



Event-based speed control on a sensor-less miniature thruster

Thibaut Raharijaoana, Lorris Dola, Bruno Boisseau, John Jairo Martinez Molina, Nicolas Marchand, Franck Ruffier

► To cite this version:

Thibaut Raharijaoana, Lorris Dola, Bruno Boisseau, John Jairo Martinez Molina, Nicolas Marchand, et al.. Event-based speed control on a sensor-less miniature thruster. AIM 2014 - IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Jul 2014, Besançon, France. pp.n/c. hal-00982430

HAL Id: hal-00982430

<https://hal.science/hal-00982430>

Submitted on 24 May 2014

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.

Event-based speed control on a sensor-less miniature thruster

Thibaut Raharijaona¹, Lorris Dola¹, Bruno Boisseau², John-Jairo Martinez-Molina², Nicolas Marchand² and Franck Ruffier¹

Abstract—The main contribution of this paper is the experimental evaluation of the performance of different event-based control strategies in the case of a sensor-less miniature thruster.

While most control engineering applications considers periodic control with equidistant sample intervals, the event-based approach updates the control signal only when required. The performances in terms of thruster power consumption and control update number are evaluated by comparing the event-based control system with respect to the time-based loop for the angular speed regulation of a sensor-less miniature thruster. The experimental results clearly show an important reduction of the control updates while the power consumptions are similar.

I. INTRODUCTION

Due to the necessity of saving the energy and increasing the autonomy of embedded systems, event-based control becomes an interesting research topic. While the classical so-called time-based framework of controlled systems consists in sampling the system uniformly in time with a constant sampling period and in computing and updating the control law every time instants, the event-based approach aims at updating asynchronously the control signal. The control signal is updated only when required. Some works addressed more recently event-based sampling where the control law is event-driven [1], [2], [3], [4].

For an application point of view, there are several possible ways to save the energy and improve the autonomy. The first way aims at reducing the computation load and the second way tends to reduce the actuation load. Low-power electronic components in all embedded and miniaturized mechatronic applications encourages companies to develop asynchronous versions of the existing time-triggered components. A significant power consumption reduction can be achieved by decreasing the samplings and consequently the CPU utilization. In [5], an event-based proportional integral derivative (PID) controller is presented with the simulation results applied to a double tank process. For stability reason, a maximal sampling period is introduced in the initial approach. In [6], this maximal period is removed and different event-based PID algorithms without safety limit condition were also developed. In [7], the efficiency of an approach without safety limit condition is highlighted by implementing such a controller in a real-time testbed. A radio-controlled vehicle embeds an event-based cruise control mechanism. In

addition, the reduction of computational load is investigated in [8] and an event-driven system theory is drawn for disturbed linear systems.

In this paper, using miniature actuators which equipped the bio-inspired robot LORA III [9], an experimental set-up was developed to test some event-based techniques for speed tracking and disturbance rejection. This work aims at comparing the performances of time-based and event-based control laws in terms of power consumption and control updates in particular an event-based state feedback approach that uses the instantaneous current information to limit the starting peak current and the power consumption. The paper is structured as follows. Firstly, the experimental platform is presented in section II. The event-based angular speed control laws of the miniature thruster are designed in section III. Experimental results are provided in section IV.

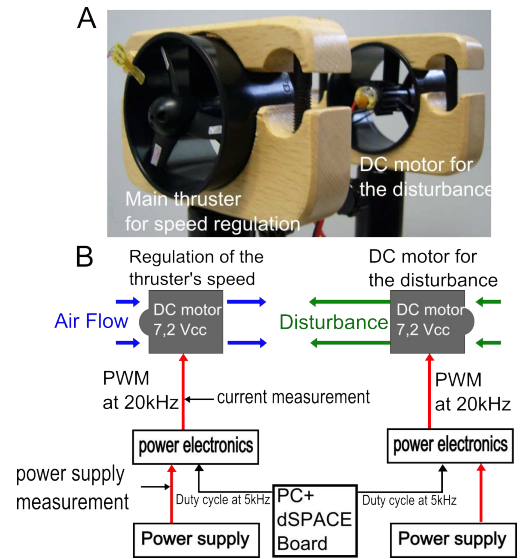


Fig. 1. A: Picture of the experimental platform which is equipped with a miniature thruster used to compare the event-based algorithms and a additional thruster which generates wind disturbance B: Sketch of the sensor-less miniature thruster testbed using a dSPACE electronic board

II. THE MINIATURE SENSOR-LESS THRUSTER TESTBED

The experimental platform is depicted in Fig. 1A. It is based on a miniature direct current thruster GWS EDF-50 (see Fig. 1A, power: 24.41W, mass: 29g) which was previously embedded on the bio-inspired robot LORA III [9]. The maximum allowable current drawn is 3.4A under 7.2V. Some extra components were added to make possible the sensor-less thruster speed regulation.

¹Thibaut Raharijaona, Lorris Dola and Franck Ruffier are with Aix-Marseille Université, CNRS, ISM UMR 7287, 13288, Marseille cedex 09, France thibaut.raharijaona@univ-amu.fr

²Bruno Boisseau, John-Jairo Martinez-Molina and Nicolas Marchand are with GIPSA-lab, Control Systems Department, UMR CNRS 5216-INPG-UJF, ENSIEG BP 46, 38402 Saint Martin d'Heres Cedex, France

A. The testbed

The aim of the experimental set-up is to implement different kinds of time-based and event-based control laws from the literature for thruster's speed regulation and disturbance rejection without any angular speed measurement. The performances in term of power consumption and control updates are considered. As presented in the right side of Fig. 1B, a second thruster generates wind disturbances.

B. The data acquisition and control

The data acquisition and control are performed in a real-time framework using a dSPACE electronic board which is connected to the experimental platform. The control laws are implemented using Matlab/Simulink. The measurements and performances are monitored online and one can analyze them off line.

C. Estimation of the rotor speed

In this section, we propose to estimate the rotor speed. It simplifies mechanical assemblies and reduces the size of electronic circuits. Moreover, one can significantly reduce the sampling frequency $F_s = 20kHz$ required if a reflective sensor would measure the thruster's angular speed. Therefore estimating the thruster's angular speed allows to save energy and computational resources. The speed estimation simply requires that the armature voltage and the motor current be measured. The estimation principle is presented in [10]. One can recall the main steps.

The dynamics of a DC motor is described by the following equations:

$$\frac{di(t)}{dt} = -\frac{R}{L} \cdot i(t) - \frac{K_e \cdot M}{L} \cdot \omega_h + \frac{1}{L} \cdot u(t) \quad (1)$$

$$\frac{d\omega_h(t)}{dt} = \frac{K_t}{J \cdot M} \cdot i(t) - \frac{D}{J \cdot M^2} \cdot \omega_h(t) - \frac{1}{J \cdot M} \cdot c_0(t) \quad (2)$$

- $\omega_h(t)$: the thruster angular speed (rad/s)
- $c_0(t)$: an additive bounded disturbance
- $u(t)$: the armature voltage (V)
- $i(t)$: the armature current (A)

The parameters are defined as:

- D : the drag coefficient of the thruster (N.m.s/rad)
- J : the inertia of the motor (kg.m²)
- K_t : the torque constant (N.m/A)
- K_e : the back-EMF voltage constant (V.s/rad)
- L : the inductance (H)
- M : the reduction ratio between the thruster and the rotor
- R : the armature resistance (Ω)

In the following, we consider that the mechanical time constant is greater than the electrical time constant. Therefore, we assume that the armature current $i(t)$ is constant during the thruster's speed $\omega_h(t)$ transient phase. As shown in Fig. 2A, the measurements of the armature voltage $u(t) = v_{bat} - v_1$ and the armature current $i(t) = \frac{v_1 - v_2}{R_s}$ measured thanks to a current shunt resistor R_s , gives an estimated thruster's speed $\omega_{est}(t) = \hat{\omega}_h(t)$ in Volt such that : $\omega_{est}(t) =$

$u(t) - R_s i(t)$. In Fig. 2B, the measured current $i(t)$ is noisy and it is not possible to use it for estimation. The current measurement is filtered using a Kalman filter whose the process noise covariance is set 0.0034 and the measurement noise covariance is set to 20 after experiments. The measured current and the estimated current using the Kalman filter are shown in Fig. 2B.

In the Laplace domain, one writes: $\Omega_{est}(s) = U(s) - R_s I(s)$ and from (1) in steady state, $U(s) = K_e M \Omega_h(s) + R I(s)$. From (2), assuming D is small and $c_0(t) = 0$, in the Laplace domain $I(s) = \frac{sJM}{K_t} \Omega_h(s)$. Finally, the estimated velocity is written such as:

$$\Omega_{est}(s) = K_e M \Omega_h(s) + R I(s) - R_s I(s)$$

$$\Omega_{est}(s) = K_e M \left(1 + \frac{sJ(R - R_s)}{K_e K_t} \right) \Omega_h(s) \quad (3)$$

We define $\tau = \frac{J(R - R_s)}{K_e K_t}$. In practice, one can remark that $\tau \ll 1$, therefore $\Omega_{est}(s) \simeq K_e M \Omega_h(s)$.

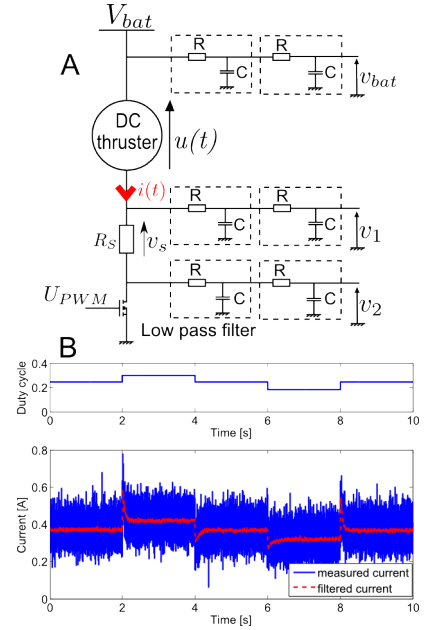


Fig. 2. A: Electric diagram of the DC thruster. The current $i(t)$ is obtained with the measured voltages v_1 and v_2 . The voltages are filtered with a low-pass filter. The voltage U_{PWM} controls the thruster speed through the transistor B: The current $i(t)$ is filtered with a Kalman filter. The top plot is the duty-cycle applied in open-loop and the bottom plot shows the measured current in blue line and the filtered current in red dashed line.

In Fig. 3, the estimated speed in dashed red line is compared to the measured speed in solid blue line using a reflective sensor at the sampling frequency $F_s = 20kHz$. One can remark that the speed estimation is fair. The gain $K_e M$ from (3) is experimentally tuned and fixed for the following work. Finally, we estimate the thruster's speed without any velocity sensor using only the following measurements $u(t)$, $v_1(t)$ and $v_2(t)$. The estimated speed will feed the controllers for regulation and disturbance rejection at a lower sampling frequency $F_s = 5kHz$.

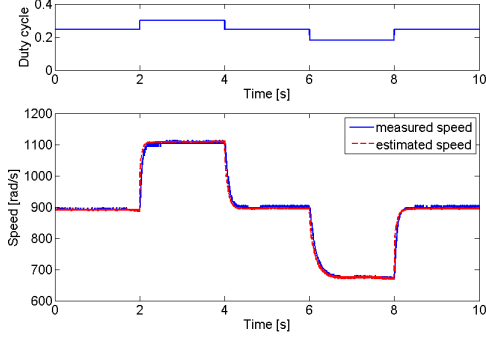


Fig. 3. Comparison between the measured speed and the estimated speed: the top plot is the duty-cycle applied in open-loop and the bottom plot shows the measured speed in blue line and the estimated speed in red dashed line.

D. Discrete-time model identification

Under the assumption of homogenous magnetic field, the miniature direct current thruster is supposed to be a linear plant. Equations (1) and (2) are considered. The variable to be controlled is the angular speed $\omega_h(t)$ which is estimated as detailed in sec. II-C. In addition, state-space oriented control is investigated in order to limit the starting peak current. Therefore, we define as state variables the armature current $i(t)$ and the thruster speed $\omega_h(t)$ such as:

$$\begin{cases} x(k+1) = A.x(k) + B.u(k) \\ y(k) = C.x(k) \end{cases} \quad (4)$$

where $x(k) = [i(k), \omega_h(k)]^T$, $u(k)$ is the control input signal, $y(k)$ is the measurement, A is a 2×2 matrix, B is a 2×1 matrix and $C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

For the discrete time state-space model identification, data are recorded at the sampling frequency $F_s = 5kHz$. The measured current varies from 0.2A to 0.5A and the estimated thruster's speed varies from approximately less than 900 rad/s to 1100 rad/s. The identified state-space model for the miniature thruster is given by the representation:

$$\begin{cases} x(k+1) = \begin{bmatrix} 0.9873 & 0 \\ -4.262 \times 10^{-5} & 0.9541 \end{bmatrix} .x(k) \\ \quad + \begin{bmatrix} 49.18 \\ 0.2022 \end{bmatrix} .u(k) \\ y(k) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} .x(k) \end{cases} \quad (5)$$

For the state space model identification, the best fits for the speed and the current are respectively equal to 87.7% and 62.2%.

The model validation is performed using a new set of data. Figures 4A and 4B exhibit the results. The fits for the speed and the current are respectively equal to 88.7% and 61.6%.

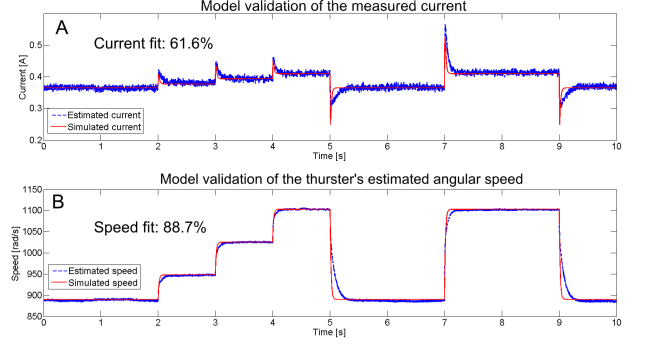


Fig. 4. Comparison between the measured outputs and the simulated model outputs with the same input. A: Comparison between the estimated current in dashed blue line and the simulated current in red solid line. B: Comparison between the estimated speed in dashed blue line and the simulated speed in red solid line.

III. EVENT-BASED SPEED CONTROL

The thruster's speed is usually controlled with a time-based proportional-integral (PI) strategy or time-based state feedback. In this section, we propose to design some event-based approaches introduced in [6] and [8] for both regulation and disturbance rejection and to compare the results to the ones obtained with the time-based controllers.

A. Time-based PI control

The proportional-integral controller in frequency domain is:

$$U(s) = K_p \left[E(s) + \frac{1}{T_i s} E(s) \right] \quad (6)$$

where $U(\cdot)$ is the control signal and $E(\cdot)$ the error between the estimated output of the controller system and a given setpoint. K_p and T_i are some tunable parameters. A discrete time controller is finally obtained, that is:

$$\begin{cases} u_P = K_p.(\omega_{ref}(k) - \omega_h(k)) \\ u_I(k) = u_I(k-1) + \frac{K_p}{T_i}.T_s.(\omega_{ref}(k) - \omega_h(k)) \\ u(k) = u_P(k) + u_I(k) \end{cases} \quad (7)$$

where $T_s = 2.10^{-4}s$ (ie: $F_s = 5kHz$).

In order to limit the amplitude of the control input and the peaks of current, we choose the same settling time at 95% for the closed-loop and the open-loop systems i.e. $\tau_{settling} = 0.173s$. Then $T_i = 0.0442s$ and $K_p = 2.9377.10^{-4}$ are the parameters of the PI controller which gives a gain margin of 64.9dB and a phase margin of 90°.

B. Time-based state feedback control

A discrete-time state feedback controller with integral action for trajectory tracking is also designed such that: $u(k) = -[K_i \ K_{\omega_h} \ K_z] \cdot [i(k) \ \omega_h(k) \ z(k)]^T$. Using a backward discretization, the discrete-time control signal is computed as follows:

$$\begin{cases} u_P(k) = -[K_i \ K_{\omega_h}] \cdot [i(k) \ \omega_h(k)]^T \\ u_I(k) = u_I(k-1) - K_z.T_s.(\omega_{ref}(k) - \omega_h(k)) \\ u(k) = u_P(k) + u_I(k) \end{cases} \quad (8)$$

Where $K_i = -0.2270$, $K_{\omega_h} = -0.0001$ and $K_z = -0.0015$. The settling time at 95% is the same as for the PI controller.

C. Event-based state feedback control

The event-based state feedback controller is designed using [8]. A geometric interpretation is proposed in Fig. 5. The control update $u(k)$ is computed at a constant sampling rate T_s when the state of the system lies outside a pre-defined set β . When the state is inside the set β , the control keeps its previous value. If we consider the system $x(k+1) = A.x(k) + B.u(k)$, we design the set β where the control is held by $\beta := \{x \in \mathbb{R}^n \mid |x| < e_T\}$. The threshold e_T ensures a certain level of performance. The discrete-time state feedback control signal is computed as follows:

$$u(k) = \begin{cases} -Kx(k) & \text{if } |x(k)| \geq e_T \\ u(k-1) & \text{if } |x(k)| < e_T \end{cases} \quad (9)$$

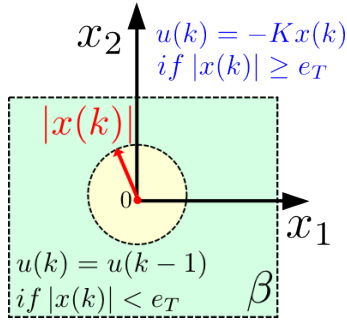


Fig. 5. Geometric interpretation of the event-based state feedback control proposed in [8]. As long as the state $x(k)$ is inside the set β , the control input signal is $u(k) = u(k-1)$.

Let introduce the instant τ_k . τ_k denotes the event times by using a zero-order hold such that: $u(t) = u(k)$ for all $t \in [\tau_k, \tau_{k+1})$. Therefore, $x(k) = x(\tau_k)$ and $u(k) = u(\tau_k)$. In this paper, the controller checks whether the state is inside or outside the set β at a constant rate. This rate is chosen to be equal to the sampling rate of the controller T_s such as $\tau_{k+1} = \tau_k + T_s$. Then as long as the state is outside the set β , the control is updated at a constant sampling time. When the state reaches the set β the control is held to the value just before entering the set β . The gains K_i , K_{ω_h} and K_z are the same as for the time-based state feedback controller.

D. Event-based PI control without safety limit condition

The event-based control described in [6] is designed. The event occurs when $|e(k) - e(k-1)| > e_{lim}$ where e is the error $e(k) = \omega_{ref}(k) - \omega_h(k)$ and e_{lim} is a pre-defined threshold. As proposed in [6], we implement an algorithm without the safety limit condition firstly introduced in [5] in order to prevent the output from high overshoot. The hybrid algorithm is a mix between the approach with an exponential forgetting factor and the approach with a saturation of the product $h_{act} \cdot e$. The resulting algorithm is:

Algorithm 1 Hybrid algorithm from [6]

```
% Inputs
ysp = u(1); % SetPoint
y = u(2); % Measurement
e = ysp - y;

% Calculate control signal
hact = hact + Ts;
if ( abs(e-e_old)>elim )

    up = Kp*e;
    if ( hact >= hmax )
        hact_i = hact*exp(hnom-hact);
        he = (hact_i - hnom)*elim + hnom*e;
    else
        he = hact*e;
    end
    ui = ui + Kp/Ti*he;
    u = ui + up;
    % Update
    e_old = e;
    y_old = y;
    hact = 0;
end
```

The parameters h_{max} and e_{lim} are chosen such as the control update are significantly reduced while the control allows a reference tracking. Therefore, $h_{max} = 4$ and $e_{lim} = 0.1$. Moreover, the gain K_p and T_i are the same as the ones used for the time-based PI controller.

IV. EXPERIMENTAL RESULTS

In this section, we propose to test the different event-based control strategies and to compare their performances to the time-based approaches on the miniature thruster for trajectory tracking and disturbance rejection.

A. Trajectory tracking

In Fig. 6A, the thruster's speed with the event-based PI controller from Sec. III-D is plotted and is compared to the time-based PI controller. The event-based approach performs a trajectory tracking without static error. One can remark increases of the overshoots during the changes of reference with the event-based approach. In Fig. 6A, the trajectory tracking is performed with only 1778 updates which is a reduction of 96.5% of the control updates (see Fig. 6B). The power consumption is smaller with the event-based controller (0.52W) than with the time-based one (0.54W). Figures 6C and 6D show the comparison between the duty cycles of the PWM in % and the currents.

In Fig. 7A, the thruster's speed with the event-based state feedback controller from Sec. III-C is plotted and is compared to the time-based state feedback controller.

We define the performance set β such that:

$$\beta := \begin{cases} \omega_{ref} - \omega_{tolerance} < \omega_h < \omega_{ref} + \omega_{tolerance} \\ i_{set} - i_{tolerance} < i < i_{set} + i_{tolerance} \end{cases} \quad (10)$$

with $\omega_{tolerance} = 40 \text{ rad/s}$, $i_{set} = 0.3 \text{ A}$ and $i_{tolerance} = 0.1 \text{ A}$. Figure 7A shows that the event-based state feedback performs the reference tracking with a static error lower than $\omega_{tolerance} = 40 \text{ rad/s}$.

The reduction of the control updates is about 91% with the event-based state feedback (see Fig. 7B).

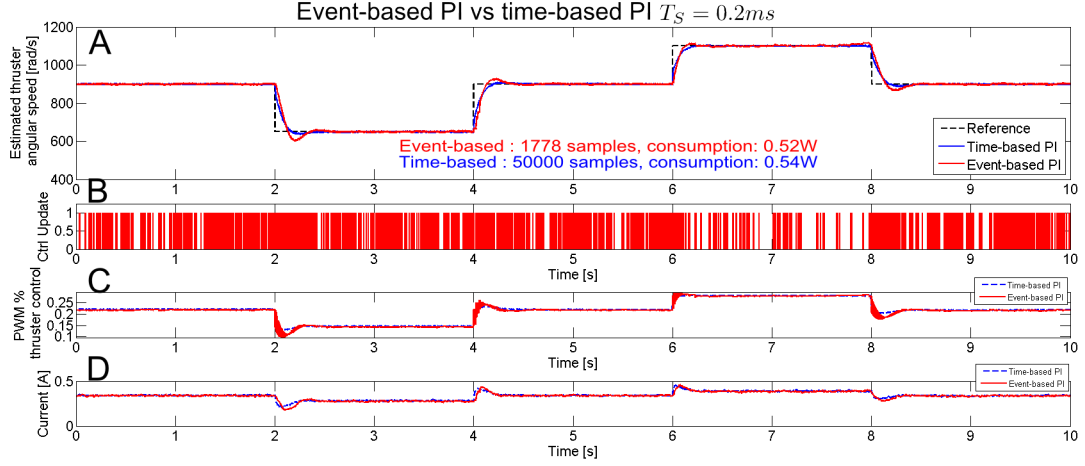


Fig. 6: A: Trajectory tracking: comparison between the time-based PI in dashed blue line and the event-based PI from [6] in red line. The figure shows the evolution of the thruster speed with both types of control. B: The event-based control update is also displayed. The update is performed when the indicator is at 1. C: The duty cycles of the PWM in % with the time-based PI in dashed blue line and the event-based PI in red line. D: The current with the time-based PI in dashed blue line and the event-based PI in red line.

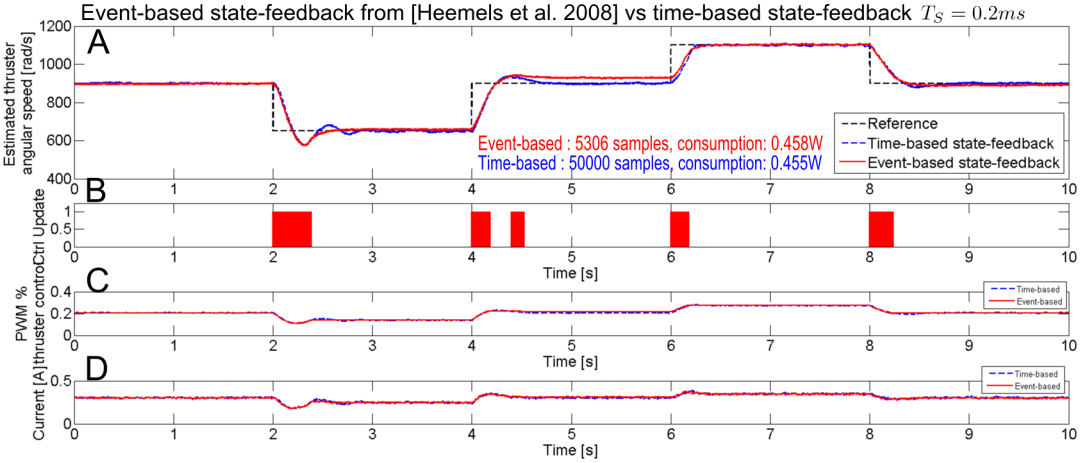


Fig. 7: A: Trajectory tracking: comparison between the time-based state feedback in dashed blue line and the event-based state feedback from [8] in red line. B: The event-based control update is also displayed. The update is performed when the indicator is at 1. C: The duty cycles with the time-based state feedback in dashed blue line and the event-based state feedback in red solid line. D: The current with the time-based state feedback in dashed blue line and the event-based state feedback in red solid line.

The power consumptions are slightly identical (0.45W). In Fig. 7C, the duty cycle of the PWM is only updated during the changes of reference and keeps its previous value during the steady-state. Figure 7D displays the currents. The trajectories of the state ω_h versus i with the time-based and event-based controllers are plotted in Fig. 8. One can see that the trajectory of the states converges to the different performance sets $Set(A)$, $Set(B)$ and $Set(C)$.

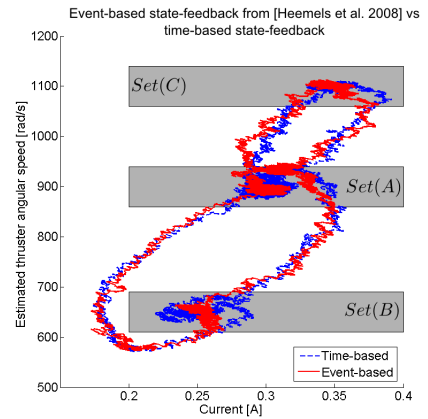


Fig. 8: The figure shows the phase portrait in closed loop: the current $i(t)$ and the thruster speed $\omega_h(t)$. The trajectories of the time-based controller and the event-based one are respectively plotted in dashed blue line and in red solid line. The performance sets are represented by the grey rectangles.

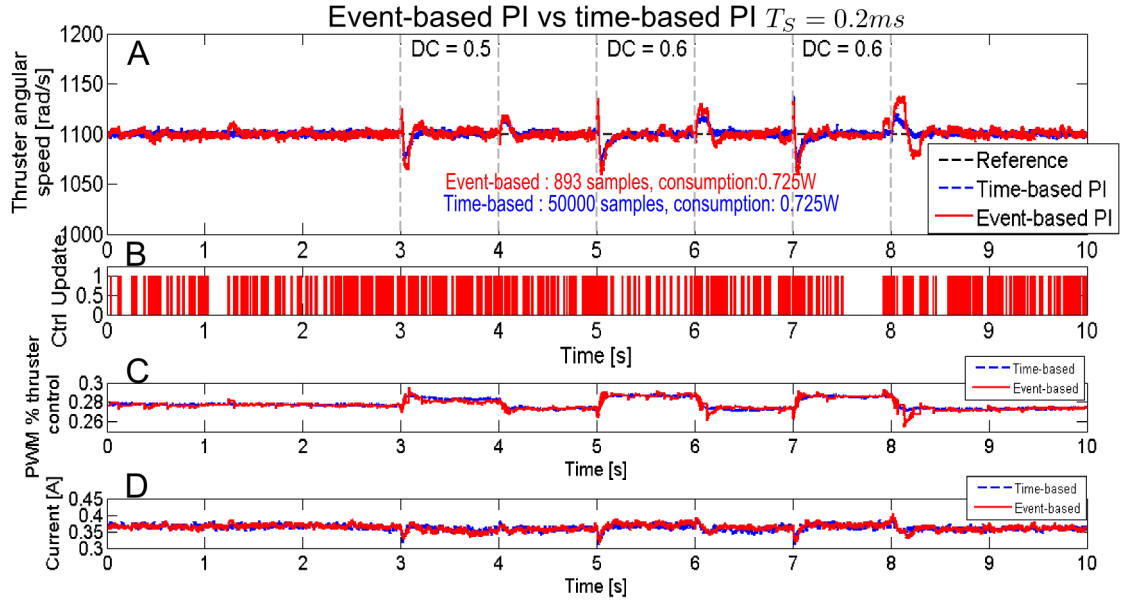


Fig. 9. A: Disturbance rejection: comparison between the time-based PI in dashed blue line and the event-based PI from [6] in red line. B: The event-based control update is also displayed. The update is performed when the indicator is at 1. C: The duty cycles of the PWM in % with the time-based PI in dashed blue line and the event-based PI in red solid line. D: The current with the time-based PI in dashed blue line and the event-based PI in red solid line.

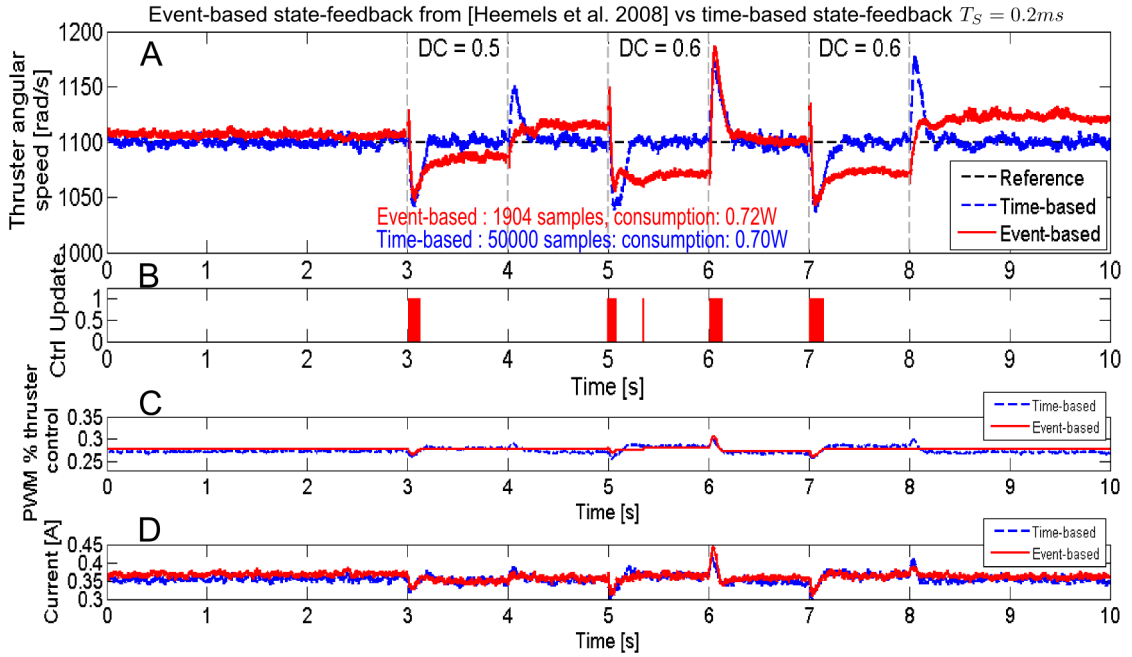


Fig. 10. A: Disturbance rejection: comparison between the time-based state feedback in dashed blue line and the event-based state-feedback from [8] in red line. B: The event-based control update is also displayed. The update is performed when the indicator is at 1. C: The duty cycles with the time-based state feedback in dashed blue line and the event-based state feedback in red solid line. D: The current with the time-based state feedback in dashed blue line and the event-based state feedback in red solid line.

B. Disturbance rejection

In this section the disturbances are gusts of wind generated by a second thruster. The latter is controlled in open-loop with different duty cycles and activated at different moments. The disturbances occurs at each interval $t \in [3s; 4s]$, $t \in [5s; 6s]$ and $t \in [7s; 8s]$ with a duty cycle respectively equals to 0.5, 0.6 and 0.6 otherwise the motor is turned off with a duty cycle equal to 0. The speed of the main thruster is regulated at 1100 rad/s. Figure 9A shows that the event-based PI controller rejects the disturbances despite larger overshoots than the ones obtained with the time-based PI controller. The control updates are displayed in Fig. 9B, the reduction of the control updates is about 92%. The power consumptions are the same with the time-based and the event-based controllers (0.725W). The comparison of the control input signals is presented in Fig. 9C and Fig. 9D. From Fig. 10A, the event-based state feedback controller rejects the disturbances. The control updates are reduced up to 96% (see Fig. 10B). The duty cycle is only updated when the disturbances are added or removed (see Fig. 10C). Figure 10D plots the currents. The power consumptions are slightly the same with the time-based controller (0.70W) and with the event-based controller (0.72W).

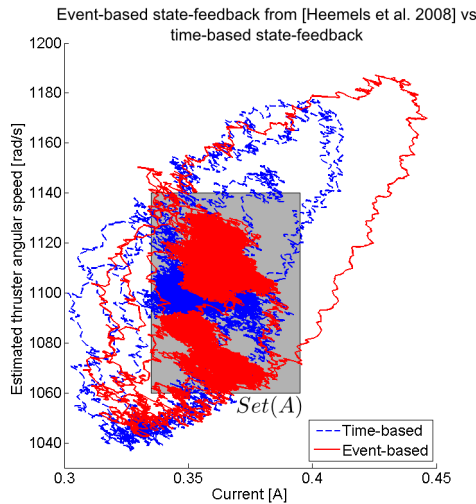


Fig. 11. The figure shows the phase portrait in closed loop: the current $i(t)$ and the thruster speed $\omega_h(t)$. The trajectories of the time-based controller and the event-based one are respectively plotted in dashed blue line and in red solid line. The performance set called $Set(A)$ is represented by the grey rectangle.

Figure 11 shows the trajectories of the states under the disturbance rejection with the time-based and event-based state feedback controllers. As soon as the states are outside $Set(A)$, the control is updated such that the states reach again the set. Once the control is held, the states converge inside the set. One can say that this strategy succeeds in limiting the power consumption and starting peak current by using the instantaneous current information.

V. CONCLUSION

An experimental platform was developed and aimed at testing some event-based control schemes using a miniature thruster. The event-based control laws performed a reference tracking and a disturbance rejection. The thruster's speed estimation was implemented thanks to the measurements of only three voltages. Therefore, the speed estimation was fed into the designed controllers. The event-based approach was performed with a PI controller and a state feedback controller. The event-based state feedback approach was based on a set defined by the controlled output and an inner state, the current, which is somehow an image of the instantaneous energy consumption. The event-based approaches allowed to significantly reduce the number of control updates while having a similar energy consumption compared to a time-based approaches while applying output disturbances.

ACKNOWLEDGMENTS

We thank M. Boyron for his assistance with the electrical design and J. Dipéri for his involvement in the mechanical design of the testbed. This work is part of the project PEPS SpikeRob, supported by The French National Center for Scientific Research (CNRS).

REFERENCES

- [1] K. Åström and B. Bernhardsson, "Comparison of riemann and lebesgue sampling for first order stochastic systems," in *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 2, Dec 2002, pp. 2011–2016.
- [2] J. H. Sandee, W. P. M. H. Heemels, and P. P. J. Van den Bosch, "Event-driven control as an opportunity in the multidisciplinary development of embedded controllers," in *Proceedings of the American Control Conference*, vol. 3, June 2005, pp. 1776–1781.
- [3] J. Lunze and D. Lehmann, "A state-feedback approach to event-based control," *Automatica*, vol. 46, no. 1, pp. 211 – 215, 2010. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0005109809004968>
- [4] A. Anta and P. Tabuada, "To sample or not to sample: Self-triggered control for nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2030–2042, Sept 2010.
- [5] K.-E. Årzén, "A simple event-based pid controller," in *Proceedings of the 14th World Congress of IFAC*, Beijing, March 1999, pp. 423–428.
- [6] S. Durand and N. Marchand, "Further results on event-based pid controller," in *Proceedings of the European Control Conference (ECC)*, 2009.
- [7] S. Durand, J. Minet, J. Castellanos, and N. Marchand, "Asyn-car, a radio-controlled vehicle for asynchronous experiments implementation of an event-based cruise control," in *Proceedings of the 8th International Conference on Electrical Engineering Computing Science and Automatic Control (CCE)*, Oct 2011, pp. 1–6.
- [8] W. P. M. H. Heemels, J. H. Sandee, and P. P. J. Van Den Bosch, "Analysis of event-driven controllers for linear systems," *International Journal of Control*, vol. 81, no. 4, pp. 571–590, 2008.
- [9] F. L. Roubieu, F. Expert, G. Sabiron, and F. Ruffier, "Two-directional 1-g visual motion sensor inspired by the fly's eye," *Sensors Journal, IEEE*, vol. 13, no. 3, pp. 1025–1035, march 2013.
- [10] S. Viollet, L. Kerhuel, and N. Franceschini, "A 1-gram dual sensorless speed governor for micro-air vehicles," in *Proceedings of the 16th Mediterranean Conference on Control and Automation*, 25-27 June 2008, pp. 1270–1275.