Analysis of a Simple Feedback Scheme for Error Correction over a Lossy Network

Oscar Flärdh, Carlo Fischione, Karl H. Johansson and Mikael Johansson
Automatic Control Lab, School of Electrical Engineering
KTH (Royal Institute of Technology)
Stockholm, Sweden
Email: {flardh|carlofi|kallej|mikaelj}@ee.kth.se

Abstract—**A control theoretic analysis of a simple error correction scheme for lossy packet-switched networks is presented. Based on feedback information from the error correction process in the receiver, the sender adjusts the amount of redundancy using a so called extremum-seeking controller, which do not rely on any accurate model of the network loss process. The closed-loop system is shown to converge to a limit cycle in a neighborhood of the optimal redundancy. The result are validated using packet-based simulations with data from wireless sensor network experiments.**

I. INTRODUCTION

There is a wide range of applications using communication networks that need timing and reliability guarantees. Examples are found from multimedia streaming and IP telephony to networked embedded systems and industrial automation. Varying delays and information loss impose fundamental limitations for what can be achieved. For real-time applications, retransmission of lost data is not an option, but other techniques are needed. For certain application domains, such as multimedia streaming, several error concealment techniques have been proposed [1]. In the control literature, there are recent advances on how to cope with resource limitations in the communication of sensor and actuator data, e.g., [2]. In this paper, we take a different approach and ask the question on how to develop general mechanisms between the network and the application to handle the network imperfections.

A common way to deal with network imperfections is to introduce channel coding. By modifying the transmitted data (e.g., adding redundancy), the probability that the message is successfully transmitted can be increased. There is a tradeoff between quality and real-time performance, so the optimal amount of redundancy depends on the current network state. Hence, the goal of an adaptive coding scheme is to find an optimal operating point given the current application demand and the network state. Previous work in this area includes both analysis of coding with different amount of redundancy [3], [4] and algorithms for adapting the coding [4], [5].

Most adaptation schemes for error correcting codes rely on a model of the network loss process for determining the amount of redundancy. For general packet-switched networks, it is hard to obtain accurate network models. We have therefore studied an extremum seeking error control scheme that provide good adaptation without relying on a loss model. In [6] it was shown that by considering the outcome of the error correction process, it is possible to adapt the redundancy amount even in presence of model errors. This work was extended in [7] with

the introduction of a cost function and a hybrid control law that switches between feedforward and feedback control actions. In contrast to several other extremum seeking controllers, like [8], no local convex approximation of the objective function is needed. Simulations showed that the extremum-seeking feedback mechanism provides a robust way for finding the optimum without relying on a model of the loss process, while the feedforward change detection filter improves the transient behavior. This scheme is not intended to act as a new protocol but rather as a protocol booster. These are modules that can improve the performance of a protocol, but still being transparent to it. The main purpose of the protocol boosters is to enable faster evolution and customization of existing protocols, compared to changing or replacing the protocol itself [9]. In this context, error correction has been tested with encouraging results [10].

The main purpose of this paper is to provide a rigorous analysis of the feedback scheme for error correction discussed above. It is shown that the scheme can be considerably simplified under certain conditions, and that this simplified model is suitable for stability and performance analysis. The discrepancies between the original and the simplified models are discussed, and the analysis is validated with experimental data from a real sensor networks. The outline of the paper is as follows. In Section II the problem formulation is given, and Section III presents our models and the control architecture. Closed-loop analysis is done in Section IV and the experimental validation in Section V. Finally, Section VI concludes the paper.

II. PROBLEM FORMULATION

We consider a setup where packets are collected into blocks of fixed size N in the sender. In the block generated at time t, there are u_t redundancy packets and $N - u_t$ packets of application data. The so called minimum distance is $u_t + 1$, so if the code is maximum distance separable (e.g., a Reed Solomon code), then up to u_t packet losses can be corrected [11]. In other words, if at least any $N - u_t$ are received, all the original information can be recovered. When systematic coding is used, the original data are not encoded but sent as is in the $N - u_t$ data packets. Hence, information is transmitted even if more than u_t packets are lost. This amount will then depend on how many original application packets are actually received. Let $X_t^1 \in [0, N - u_t]$ and $X_t^2 \in [0, u_t]$ be stochastic variables denoting the number of received application and redundancy packets, respectively. Then the number of recovered data packets are

$$y_t = \begin{cases} N - u_t & \text{if } X_t^1 + X_t^2 \geq N - u_t \\ X_t^1 & \text{if } X_t^1 + X_t^2 < N - u_t \end{cases}$$

Fig. 1. The cost function (1) in the given setup for three different values of the packet loss probability.

If an insufficient number of application packets are received, some data is lost introducing a distortion on the decoding of the original message. The distortion can be measured in several ways. Here, we consider the distortion signal d_t to be simply the difference between the number of sent application data packets and the number of recovered data packets, *i.e.*,

$$d_t = N - u_t - y_t$$

This distortion is affected by the network loss process, denoted p_t, and the amount of redundancy used.

The cost function used in this analysis is given by

$$c(d, u) = \left(\frac{d}{N}\right)^2 + \rho \frac{u}{N} \tag{1}$$

where $\rho > 0$ is a given parameter. The cost function reflects that it is usually important to not allow high distortion or low rate of transmission information. By assuming a fixed packet loss probability it is possible to calculate the cost function, which is shown in Figure 1 for some values of p.

III. CONTROL STRUCTURE AND CLOSED-LOOP MODEL

In this section we present the control architecture for the error correction scheme and the closed-loop model of the feedback system.

A. Control Structure

Figure 2 indicates that both the network loss process p and the cost c are sent back from the receiver to the sender. Note that the plant for this feedback control system is the error correction process, with redundancy u as input and the cost c as output. While u has a direct influence on the cost c, it has little influence on the loss process. Thus, the signal p is in the figure considered as an external disturbance acting on the system and the cost can be seen as a function of u parameterized by p. In the following we discuss the feedback loop of this system. A discussion on the general control structure and the feedforward control based on p is given in [7].

The purpose of the feedback is to find the optimum of the cost function c. The optimum is unknown, but can be searched for implicitly since c is measured. To find the optimum, we therefore utilize an extremum-seeking controller. Such a controller for this specific setup is given by

$$u_{t+1} = u_t - \beta \operatorname{sgn}(u_t^d c_t^d) \tag{2}$$

Fig. 2. A general control structure for error correction with indication of feedforward and feedback information.

Fig. 3. Block diagram representing a simplified model of the systems in Figure 2. The controller in Figure 2 is given by the dashed line and the network by the nonlinear block $h(\cdot)$. This model is referred to as the *Double loop model*.

where $\beta > 0$ is a control parameter and u_t^d and c_t^d denote derivative estimates at time t of the signals u_t and c_t, respectively. The controller is inspired by the switching controllers in [12]. It basically estimates on which side of the peak of the cost function the system is, and then moves towards that peak.

B. Closed-Loop Model of the Feedback System

A continuous-time model for the extremum-seeking controller and the plant is derived next. Suppose the network loss process is constant $p_t = p$. Then we can view the cost as a static, nonlinear function of the redundancy:

$$c(t) = h\left(u(t)\right)$$

The continuous version of the control law (2) is

$$\dot{u}(t) = -\beta \operatorname{sgn}(u^d(t)c^d(t)) \tag{3}$$

where the estimated derivatives are obtained by a linear filter $F(s)$. With these assumptions, we have the model, denoted *Double loop model*, shown in Figure 3. In the figure, we have $x(t) = u^d(t)$ and $w(t) = c^d(t)$.

Figure 4 represents a simplified model of Figure 3, and is called the *Single loop model*. It is obtained by opening the loop to the right in Figure 3, interpreting $w(t)$ as a disturbance, and introducing $G(s) = \frac{-\beta}{s}F(s)$. Note that $w(t)$ depends on $u(t)$ as well as on the nonlinearity and the filter $F(s)$. The purpose of this model is to use it for describing function analysis in next section.

IV. ANALYSIS

A more thorough investigation of the behavior of the feedback system can be done using describing function analysis. First the simplified control loop is studied. Simulations of the model in Figure 3 will also be used to investigate the agreement between the models. The analysis aims at understanding
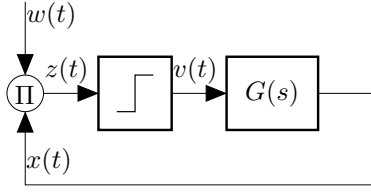
Fig. 4. Block diagram representing the simplified network model denoted the *Single loop model*. One of the loops of the Double loop model in Figure 3 has been cut and is represented by the external disturbance w.

and describing the solution to the control system during steady state. Our proposition is that that the feedback controller will minimize the cost without explicit knowledge of the cost function. This will be showed by proving that the solution will converge to a neighborhood of the optimum. Moreover, the solution converges to a stable limit cycle with amplitude and frequency dependent on the filter, the controller gain β and the nonlinearity.

A. Stability Analysis of Feedback Algorithm

First we prove that the feedback algorithm converges to an invariant neighborhood of the optimum. Suppose

$$u_{t+1} = u_t - \beta \operatorname{sgn}(\Delta u_t \Delta c_t) \qquad (4)$$

where Δ is defined as the difference operator, *i.e.* $\Delta u_t = u_t - u_{t-1}$. This difference is an idealization of the derivative approximation performed by the filter F.

Proposition 4.1: Assume that the relation between c and u are given by a static and unimodal function $h(\cdot) \in \mathcal{C}^1$, *i.e.* $c = h(u)$. Then the control law (4) will give convergence to a neighborhood of the optima u^*, in the sense that $\exists T : |u_t - u^*| < 2\beta, \forall t > T$. Moreover, the number of blocks it will take to converge is bounded above by

$$T \leq \left\lceil \frac{\min(|u^0 - (u^* - 2\beta)|, |u^0 - (u^* + 2\beta)|)}{\beta} \right\rceil + 2$$

with u^0 being the initial value.

Proof: Without loss of generality we may assume that $u^* = 0$. Moreover, since $h(\cdot)$ is assumed to be unimodal we have that

$$\begin{aligned} h'(u) < 0 &\text{ if } u < 0 \\ h'(u) > 0 &\text{ if } u > 0 \end{aligned} \qquad (5)$$

The time difference Δc_t can be expressed as

$$\begin{aligned} \Delta c_t &= c_t - c_{t-1} = h(u_t) - h(u_{t-1}) = \\ &= (u_t - u_{t-1})h'(\xi) = \Delta u_t h'(\xi), \quad \xi \in [u_{t-1}, u_t] \end{aligned}$$

with help from the mean value theorem. Inserting this in Equation (4) gives

$$\begin{aligned} u_{t+1} &= u_t - \beta \operatorname{sgn}(\Delta u_t \Delta c_t) = \\ &= u_t - \beta \operatorname{sgn}(\Delta u_t h'(\xi) \Delta u_t) = \\ &= u_t - \beta \operatorname{sgn}\left((\Delta u_t)^2 h'(\xi)\right) = \\ &= u_t - \beta \operatorname{sgn}(h'(\xi)), \quad \xi \in [u_{t-1}, u_t] \end{aligned}$$

For this system, at a given time $t = t_0$, there may be several possible situations.

We may first consider the case where both u_{t_0} and u_{t_0-1} are less than $u^* = 0$. In that case, $h'(\xi) < 0$ and the control

update will be $u_{t_0+1} = u_{t_0} + \beta$. Hence the control signal will increase towards the optimum. The case with both u_{t_0} and u_{t_0-1} greater than u^* is similar.

To see what happens when the control signal reaches the optimum, we consider the other case when u_{t_0} and u_{t_0-1} are on different sides of u^*. If

$$u_{t_0-1} < u^* < u_{t_0} \qquad (6)$$

then $h'(\xi)$ may have either sign and u_{t_0+1} may move either towards $(u_{t_0+1} = u_{t_0} - \beta)$ or from $(u_{t_0+1} = u_{t_0} + \beta)$ the optimum. In the first case, $|u_{t_0+1} - u^*| < \beta$ since

$$u^* < u_{t_0} < u^* + \beta \Rightarrow u^* - \beta < u_{t_0+1} < u^*$$

where the first inequality comes from (6) together with the control law (4) and the implication follows from $u_{t_0+1} = u_{t_0} - \beta$. If $u_{t_0+1} = u_{t_0} + \beta$, on the other hand, we have that

$$u^* < u_{t_0} < u^* + \beta \Rightarrow u^* + \beta < u_{t_0+1} < u^* + 2\beta$$

and $|u_{t_0+1} - u^*| < 2\beta$.

To conclude, we see that we will always move towards the origin if both u_t and u_{t-1} are on the same side of the optimum. This will always be the case if $|u_t - u^*| \geq \beta$. If $|u_t - u^*| < \beta$, on the other hand, u_t and u_{t-1} may be on each side of the optimum. In that case the state may move away to at maximum 2β from the optimum. If it does, we are back in the case when $|u_t - u^*| \geq \beta$ and the state will thus move towards the optimum again.

If the system is not started in this invariant region, time to converge to can be found by considering the system at some initial condition u^0. Since the initial differences may be incorrect, the system may initially move away from the optimum. On the next block, however, the differences will be correct and the system will move towards the optimum. The distance to the invariant region from u^0 is $|u^0 - (u^* - 2\beta)|$ if $u^0 < u^*$ and $|u^0 - (u^* + 2\beta)|$ if $u^0 > u^*$. With a step of β each block, it will take $\left\lceil \frac{\min(|u^0-(u^*-2\beta)|,|u^0-(u^*+2\beta)|)}{\beta} \right\rceil$ to cover this distance. Adding two blocks includes the case with an initial step away and then back again. ∎

B. System with Time Delay on the Feedback Channel

If there is a fixed an known time delay τ on the feedback channel, it is still possible to prove stability. This means that at time t, information on the cost at time $t - \tau$, *i.e.* $c_[t - \tau$, is available. Note that this can be achieved by using timestamps on the feedback packets. Hence, the control law in this case is

$$u_{t+1} = u_t - \beta \operatorname{sgn}(\Delta u_{t-\tau} \Delta c_{t-\tau}) \qquad (7)$$

Because of the time delay, however, the region to which the system converges will be larger and in fact also depend on the time delay.

Proposition 4.2: Assume that the relation between c and u are given by a static and unimodal function $h(\cdot) \in \mathcal{C}^1$, *i.e.* $c = h(u)$. Then the control law (7) will give convergence to a neighborhood of the optima u^*, in the sense that $\exists T : |u_t - u^*| < 2\tau\beta, \forall t > T$. Moreover, the number of blocks it will take to converge is bounded above by

$$T \leq \left\lceil \frac{\min(|u^0 - u^* + (2+\tau)\beta|, |u^0 + u^* - (2+\tau)\beta|)}{\beta} \right\rceil + 2\tau$$

with u^0 being the initial value.

Proof: Without loss of generality we may assume that $u^* = 0$. The time difference $\Delta c_{t-\tau}$ will then be, using the unimodal property (5) similarly as in the proof of Theorem 4.1,

$$\Delta c_{t-\tau} = \Delta u_{t-\tau} h'(\xi), \quad \xi \in [u_{t-\tau-1}, u_{t-\tau}]$$

Inserting this in Equation (7) gives

$$u_{t+1} = u_t - \beta \operatorname{sgn}(h'(\xi)), \quad \xi \in [u_{t-\tau-1}, u_{t-\tau}]$$

First, consider the case where $u_t, t \in [t_0 - \tau - 1, t_0]$ are all less than $u^* = 0$. In that case, $h'(\xi) < 0$ and the control update will be $u_{t+1} = u_t + \beta$. This will continue at most until $u_{t-\tau-1} > u^*$. Then, $u_t = u_{t-\tau} + \tau\beta$ since there has been an increase of β for τ time steps. u_t is then bound by $u^* + (\tau + 2)\beta$ since

$$u^* < u_{t-\tau-1} < u^* + \beta \Rightarrow u^* + (\tau+1)\beta < u_t < u^* + (\tau+2)\beta$$

The case with $u_t, t \in [t_0 - \tau - 1, t_0]$ are all being greater than u^* is similar.

The time to convergence is found in the same way as the proof of Theorem 4.1, with the invariant region now being $(\tau + 2)\beta$. Also, the initial conditions may drive the system in the wrong direction for τ samples. This, together with τ more to get back to the initial condition, adds 2τ to the number of blocks it takes to cover the distance from the initial condition to the invariant region. ∎

C. Describing Function Analysis

The two previous results showed that the extremum seeking controller is stable but not asymptotically stable. Hence it is interesting to further study the solution around the equilibria. Using describing function analysis, the following proposition can be proved.

Proposition 4.3: Assume that $w(t)$ is produced by a squared version of $x(t)$, with signals defined as in Figure 4. Moreover, $w(t)$ has zero static component and is phase shifted π compared to $x(t)$. Finally, $G(s)$ is considered as low-pass.

Then, the system in Figure 4 will fulfill the describing function conditions, with $x(t) = A\sin(\omega t)$ where A and ω are found by solving

$$\frac{4}{A\pi} e^{-i\frac{\pi}{2}} G(i\omega) = 1$$

Proof: First, let $x(t)$ be given as $A\sin(\omega t)$ and $w(t) = B\sin(2\omega t + \varphi)$. The modulated signal $z(t)$ will thus have two frequency components with the same amplitude, and the input to the relay is

$$z(t) = \bar{A}\sin(\overbrace{\omega t + \varphi + \pi/2}^{\Psi_1}) + \bar{A}\sin(\overbrace{3\omega t + \varphi - \pi/2}^{\Psi_2}) \quad (8)$$

with $\bar{A} = AB$. The output $v(t)$ from the relay is found by assuming a Fourier series expansion, and then finding the coefficients. This approach is described in [13]. A general representation of $v(t)$ is then given as

$$v(t) = \sum_{m=0}^{\infty} \sum_{\substack{n=-\infty\,(m\neq 0)\\n=0\,(m=0)}}^{\infty} [P_{mn}\sin(m\Psi_1 + n\Psi_2) + Q_{mn}\cos(m\Psi_1 + n\Psi_2)] \quad (9)$$

The coefficients P_{mn} and Q_{mn} are found by the following integrals

$$P_{mn} = \frac{1}{2\pi^2} \iint_{-\pi}^{\pi} \operatorname{sgn}(B\sin(\Psi_1) + B\sin(\Psi_2)) \\ \times \sin(m\Psi_1 + n\Psi_2)\, d\Psi_1\, d\Psi_2$$

$$Q_{mn} = \frac{1}{2\pi^2} \iint_{-\pi}^{\pi} \operatorname{sgn}(B\sin(\Psi_1) + B\sin(\Psi_2)) \\ \times \cos(m\Psi_1 + n\Psi_2)\, d\Psi_1\, d\Psi_2$$

First we note that, since sgn is odd, $Q_{mn} \equiv 0$. Evaluating P_{mn} gives

$$P_{mn} = \begin{cases} \frac{8}{\pi^2(m^2-n^2)} \sin\left(\frac{\pi}{2}(m-n)\right) & \text{for } m-n \text{ odd.} \\ 0 & \text{for } m-n \text{ even.} \end{cases} \quad (10)$$

Replacing Ψ_1 and Ψ_2 from (9) the output is

$$v(t) = \sum_{m=0}^{\infty} \sum_{\substack{n=-\infty\,(m\neq 0)\\n=0\,(m=0)}}^{\infty} P_{mn}\sin(m(\omega t + \varphi + \pi/2) \\ + n(3\omega t + \varphi - \pi/2)) \\ = \sum_{m=0}^{\infty} \sum_{\substack{n=-\infty\,(m\neq 0)\\n=0\,(m=0)}}^{\infty} P_{mn}\sin((m+3n)\omega t \\ + (m+n)\varphi + (m-n)\pi/2) \quad (11)$$

With the filter $G(s)$ being low-pass, we make the standard assumption in describing function analysis that frequencies above ω are damped significantly more than at ω. Therefore, the output components of frequency ω is the only one of interest. This corresponds to the terms of the sum (12) where $m + 3n = \pm 1$, for which $m + n$ is odd and thereby $P_{mn} \neq 0$ according to (11). The output with frequency ω is then (remember that $m \geq 0$)

$$v(t) = \sum_{n=-\infty}^{0} \frac{8}{\pi^2(8n^2 - 6n + 1)} \sin(\omega t + \varphi + \pi/2 - 2n\varphi) \\ + \sum_{n=-\infty}^{-1} \frac{8}{\pi^2(8n^2 + 6n + 1)} \sin(\omega t + \varphi + \pi/2 + 2n\varphi)$$

These sums are quite hard to evaluate for arbitrary φ, but simplifies significantly if $\varphi = j\pi, j \in \mathbb{Z}$. For example, if $\varphi = \pi$ then, with $k = -n$,

$$v(t) = \frac{8}{\pi^2} \left(\sum_{k=0}^{\infty} \frac{1}{8k^2 + 6k + 1} + \sum_{k=1}^{\infty} \frac{1}{8k^2 - 6k + 1} \right) \\ \times \sin(\omega t + 3\pi/2) \\ = \left(\frac{2\pi + 4\ln 2}{\pi^2} + \frac{2\pi - 4\ln 2}{\pi^2} \right) \sin(\omega t - \pi/2) \\ = \frac{4}{\pi} \sin(\omega t - \pi/2)$$

The signal $x(t) = A\sin(\omega t)$ in Figure 4 becomes $v(t) = \frac{4}{\pi}\sin(\omega t - \pi/2)$, and thus the describing function in this case is

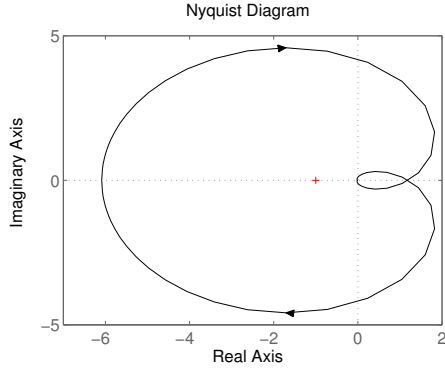$$N_\pi(A) = \frac{4}{A\pi} e^{-i\frac{\pi}{2}}$$

Fig. 5. The Nyquist plot indicates that the limit cycle is stable, since the Nyquist curve crosses the positive imaginary axis.
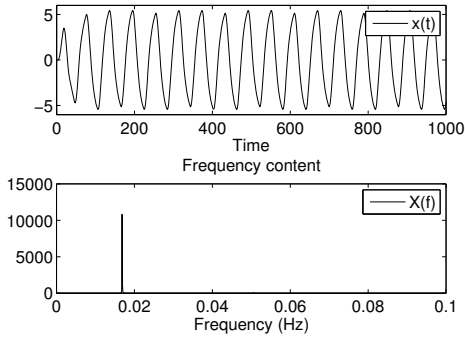

Fig. 6. A simulation of the Single loop model (Figure 4), where the oscillation fortifies our describing function assumption.
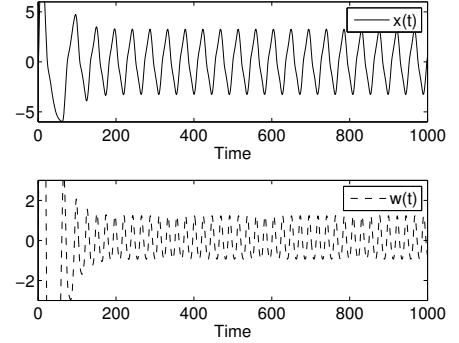

Fig. 7. A simulation of the Double loop model (Figure 3). The oscillation in $x(t)$ and $w(t)$ indicates that the Single loop model captures the main dynamic behavior.
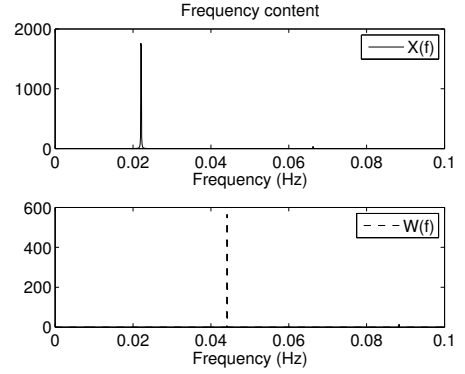

Fig. 8. A simulation of the Double loop model (Figure 3). The frequency is slightly higher than in the simplified model, but the assumptions with one frequency component in $x(t)$ and one in $w(t)$ with twice the frequency seem to hold.

With use of the describing function found above, the amplitude and frequency of the oscillation is computed by solving the following equation.

$$\frac{4}{A\pi} e^{-i\frac{\pi}{2}} G(i\omega) = 1 \qquad (12)$$

∎

A numerical example will illustrate the describing function analysis. Let the filter $G_d(z)$ be chosen as

$$G_d(z) = -\beta \frac{(1-a)(z+1)}{(z-a)^3} \qquad (13)$$

with parameters $a = e^{-0.2}$ and $\beta = 0.1$. A continuous approximation of (14) is then given by

$$G(s) = \frac{-0.01205s^2 + 0.04821s - 0.04821}{s^3 + 0.598s^2 + 0.1192s + 0.007921} \qquad (14)$$

This gives the solution to (13) as $[\omega, A] = [0.106, 5.35]$ which is an oscillation with frequency 0.017 Hz and amplitude 5.35. With help of the Nyquist plot of $G(i\omega)$ reported in Figure 5, we also note that the limit cycle is expected to be stable.

Simulation of the Single loop model with the parameters as in (15) gives the result shown in Figure 6. The amplitude is around 5.1 and from the frequency spectrum of the signal of $x(t)$ it is found that there is one major component with

frequency 0.017 Hz. This coincides very well with the results in the calculations, why the describing function approach seems to hold.

Now, it is also interesting to see how well this simplified model describes the more detailed Double loop model. By simulating that system, yet again with the same parameters, we can see that the behavior is not that different. In fact, the oscillations with amplitude 3.3 and frequency 0.022 Hz indicate that the simplified model captures the main dynamics and properties in the system.

The table below shows the frequency and amplitude of the oscillation with a few different parameter setups, comparing the two presented models with the predicted values from Proposition 4.3.

$[\beta \quad a]$		$[0.1 \; e^{-\frac{1}{5}}]$	$[0.2 \; e^{-\frac{1}{5}}]$	$[0.1 \; e^{-\frac{1}{2}}]$
Pred.	Amp.	5.35	10.71	1.22
	Freq.	0.017	0.017	0.037
Single	Amp.	3.3	6.5	0.86
	Freq.	0.022	0.022	0.048
Double	Amp.	5.1	10.1	1.2
	Freq.	0.017	0.017	0.037

V. VALIDATION

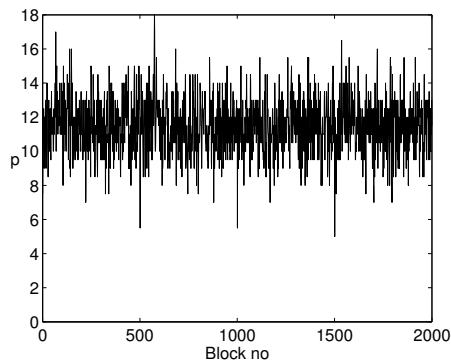To further investigate the feedback control scheme approach, we have validated it using packet based simulations

Fig. 9.   The loss data from the experiment on the wireless sensor network.
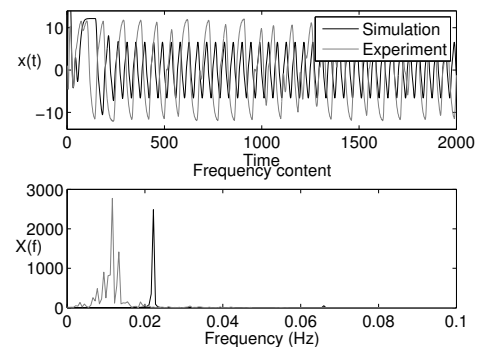

Fig. 10.   Comparing simulation of Model A (black line) with results of packet-based simulations using data from an experiment on a wireless sensor network (grey line). Model A predicts the stable limit cycle on the network system, with some discrepancy in frequency and amplitude.

with loss data from experiments on a wireless sensor network.

A. Test Bed Description

The test bed includes applications implemented in TinyOS [14] running on Telos motes [15] and a server (or gateway) software for bridging between the individual sensor network and the IP network. The Telos motes read data from their onboard sensors and send the readings to the server using the standard TinyOS multi-hop communication protocol. The server is in charge of the implementation of the algorithm, since the packet loss rate from the WSN nodes up to the node base station can be very high (even more then 10%), while the packet error rate from the server to the remote application may be small.

B. Experiments

The packet losses from the experiments on the Telos motes are shown in Figure 9. As seen, the average loss probability is quite high and the mean value during the experiment was 11.5%. This data were used for packet based simulations, and also compared with simulations of Model A using the cost function in Figure 1 with 11.5% packet loss probability. Time and frequency plots for $x(t)$ in both cases are shown in Figure 10. There is obviously a limit cycle present, just as the Double loop model predicts. The oscillation in the Double loop model has frequency 0.022 Hz and amplitude 6.5 while the packet based simulations gives an oscillation with approximately half the frequency (0.011 Hz) and an amplitude of 11. Even though the values differ by approximately a factor two, it indicates that the model can predict the main behavior with a limit cycle and also find the order of magnitude of the oscillation frequency and amplitude.

VI. Conclusion

A simple feedback scheme for error correction control has been studied. In the time domain, it has been proven to be stable, though not asymptotically stable, and describing function analysis in the frequency domain has shown the existence of a limit cycle. This limit cycle is desired for this type of extremum seeking controllers, and enables it to find the optimum of an unknown cost function.

The results obtained has also been validated against packet based simulations using data from experiments on a wireless sensor network. This comparison indicates that the results are relevant also for real networks, with presence of network characteristics such as loss bursts as well as implementation aspects such as discretization and quantization.

Future work includes the full implementation of this scheme on the sensor network test bed to have an even more realistic validation of this scheme. It is also interesting to try to improve the models presented here to give more accurate predictions of the solution properties.

References

[1] D. Miras, "A survey of network qos needs of advanced internet applications," Working Document, Computer Science Department, University College London, Tech. Rep., 2002.

[2] Z. Wei, M. Branicky, and S. Phillips, "Stability of networked control systems," *IEEE Control Systems Magazine*, vol. 21, no. 1, pp. 84–99, 2001.

[3] C. Barakat and A. A. Fawal, "Analysis of link-level hybrid fec/arq-sr for wireless links and long-lived tcp traffic," *Performance Evaluation Journal*, vol. 57, no. 4, pp. 43–500, 2004.

[4] H. Lundqvist, "Error and traffic control for high-speed networks," Ph.D. dissertation, Royal Institute of Technology (KTH), 2005.

[5] K. Park and W. Wang, "AFEC: an adaptive forward error-correction protocol and its analysis," Technical Report CSD-TR97-038, Department of Computer Sciences, Purdue University, Tech. Rep., 1997.

[6] O. Flärdh, K. H. Johansson, and M. Johansson, "A new feedback control mechanism for error correction in packet-switched networks," in *44th IEEE CDC - ECC*, 2005.

[7] O. Flärdh, C. Fischione, K. H. Johansson, and M. Johansson, "A control framework for online error control adaptation in networked applications," in *IEEE Second International Symposium on Communications, Control and Signal Processing*, Mar. 2006.

[8] J.-Y. Choi, M. Krstic, K. Ariyur, and J. Lee, "Extremum seeking control for discrete-time systems," *IEEE Transactions on Automatic Control*, vol. 47, no. 2, pp. 318–323, 2002.

[9] D. Feldmeier, A. McAuley, J. Smith, D. Bakin, W. Marcus, and T. Raleigh, "Protocol boosters," *IEEE JSAC, Special Issue on Protocol Architectures for 21st Century*, vol. 16, no. 3, pp. 437–44, 1998.

[10] D. Bakin, W. Marcus, A. McAuley, and T. Realeigh, "An fec booster protocol over terrestrial and satellite wireless networks," in *International Mobile Satellite Conference*, June 1997.

[11] R. E. Blahut, *Theory and Practice of Error Control Codes*. Addison-Wesley, 1983.

[12] J. Sternby, "A review of extremum control," Department of Automatic Control, Lund Institute of Technology, Tech. Rep., April 1979.

[13] A. Gelb and W. E. V. Welde, *Multiple-Input Describing Functions and Nonlinear System Design*. McGraw-Hill Electronic Sciences Series, 1968.

[14] "Tinyos alliance." [Online]. Available: http://www.tinyos.net/

[15] J. Polastre, R. Szewczyk, D. Culler, "Telos: Enabling ultra-low power wireless research," *IPSN/SPOTS*, April, 2005.