes:

$$\delta\mathcal{M}_s \left\{ \begin{array}{l}
\delta\dot{W}(t) = -\frac{N}{\tau_0} (\delta W(t) + \delta W(t - \tau(t))) \\
-\frac{1}{\tau(t)} (\delta q(t) - \delta q(t - \tau(t))) \\
-\frac{2N}{\tau_0} \delta p(t) - \tau(t)) \\
\delta\dot{q}(t) = \frac{N}{\tau_0} \delta W(t) - \frac{1}{\tau_0} \delta q(t) + d(t)
\end{array} \right. \quad (2)
$$

where $\delta W = W - W_0$, $\delta q = q - q_0$, $\delta p = p - p_0$ are the perturbed variables around the operating point defined by:

$$\begin{array}{l}
d(t) = 0 \\
W(t) = 0 \Rightarrow W_0^2 p_0 = 2 \\
q(t) = 0 \Rightarrow \begin{array}{l}W_0 = \frac{q_0}{N} \\
\tau_0 = \frac{q_0}{2} + T_p
\end{array}
\end{array} \quad (3)
$$

Inspired by the theory of time delay systems [10] the dynamics of the queue and the congestion window are modelled to address delay issue. Indeed time delay is an intrinsic phenomenon in networks whose control should improve the precision of $\delta\mathcal{M}_s$. The idea is to exploit the linearized TCP fluid model within a time delay framework as follows where $\delta\dot{x}(t) = [\delta W(t) \quad \delta q(t)]^T$ is the state vector and $\delta u(t) = \delta p(t)$ the input:

$$\delta\mathcal{M}_s \left\{ \begin{array}{l}
\delta\dot{x}(t) = A\delta x(t) + A_d\delta x(t - \tau(t)) \\
+ B\delta u(t - \tau(t)) + B_d d(t) \\
y(t) = \begin{bmatrix} 0 & 1 \end{bmatrix} \delta\dot{x}(t)
\end{array} \right. \quad (4)
$$

with

$$A = \begin{bmatrix}
\frac{N}{\tau_0} & \frac{1}{\tau_0} \\
\frac{-1}{\tau_0} & \frac{-1}{\tau_0}
\end{bmatrix} \quad (5)
$$

$$A_d = \begin{bmatrix}
\frac{N}{\tau_0} & \frac{1}{\tau_0} \\
\frac{-1}{\tau_0} & \frac{-1}{\tau_0}
\end{bmatrix} \quad (5)$$
and

\[
\begin{bmatrix}
B \\
B_d
\end{bmatrix} = \begin{bmatrix}
\frac{-\alpha^2 p_0}{2N_0^2} \\
0 \\
0
\end{bmatrix}
\] (6)

Based on such a linearized model, formulated in the general form of a time delay system, it is possible to design both AQM and estimator by using the Lyapunov Krasovskii method [22] which is an extension of the traditional Lyapunov theory. It is an effective and practical method which provides LMI (Linear Matrix Inequalities [23]) criteria easy to test.

III. CONTROLLER/OBSERVER DESIGN FOR AN IDS

Recent progresses in congestion control systems such as AQM have been led to achieve high efficiency and reliability of communications in computer networks. Under conditions where a malicious intrusion is not present, various mechanisms exist in the literature such as Random Early Detection [25] (RED), Random Early Marking [26] (REM) and more recently using control theory (proportional and proportional integral controller [27] or state feedback controller [28]). As shown in Figure 2 the control law stabilizes the TCP network (queue lengths and rates) to a desired equilibrium \((W_0, \tau_0, \eta_0)\) in spite of the presence of some non-responsive traffics, ensuring then a certain level of quality of service (QoS).

![Fig. 2. Introduction of an additional TCP traffic as anomaly](image)

A major issue in the certificating of a fleet of UAVs is to estimate the malicious intrusion while taking into account a level of QoS (i.e., the drop probability \(p(t)\)). In that way, the non modelled malicious traffic \(d(t)\) has to be estimated. The estimator has to be designed in addition to an efficient AQM. Thus, we propose to design a robust controller/observer for IDS by solving an LMI criteria on the following augmented model:

\[
\begin{bmatrix}
\hat{\delta x}(t) \\
\hat{\delta y}(t) \\
\hat{\delta z}(t)
\end{bmatrix} = \begin{bmatrix}
\hat{\delta x}(t) + \hat{\delta y}(t) - \delta z(t) \\
\hat{\delta y}(t) - \delta p(t) \\
\hat{\delta z}(t)
\end{bmatrix}
\] (7)

With

\[
\begin{bmatrix}
\bar{A}_{\delta} \\
\bar{A} \\
\bar{B}
\end{bmatrix} = \begin{bmatrix}
A & 0 & 0 \\
0 & 0 & 0 \\
B & 0 & 0
\end{bmatrix} \\
\bar{C} = \begin{bmatrix}
0 & 1 & 0
\end{bmatrix}
\] (8)

where \(\delta \bar{z}(t)^T = [\delta W(t) \delta q(t) \delta d(t)]^T\) represents the augmented state. Practically, the objective is to reconstruct the whole state of Eq. (7), in particular the non-TCP malicious flows modelled by \(d(t)\), and design an output feedback AQM. Consequently, we are looking for gain controller \(K\) and gain observer \(L\) defined as:

\[
\begin{bmatrix}
\delta u(t - \tau(t)) \\
\delta \dot{\bar{z}}(t)
\end{bmatrix} = \begin{bmatrix}
-K_p(t) \delta q(t) \\
\bar{A}\delta \bar{z}(t) + \bar{A}_d\delta \bar{z}(t) - \tau(t)
\end{bmatrix}
\] (9)

The first equation corresponds to the dynamics of the AQM. We assume that the time-delayed system described by Eq. (4) is controlled by a static output feedback AQM such as the dynamics of the closed loop system is the following:

\[
\delta \bar{z}(t) = A_{cl}\delta \bar{z}(t) + A_d\delta \bar{z}(t) - \tau(t)
\] (10)

where:

\[
A_{cl} = A - BKC
\] (11)

The second equation corresponds to the estimation of the state vector \(\delta \bar{z}\) and describe the dynamics of the observer. We recognize the typical mathematical expression of a linear state estimator with correction terms \(L\). The idea is to build an additive correction term based on linear gains \(L\) which keeps stable the dynamics of the estimation error \(e(t)\).

Combining all these results, a direct and analytical design of the observer can be proposed thanks to the error between the state of the system \(\delta \bar{z}(t)\) and its estimated value \(\delta \hat{z}(t)\):

\[
e(t) = \delta \bar{z}(t) - \delta \hat{z}(t)
\] (12)

Thus, the dynamics of the estimation error is the following:

\[
\dot{e}(t) = A_{cl}e(t) + A_d\delta \bar{z}(t) - \tau(t)
\] (13)

where:

\[
A_{cl} = A - LC
\] (14)

The dynamics of the estimation error \(e(t)\) given by Eq. (13) in the observer problem is similar to the dynamics of the closed loop state vector \(\delta \bar{z}(t)\) given by Eq. (10) in the controller problem assuming \(B = I\) and \(K = L\). This enables to use the same methodology for the observer and the controller design to determine all correction terms associated with the time delay linear state-space representation of TCP model. By using the theorem in annexes which introduces an appropriate Lyapunov Krasovskii functional, the following methodology gives a simple way to construct unknown matrices \(K\) and \(L\) in order to stabilize Eq. (13):
1) Use a static output feedback [30] for the AQM to
compute the static output gain matrix $K$ corresponding
to a set of predefined eigenvalues of the closed loop
state matrix $A_{cl}$.

2) Check the robustness of the proposed design by solving
the following LMI:

$$
\begin{align}
& P = P^T > 0 \\
& Q = Q^T > 0 \\
& R = R^T > 0 \\
& \begin{bmatrix} 
\Gamma_{11} & \Gamma_{12} \\
\Gamma_{21} & \Gamma_{22} 
\end{bmatrix} < 0
\end{align}
$$

where:

$$
\begin{aligned}
\tau &= \sup_t (\tau(t)) \\
\Gamma_{11} &= A_{cl}^T P + PA_{cl} + Q - \frac{1}{\phi} R + \tau A_{cl}^T R A_{cl} \\
\Gamma_{12} &= P A_d + \tau A_{cl}^T R A_d + \frac{1}{\phi} R \\
\Gamma_{22} &= -Q + \tau A_{cl}^T R A_d - \frac{1}{\phi} R
\end{aligned}
$$

3) The solving of LMI in Eq. (15) has to be done also
for $L$ in order to design the observer.

Such an approach is systematic for more complex dynamical
systems than the ones represented by a single router.

IV. CONTROLLER/OBSERVER VALIDATION IN REAL
TRAFFIC CONDITIONS

A. Hybrid UAV network simulator

In order to validate the new traffic estimator in real traffic
conditions, we use a hybrid experimental system to combine
the low cost of a simulation with the accuracy of a real
protocol stack. We use virtual machine implementations to
deal with the entire complexity of Linux operating system.
The traces used to generate UAVs mobility patterns were
extracted from real traces so that physical related factors
could be as realistic as possible. The system we used to
evaluate protocols is divided in several parts. It includes
a set of tools that can fit to several scenarios: an hypervisor
to run the virtual machines, measurement tools and a framework
to allow virtual machines to communicate through a virtual
wireless medium. We chose to use VirtualBox as a virtualisation
tool because it is an easy-to-use and efficient hypervisor.
The virtualized system is a 12.04 version Ubuntu, working
with the 2.6.38 version of the Linux kernel. Our testbed
architecture uses a Virtualmesh framework. It is a framework
that interfaces a Linux-based system with an OMNeT++
simulation. OMNeT++ is a powerful network simulator
which simulates several systems and normalized protocols.
An illustration of this system is depicted in Figure 3. In [29]
more details about this hybrid tool can be found.

The main advantage using such an hybrid simulator is to
extract any characteristics from the simulation and to inject
them in the Simulink design directly. The theoretical model is
then confronted to real traffic characteristics and not only the-
oretical stimulus that are pertinent for a first stage evaluation
but that do not take into account the huge variability of real
traffic. Consequently, we have been able to generate DDoS
between the different virtual machines considering the exact
UAV environment of the drone mission we have considered
in section II. Then, we captured the network traffic generated
(both regular traffic and the DDoS traffic) and finally, we
have injected this traffic in the Simulink design.

B. Experimental results

We now illustrate the performances reached by the de-
volved controller/observer on the basis of hybrid UAV
network simulator. As it was aforementioned, the normal
TCP traffic is generated by 5 TCP sources generating long
lived TCP flows to a receiver through a router with a link
capacity $C=1250$ packets/s (which is equivalent to $3$ Mbit/s),
and $T_p = 30\text{ms}$ the propagation delay. This is illustrated in
Figure 4.

Moreover, we define in the Table below the values of
the congestion window size and the queue length at the
equilibrium of the system, i.e. $W_0$ and $q_0$ from the mean
value for $N$ sessions around which $W(t)$ and $q(t)$ oscillate
respectively. The proposed observer has been tested with
the state feedback AQM in [10] and observer gains are
$L = [0.024, 1.598, 20.978]^T$. First of all, we present the
TCP congestion window $W(t)$ in Figure 5 for which the
differences between linear and nonlinear models may be
wrong. Indeed, we can observe a slightly influence on the
linearized model when $W(t)$ goes down and up strongly (i.e.
far from its equilibrium point).
In Figures 5, 6 and 7 we illustrate a typical realization of traffic including four different traffic anomalies which can be detected by our time-delay linear observer. The four different intrusions generated by the hybrid simulation tool have been injected in Simulink to confront our IDS model to the real traffic traces. These anomalies have been generated using the HPing3 tool. This software is run on the hacker node (see Figure 4 for details about the network topology simulated) and can run different types of attack (but mainly flooding attacks). In our scenario, HPing3 exchanges thousand of small TCP flows in order to generate a SYN flood attack on the receiver node. The resulting malicious traffic is quite more important than the regular traffic. This is depicted in the Figure 7 where regular traffic is around 50 pkt/s when, for the malicious traffic, the throughput is increased to 150 pkt/s. Then, the real traffic (blue) and estimated intrusion (orange) are plotted on the same figure in order to be compared. Figure 6 shows the time response of the estimation queue $q(t)$ calculated by the time-delay linear observer method. As expected, the queue is stabilized above the desired level and the intrusion do not affect the different steady states of the system.

These results are positive given that the estimator simulated with Matlab Simulink is able to detect the different intrusions rapidly and with an accurate threshold. The delay in the detection is negligible and the estimator can make an accurate difference between legitimate traffic and intrusion traffic. This is a first promising result for intrusion detection system design applied to drone fleet network.

V. CONCLUSION AND FUTURE WORK

With this paper, we demonstrate that a linear controller/observer can improve intrusion detection systems in the specific context of drone fleet. Our design methodology provides a simple way to construct and instantiate our gain matrices for both the AQM controller and the observer. This approach gives us first promising results with a simple topology (within a time-delay framework). We plan to apply this approach also in the context of more dynamical systems (such as fleet of drones) where several nodes are involved in the network topology in order to build an ad hoc network.

In our future work we can identify several contributions. First of all, we plan to propose an evolution of the modelling of the drone fleet network. This should include different types of traffic (UDP and TCP for instance) and also take into account network mobility. Moreover, we plan to analyse different types of attack: not only DDoS but also intrusion where the traffic generated in the network is less consistent and then more difficult to detect. A first solution would be to consider a bank of models in order to detect with different signatures DDoS and other types of attack. Finally, we plan to investigate a way to implement and test this new generation of traffic estimator in real system and real environment. To address this last objective, we would like to consider real experiments with real UAV. Each UAV could integrate the bank of models previously introduced and by
conducting a collaborative mission in the context of a UAV fleet we could test and validate the theoretical estimators validated only in simulations.

VI. ANNEXES

Theorem 1: For given \( r = \sup_{t}(\tau(t)) \) and \( \alpha > 0 \), if there exists two definite positive matrices, \( W = PL \) and \( R = \alpha P \) of appropriate dimension such that the following LMI holds

\[
\begin{align*}
  &P = P^T > 0 \\
  &Q = Q^T > 0 \\
  &\begin{bmatrix} 
    \Phi_{11} & \Gamma_{12} & \Phi_{12} \\
    \Gamma_{12}^T & \Gamma_{22} & 0 \\
    \Phi_{12} & 0 & -P 
  \end{bmatrix} < 0 \\ 
\end{align*}
\]

(17)

with:

\[
\begin{align*}
  \Phi_{11} &= A^T P - C^T W^T + PA - WC + Q - \frac{\tau}{2} P \\
  \Gamma_{12} &= PA_d + \tau \alpha (PA - WC)^T A_d + \frac{\tau}{2} P \\
  \Phi_{12} &= \sqrt{\tau \alpha} (PA - WC)^T \\
  \Gamma_{22} &= Q + \tau \alpha A_d^T PA_d - \frac{\tau}{2} P \\
\end{align*}
\]

(18)

then the system defined by Eq. (13) is asymptotically stable and the observer gain is given by \( G = P^{-1} W \).

REFERENCES