Accuracy-awareness: A pessimistic approach to optimal control of triggered mobile communication networks ⋆

Omar J. Faqir * **Eric C. Kerrigan** *,** **Deniz Gündüz** *

* *Department of Electrical & Electronic Engineering, Imperial College London, SW7 2AZ London, U.K. (e-mails: omar.faqir12@imperial.ac.uk; e.kerrigan@imperial.ac.uk; d.gunduz@imperial.ac.uk).*
** *Department of Aeronautics, Imperial College London, SW7 2AZ London, U.K.*

Abstract: We use nonlinear model predictive control to procure a joint control of mobility and transmission to minimize total network communication energy use. The nonlinear optimization problem is solved numerically in a self-triggered framework, where the next control update time depends on the predicted state trajectory *and* the accuracy of the numerical solution. Solution accuracy must be accounted for in any circumstance where systems are run in open-loop for long stretches of time based on potentially inaccurate predictions. These triggering conditions allow us to place wireless nodes in low energy 'idle' states for extended periods, saving over 70% of energy compared to a periodic policy where nodes consistently use energy to receive control updates.

Keywords: Autonomous mobile robots, Communication networks, Model-based control, Nonlinear control, Optimal control, Triggered control

1. INTRODUCTION

Energy consumption limits the use of autonomous agents, such as unmanned aerial vehicles (UAVs), with transmission and propulsion being large drains. A wireless nodes' transmission energy is the superposition of the energy usage in send, receive and idle (sleep) modes;

$$E_{\text{Transmission}} := E_{\text{send}} + E_{\text{receive}} + E_{\text{idle}}. \tag{1}$$

Idle nodes are neither listening for a message nor sending one, but can be woken by some internal clock mechanism. Typically $E_{\text{send}} \gg E_{\text{receive}} > E_{\text{idle}}$ and so it is advantageous to idle energy-scarce nodes whenever possible. In data gathering applications (e.g. mobile sensor networks (Thammawichai et al., 2018)) movement is used to facilitate transmission, and therefore should be considered part of the network's communication energy expenditure. Apart from energy, communication bandwidth is another limiting factor for UAVs and places restrictions on the amount of mission- and control-centric data that can be sent. Our proposed triggering scheme aims to reduce both the energy and bandwidth requirements of the network.

Zeng et al. (2016) uses a UAV as a relay between a stationary source and sink. Directional waterfilling is shown to maximize the throughput for a fixed trajectory. For a fixed transmission policy, the trajectory is optimized through a sequence of convex optimizations. Energy efficiency of a circular UAV trajectory is minimized in Zeng and Zhang (2017), while an optimal hover-fly-hover strategy is found by Wu et al. (2018) for a two-user broadcast UAV channel. Yang et al. (2018) trades-off a ground node's communication energy and a UAV's propulsion energy for circular or straight line flights. We use similar transmission and mobility models, as presented in Faqir et al. (2018), wherein we only considered a periodic control.

Apart from our previous works, most of the above determine optimal control strategies offline, but are not interested in using any optimization for a feedback control. Here we jointly control mobility and transmission to minimize the total network communication energy. As in Faqir et al. (2018), this is accomplished by formulating an optimal control problem (OCP) to minimize the energy required to aggregate mission-centric data. Here the novelty is to solve the OCP numerically in a triggered framework, where the optimization is solved again only if a given condition – which importantly accounts for the OCP solution accuracy – is met. This approach should be taken as a necessity when triggering conditions depend on approximate state predictions resulting from the numerical solution of an optimization. Gommans et al. (2014) argues that any metric of control performance should reflect the offline and online cost of controller implementation. Our approach allows to trade-off the control energy and bandwidth resource usage with the online cost of updating the control. We also generalize our previous framework (Faqir and Kerrigan, 2020b,a), which ties the numerical accuracy of the OCP with the time between control update events to

⋆ The support of the EPSRC Centre for Doctoral Training in High Performance Embedded and Distributed Systems (HiPEDS, Grant Reference EP/L016796/1) is gratefully acknowledged. D. Gündüz acknowledges funding from the European Research Council through project BEACON (Grant no. 677854).

allow for time-varying uncertainty bounds. Applying our scheme to the UAV communication problem allows us to trade-off the frequency and cost of the online optimization and idle inactive nodes, saving additional energy.

2. NONLINEAR OPTIMAL CONTROL

We can model a physical process by a set of ordinary differential equations (ODEs). When mathematically optimizing over these models the control enters as an algebraic variable, resulting in OCPs (Betts and Huffman, 1998). Take the nonlinear dynamics

$$\dot{x}(t) = f(t, x(t), u(t)) + v(t), \tag{2}$$

where $x(t) \in \mathbb{R}^n$ is the system state, and $u(t) \in \mathbb{R}^m$ the control input. The dynamics and output are affine in a disturbance $v(t) \in \mathbb{R}^n$ satisfying $|v_i(t)| \leq \hat{v}_i(t), i \in [1, n]$ while $f(\cdot)$ describes the nominal dynamics satisfying $\|f(t, x_1, u) - f(t, x_2, u)\| \leq L_x \|x_1 - x_2\|$ for all x_1, x_2, t, and u with Lipschitz constant L_x.

In nonlinear model predictive control (NMPC) we determine the input sequence as a result of a continuous-time OCP. NMPC can be computationally demanding but guarantees constraint satisfaction and is able to handle complex systems. We write the OCP in Bolza form as

$$\min_{\hat{x},\hat{u}} \ \Phi(\hat{x}(t_0), \hat{x}(t_f)) + \int_{t_0}^{t_f} L(\hat{x}(t), \hat{u}(t), t) \ \mathrm{d}t \tag{Pa}$$

$$\text{s.t. } \forall t \in [t_0, t_f],$$

$$\dot{\hat{x}}(t) = f(\hat{x}(t), \hat{u}(t), t), \tag{Pb}$$

$$c(\hat{x}(t), \hat{u}(t), t) \leq 0, \tag{Pc}$$

$$\phi(\hat{x}(t_0), \hat{x}(t_f)) = 0, \tag{Pd}$$

$$\hat{x}(t_0) = x(t_i^u). \tag{Pe}$$

Optimization is performed over a prediction horizon $T_{\text{op}} := t_f - t_0$. Internal variables $\hat{x}, \hat{u}$ represent the state and input trajectories. The cost consists of the Lagrange functional $L(\cdot)$ and Mayer functional $\Phi(\cdot)$. The solution satisfies $f(\cdot)$, enforced as ODEs in (Pb). We enforce path constraints (Pc) for all time and terminal constraints (Pd). Feedback is incorporated by re-solving problem (P) which, through (Pe), is parametric in the most recent state measurements $x(t_i^u)$, at every time $t_i^u \in \mathcal{T}_u$, where $\mathcal{T}_u$ is the set of input update times (IUTs).

We refer to $(x^*(\cdot), u^*(\cdot)) := \arg\min_{\hat{x}\hat{u}}$ (P) as the *true* solution of problem (P). There exist algorithmic or hardware approaches for efficiently solving this nonlinear optimization in real-time. None give the true solution $(x^*(\cdot), u^*(\cdot))$ but instead yield an *approximate* solution $(\tilde{x}, u)$. We can assess and guarantee the quality of these solutions through the process of mesh refinement.

2.1 Mesh refinement

Except for the simplest cases, the continuous-time dynamic optimization (P) is intractable or impossible to solve analytically. Direct transcription methods, an overview of which is given in Kerrigan et al. (2020), approximate (P) as a finite dimensional NLP, which is solved numerically. The NLP solution is interpolated into an *approximate* solution $\tilde{x}(\cdot), u(\cdot)$ (Betts, 2010). State trajectories may be approximated as continuous piecewise polynomials using a combination of h- and p-methods (known as hp-methods) (Kelly, 2017). In these methods the input is approximated as discontinuous piecewise polynomials. Define the interval $\Omega := (t_0, t_f)$, and $\overline{\Omega}$ as the closure of Ω.

Definition 1. (Mesh). The set $\mathcal{T}_{\text{h}}$ is called a **mesh** and consists of open intervals $T \subset \Omega$ satisfying conditions

(1) Disjunction $T_1 \cap T_2 = \emptyset \ \forall$ distinct $T_1, T_2 \in \mathcal{T}_{\text{h}}$,
(2) Coverage $\cup_{T \in \mathcal{T}_{\text{h}}} \overline{T} = \overline{\Omega}$,
(3) Resolution $\max_{T \in \mathcal{T}_{\text{h}}} |T| = \text{h}$,
(4) Quasi-uniformity $\min_{T_1, T_2 \in \mathcal{T}_{\text{h}}} \frac{|T_1|}{|T_2|} \geq \sigma > 0$,

where constant σ must not depend on mesh parameter h.

The indexed set of mesh points is $\mathcal{T}_m := \cup_{k \in \mathcal{K}_h} \inf \overline{T}_k$, where two polynomial segments are joined. Since the pair $(x^*(\cdot), u^*(\cdot))$ are unknown, they cannot be used to evaluate the solution accuracy. A possible metric is the *absolute local error* at time t in the i^{th} state, $\varepsilon_i(t) := \dot{\tilde{x}}_i(t) - f_i(t, \tilde{x}(t), u(t))$. The vector $\varepsilon(t) := [\varepsilon_1(t), \ldots, \varepsilon_n(t)]'$ is the error in the differential equations (Pb) resulting from direct transcription. The error quadrature for state i over mesh interval $T_k \in \mathcal{T}_m$ is $\eta_{k,i} := \int_{T_k} |\varepsilon_i(\tau)| \, \mathrm{d}\tau$, where $|\cdot|$ is the scalar norm. As in Betts (2010, Ch. 4), we may obtain a *relative* measure of the quadrature of the local error

$$\epsilon_{k,i} := \eta_{k,i}(w_i + 1)^{-1}, \tag{4}$$

where the scaling weight, $w_i := \max_{t \in \mathcal{T}_m} \left\{ \left| \dot{\tilde{x}}_i(t) \right|, |\tilde{x}_i(t)| \right\}$. In short, refinement schemes work by iteratively adding mesh points or increasing polynomial order until upper bounds are satisfied on the error quadrature in each mesh segment. Our numerical examples use an h- method, and our refinement algorithm is similar to that suggested in Betts (2010, Ch. 5), with refinement based on the *maximum relative local error* over all i state components in interval $T_k \in \mathcal{T}_m$, $\epsilon_k := \max_i \epsilon_{i,k}$. Although we use time-invariant tolerances, our analysis is easily extended to allow for different tolerances at each mesh interval, as done by Paiva and Fontes (2017).

2.2 Triggered control

Periodic controls are usually used without question. At each update period the system is measured, a control is calculated and applied. However, policies of this type are inherently open-loop because the control engineer must find a single sample interval that satisfies performance and stability criteria for all possible initial conditions and disturbance realizations (Gommans et al., 2014). Many current control applications are also restricted by resource usage. A worst-case periodic design leads to unavoidable over-allocation of communication and computation resources with no necessary gain in control performance.

Triggered control deviates from classic periodic paradigms by using feedback to dynamically determine sample times online. Two common approaches are event-triggered and self-triggered control (ETC and STC). In ETC the system is sampled continuously and the control is updated only when a certain 'triggering' condition is met. We formalize this in the following problem statement.

Problem 1. (**ETC**). Consider the dynamical system described by (2), and a time-varying state feedback law $u(t) = \mu(x(t), t)$ which renders the closed-loop system globally asymptotically stable (GAS) and/or satisfies

given performance criteria. Identify a set of state- and input- dependent conditions $F_{\text{event}} : \mathbb{R}^n \to \mathbb{R}$, resulting in update times

$$t^u_{i+1} = \inf\{t \in \mathbb{R}^+_0 | t > t^u_i, F_{\text{event}}(x(t)) \geq 0\},$$

for which the closed-loop system with sampled-data implementation is GAS and/or satisfies appropriately defined performance criteria).

F_{event} depends on the current state, and so the next update time may not be known in advance. Therefore, even if we reduce the *average* number of updates we cannot reallocate resources in the meantime. Compute time and communication bandwidth must always be kept available for the possibility of a control update. By contrast, in STC the next update t^u_{i+1} is determined at time t^u_i, using a triggering condition of the form $\tau^u_i = \inf\{\tau \in \mathbb{R}^+_0 | F_{\text{self}}(x(t^u_i), \tau) \geq 0\}$. As $F_{\text{self}} : \mathbb{R}^n \times \mathbb{R}^+ \to \mathbb{R}$ uses only the system measurements from time t^u_i to determine τ^u_i it allows for resource redistribution over interval τ^u_k. Define the error $\delta(t, t^u_k)$ between the predicted state $\tilde{x}(t, t^u_k)$, based on state measurement $x(t^u_k)$, and measurement $x(t)$ as $\delta(t, t^u_k) := \tilde{x}(t, t^u_k) - x(t)$. The components of the *prediction error* in each state are $\delta_i(t) := \tilde{x}_i(t) - x_i(t)$, with $\delta(t) := [\delta_1(t), \ldots, \delta_n(t)]^\intercal$ where $\delta(t) = 0, \forall t \in \mathcal{T}_u$, with the associated prediction error dynamics $\dot{\delta}(t) := \dot{\tilde{x}}(t) - \dot{x}(t)$. We employ the triggering condition

$$\tilde{F}_{\text{event}} := \Delta - \|\delta(t)\|_M , \tag{5}$$

$\|\cdot\|_M$ is assumed to be an appropriately defined p–norm with weighting matrix M.

3. SYSTEM MODEL

We consider a network of immobile nodes and a single mobile data gathering node, tasked with aggregating/distributing data among the network. Specifically, $\exists N_T$ stationary transmitting nodes $U_n, n \in \mathcal{N}_T := \{1, \ldots, N_T\}$ and N_R receiving nodes $U_m, m \in \mathcal{N}_R := \{N_T, \ldots, N_T + N_R\}$. These nodes are located at positions $X_n := (q_n, \delta_n, \varrho_n), n \in \{\mathcal{N}_T, \mathcal{N}_R\}$. The single mobile node U_0 travels with trajectory $t \to X_0(t) := (q_0(t), \delta_0(t), \varrho_0(t))$ over time $t \in \mathcal{T} := [0, T]$. The components of $X_n(t)$ are (time-varying) longitudinal, lateral and altitude displacements. U_0 is constrained to move from positions $X_0(0) = X_{0,\text{init}}$ and $X_0(T) = X_{0,\text{final}}$. We assume full and correct location information of all nodes to be known. At time t node $U_n, \in \mathcal{N}_0 := \{0, \mathcal{N}_T, \mathcal{N}_R\}$ has internal data load $s_n(t) \leq M_n$ bits, where M_n is the node's on-board memory. Storage buffers satisfy boundary conditions

$$s_n(0) := D_{n,\text{init}} \qquad s_n(T) := D_{n,\text{final}}, \qquad \forall n \in \mathcal{N}_0.$$

We can model an infrastructure-connected receiving node U_n as an ideal/infinite by setting $D_{n,\text{final}} = M_n = \infty$.

3.1 Mobility model

The UAV travels in a linear, constant altitude trajectory. For notational simplicity we set $\delta_0(t) = \delta_0$, $a_0(t) = a_0, \forall t \in \mathcal{T}$. UAV U_0 moves at speed $v_0(t) \in \mathcal{V} \triangleq [\underline{V}, \overline{V}]$, where $0 < \underline{V} \leq \overline{V}$. The propulsion force generated by U_0 must satisfy the Newtonian dynamic force balance equation $F(t) - D(v(t)) = Wa(t)$, where W is the mass and $a(t) \triangleq \dot{v}(t)$. $F(t)$ is the force used to accelerate and $D(\cdot)$ is a smooth drag model of the resistive forces on U_n. The instantaneous propulsion power is $F_n(t)v_n(t)$, with total propulsion energy as the integral of power over time (Zeng and Zhang, 2017). We model $D(v)$ of the fixed-wing UAV as the sum of parasitic and lift-induced drag $D(v) = C_{D1}v^2 + C_{D2}v^{-2}, \ \forall v \in \mathcal{V}$. We have taken $C_{D1} = 9.26 \times 10^{-4}$ and $C_{D2} = 2250$ for our simulations, as in Zeng and Zhang (2017). The parasitic drag is proportional to the speed squared, while lift-induced drag is inversely proportional to speed squared.

3.2 Slow fading wireless communications model

Wireless links exist from $U_n, n \in \{U_0, \mathcal{N}_T\}$ to $U_m, n \in \{U_0, \mathcal{N}_R\}$, $m \neq n$, over channels with corresponding gains $h_{mn} := v^2_{mn}$ with realization v_{mn}. χ_{mn} is the squared distance between U_n and U_m,

$$\chi_{mn}(t) := \|X_{mn}(t)\|^2 = \|(q_{mn}(t), \delta_{mn}(t), \varrho_{mn}(t))\|^2 ,$$

where $q_{mn}(t) := q_n(t) - q_m(t)$ and $\delta_{mn}, \varrho_{mn}$ are equivalently defined. The link gain is $\eta_{mn}(\chi_{mn}, h_{mn}) := h_{mn} G_{mn} \chi^{-\alpha/2}_{mn}$, where $\alpha > 1$ is the path loss exponent. $G_{mn} := \tilde{G}_{mn} d^\alpha_0$ is a unitless constant of receive and transmit antenna gains $\tilde{G}_{mn}$ at reference distance d_0. U_n transmits to U_m using a transmission power $p_{mn}(t) \in \mathcal{P} := [0, P_{\max}]$ Watts, at a strictly non-negative data rate $r_{mn}(t)$.

Messages for $U_m \in \mathcal{N}_R$ are transmitted over allocated bandwidth B_m, resulting in a multiple access channel (MAC). U_0 additionally transmits on bandwidths allocated to other nodes in $\mathcal{N}_R$. The set of achievable MAC rate tuples defines a polymatroid capacity region (Tse and Hanly, 1998). If N nodes transmit independent information to receiver U_m in the same interval, the received signal is a superposition of the N transmitted signals scaled by their respective channel gains $\eta(\chi_{mn}, h_{mn})$. The MAC capacity region $\mathcal{C}_{\tilde{N}}(\cdot)$ from sources $U_n, n \in \{U_0, \mathcal{N}_T\}$ to a single sink $U_m, m \in \{U_0, \mathcal{N}_R\}, m \neq n$ denotes the set of achievable rate tuples r, and is defined as

$$\mathcal{C}_{\tilde{N}}(\chi, p, h) := \left\{ r \geq 0 \mid f_m(\chi, p, r, h, \mathcal{S}) \leq 0, \forall \mathcal{S} \subseteq \tilde{\mathcal{N}} \right\},$$

$$f_m(\chi, p, r, h, \mathcal{S}) := \sum_{n \in \mathcal{S}} r_n - B_m \log_2 \left(1 + \sum_{n \in \mathcal{S}} \frac{\eta_{mn}(\chi_{mn}, h_{mn}) p_n}{\sigma^2_m} \right),$$

where χ is the tuple of distances χ_{mn}, $p \in \mathcal{P}^N$ is the N-tuple of transmission powers allocated by the N users on the channel reserved for node U_m. Further, r_n is the n^{th} component of r, $\sigma^2_m = 1$ is the receiver noise power. Throughput maximization can be achieved through the decoding process of successive interference cancellation (SIC) (Tse and Hanly, 1998).

Aerial channels are usually dominated by line-of-sight components, resulting in a flat fading channel with all signal components undergoing similar amplitude gains (Tse and Viswanath, 2005). We consider a slow fading channel, where the actual gains are random but remain constant over a certain communication interval, denoted t_p. For a random vector h, $\mathcal{C}_{\tilde{N}} = \emptyset$ with non-zero probability. Regardless of power and distance, we cannot guarantee successful transmission at any strictly positive rate with zero

probability of error [1] (Tse and Viswanath, 2005). However, the channel distribution is often known, even if the actual realization is unknown. In this case we propose using the $\epsilon-$*outage capacity* $\mathcal{C}^{\epsilon}_{\tilde{N}}$, defined as the set of achievable rates that guarantee a maximum outage probability of ϵ, namely $\mathcal{C}^{\epsilon}_{\tilde{N}} \triangleq \mathcal{C}_{\tilde{N}}(\chi, p, F_h^{-1}(1-\epsilon))$, where F_h is the complementary cumulative distribution of h, $F_h(x) \triangleq \Pr\{h \geq x\}$ (Tse and Viswanath, 2005). In doing so we will be performing chance-constrained optimization. However, since the probability density of h is known, the problem may be written deterministically with no additional complexity (Schwarm and Nikolaou, 1999).

3.3 Control strategy

Optimization is performed over state and input variables $\Phi_n := (s_n, p_n, r_n), \forall n \in \mathcal{N}$, and $\Psi_0 := (X_0, v_0, a_0, F_0)$. p_n is the tuple of outgoing transmission powers $p_{mn}(t)$ and r_n is similarly defined. The continuous-time OCP solved at times $\mathcal{T}_u$ to generate a control input is

$$\min_{\Psi_n, n\in\mathcal{N}} \int_{0,T} \sum_{\mathcal{N}_0} p_n(t) + v(t)F(t)\mathrm{d}t \tag{6a}$$

$$\text{s.t. } \forall n \in \{U_0, \mathcal{N}_T\}, m \in \{U_0, \mathcal{N}_R\}, t \in \mathcal{T}, \mathcal{S} \subseteq \mathcal{N}_R$$

$$f_m(\chi(t), p(t), r(t), \tilde{h}, \mathcal{S}\setminus\{m\}) \leq 0, \tag{6b}$$

$$\chi_{mn}(t) = \|X_{mn}(t)\|^2, \tag{6c}$$

$$\dot{s}_n(t) = \sum_{m\neq n} \left(r_{nm}(t) - r_{mn}(t)\right), \tag{6d}$$

$$\dot{X}_0(t) = f_{aero}(\Phi_0(t)), \tag{6e}$$

$$s_n(0) = D_{n,\text{init}}, \quad s_n(T) = D_{n,\text{final}}, \tag{6f}$$

$$X_n(0) = X_{n,\text{init}}, \; X_n(T) = X_{n,\text{final}}, \tag{6g}$$

$$\underline{\Phi}_n \leq \Phi_n(t) \leq \overline{\Phi}_n., \; \underline{\Psi}_0 \leq \Psi_0(t) \leq \overline{\Psi}_0. \tag{6h}$$

The cost function (6a) is the sum of transmission energy of all nodes and the mobility energy of U_0. Dynamic stage constraints (6b)–(6c) bound the achievable data rates to be within the $\epsilon-$*outage capacity* of each receiving node. Transmission dynamics are enforced in (6d), which updates the storage buffers with the sent and received data. Mobility dynamics are included in (6e). Boundary conditions provide initial and final requirements on the state of the network; variable bounds are included in (6h). The continuous-time problem is transcribed using direct-collocation and solved with the open source primal dual interior point solver IPOPT (Wächter and Biegler, 2006). Problem (6) is solved centrally on U_0, assumed to have more on-board compute. Control trajectories are then disseminated to all other nodes. Although power aware control of control-centric data (vs mission-centric data) is not included in problem (6), this does use additional energy — particularly if nodes are unaware of *when* they will receive control updates.

3.4 Feedback and decoding strategy

We determine a decoding procedure for received data across the MAC. Data will be encoded and sent in discrete codewords over packet intervals t_p. At each update time t_i^u, the complete infomation at each node is encoded at a rate determined by (6). Feedback occurs through a repeat request (ARQ) protocol where transmitters receive 1-bit acknowledgement (ACK/NAK) signals. Buffers are updated with successfully decoded information, while an unsuccessfully decoded packet is retransmitted at a later time. Take a single MAC channel over a single codeword interval, dropping time dependency in notation. For actual realization $\tilde{h}_{mn}$, channel outage — where decoding is unsuccessful — occurs because one or more of the received powers $\tilde{\beta}_n \triangleq \eta(\chi_{nm}, \tilde{h}_{mn})p_{mn}$ was smaller than predicted and cannot support rate r_n. The decoder may perform joint decoding of received signals, or decode a subset of received signals, treating others as interference. Precisely, for an N-user MAC, information from users in $\mathcal{S} \subseteq \mathcal{N} \subseteq \mathcal{N}_T$ is successfully decoded if

$$r \in \mathcal{D} \triangleq \bigcup_{\mathcal{S}\subset\mathcal{N}} \left\{ r > 0 \mid \sum_{m\in\mathcal{M}} r_m - B\log_2\left(1 + \underbrace{\frac{\sum_{n\in\mathcal{M}}\tilde{\beta}_n}{\sigma^2 + \sum_{s\in\mathcal{S}'}\tilde{\beta}_s}}_{\text{SNIR}}\right) \leq 0, \forall \mathcal{M} \subset \mathcal{S} \right\}$$

where $\mathcal{S}' \triangleq \mathcal{N}\setminus\mathcal{S}$ is the set of users not decoded, treated only as interference. Rician fading is suitable for modelling channels with strong LoS components (Zhou et al., 2012). For each channel used, ν is a vector of random variables drawn from a Rice distribution characterized by K-factor κ, defined as the ratio of received signal power in the LoS path to the power received from scattered paths. The cumulative distribution $\Gamma(\cdot)$ of a Rician channel is an order 1 Marcum Q-function. In simulations we set $\kappa = 10$, and assume that different users' fading processes are independent and identically distributed; $v_{nm} \sim \text{Rice}\left(\sqrt{\kappa(\kappa+1)^{-1}}, \sqrt{(2(\kappa+1))^{-1}}\right)$, for each fading instance and pair of nodes n, m. Additionally assume that the control data packets are always successfully sent. Simulations are performed for $\Gamma^{-1}(1-\varepsilon) \approx 0.3$.

4. ACCURACY-BASED TRIGGERING

Many optimal control software return the error distribution ϵ as well as the solution trajectories. If triggering conditions make use of the predictions resulting from the optimization, then we must account for this error. Here we present a lower bound for the inter-update time based on the error distribution of the solution trajectories. Results of this nature were initially presented by Faqir and Kerrigan (2020b,a) for systems effected by noise $\|v(t)\| \leq \hat{v}$. However in the application of interest in this work we consider component-wise and time-varying noise bounds. This will allow us to use the command (predicted) data rates determined by the OCP solution as bounds on the uncertainty in data transfer.

Theorem 1. (Quadrature Error Triggering). The IUT

$$\tau_i^{\text{QET}} = \sup\left\{\tau > 0 \middle| \left(\left\|\sum_{T\in\mathcal{T}_t}\epsilon_k\right\|_M + \int_0^\tau \|\hat{v}(t)\|_M \,\mathrm{d}t\right) e^{L_x\tau} \leq \Delta\right\} \tag{7}$$

[1] This is under the assumption that the transmitter has no channel state information — and hence, cannot perform power allocation — and that h_{mn} cannot be bounded below by a positive value with probability 1, that is, $P\{h_n \leq \epsilon\} > 0, \forall \epsilon > 0$.

based on $\tilde{x}(\cdot)$, η_k, which approximately solves problem (P) at time t_i^u, satisfies $\|\delta(t_i^u+\tau)\| \leq \Delta, \forall \tau \leq \tau_i^{\text{QET}}$.

For completeness, a proof may be found in Appendix A. Intuitively, we can see that because we compute trajectories with less error (smaller ϵ), we can run the system in open-loop for longer between updates. Consider a multi-hop MAC scenario, where transmitting ground nodes U_1, U_2 send data to receiver U_3 via UAV U_0 passing overhead. No bandwidth is allocated for direct messages between ground nodes. We initialize data buffers $D_{1,\text{init}} = 0.78, D_{2,\text{init}} = 0.48\,\text{MB}$. In km, the ground nodes are located at $X_1 = (0,0,0)$, $X_2 = (0,100,0)$, and $X_3 = (1200,0,25)$. U_0 moves between $X_{0,\text{init}}(t) = (-1.17,0,50)$ and $X_{0,\text{final}}(t) = (1.17,5000,50)$. The transmission dynamics in (6) are written as $\dot{x} = f(x(t),u(t))$, $g(x(t),u(t)) \leq 0$, and do not fall into the class of systems for which Theorem 1 holds. By index reduction we pose the problem in explicit ODE form, at the expense of increasing problem size (Kerrigan et al., 2020). To do so we introduce a fictitious decoding order $\varphi(t) \in [0,1], \forall t \in T_{\text{opt}}$. The MAC dynamics of senders U_1, U_2 are

$$\dot{s}_i(t) = B_0 \left(\varphi(t) \log_2 \left(1 + \frac{\eta_{0i}(\chi_{0i}(t), h_{0i}) p_i(t)}{\sigma_0^2} \right) + (1-\varphi(t)) \log_2 \left(1 + \frac{\eta_{0i}(\chi_{0i}(t), h_{0i}) p_i(t)}{\sigma_0^2 + \eta_{0j}(\chi_{0j}(t), h_{0j}) p_j(t)} \right) \right),$$

for $i = \{1,2\}, j = \{2,1\}, j \neq i$. We use these expressions instead of (6b). Writing transmission dynamics $s_0(t)$ as

$$\dot{s}_0(t) = \dot{s}_1(t) + \dot{s}_2(t) - B_3 \log_2 \left(1 + \frac{\eta_{30}(\chi_{30}(t), h_{30}) p_0(t)}{\sigma_3^2} \right),$$

we will be in a position to apply the STC from Theorem 1 once we determine noise bounds. These are slow-fading channels with some probability of outage — meaning unsuccessful data transmission. It is never possible to transmit a negative amount of data, nor it is possible to transmit more data than we encode. Thus, the actual data change in storage buffer $s_i(t)$ will always be an element in the range $[0, \dot{s}_i(t)]$. We can model these channels as being affected by the additive noise $v_i(t), |v_i(t)| \leq \hat{v}(t) = s_i(t), i \in \{0,1,2\}$. A key consequence is that condition (7) now depends on the *actual* predicted trajectories as well as the *actual* error distribution of the solution. This focuses control updates around the peak transmission intervals, which allows us to successfully idle nodes at other times.

Closed-loop state and input trajectories are shown in Figures 1 and 2 for a weight matrix M with diagonal $[0.01, 10.0, 10.0, 10.0]$, thresholds $\Delta = 0.075$ and $\hat{\epsilon} = 0.01$. This choice of values guarantees all IUTs are larger than the packet interval t_p. Both U_1, U_2 successfully transmit all required data to UAV U_0. The closed-loop policy offloads all but 0.15 MB from U_0, which is 11% of the total starting data. Our policy uses 47 control updates with an average IUT of 3.74 s and a minimum IUT of 0.26 s. We see both mesh points (crosses) and triggered updates (circles) are centered around transmission events. Outside of the transmission events and control updates we can set nodes to 'idle,' where they are not awaiting data reception. From (1), the total *receive* and *idle* network energy is only 30% compared to if we had updated the

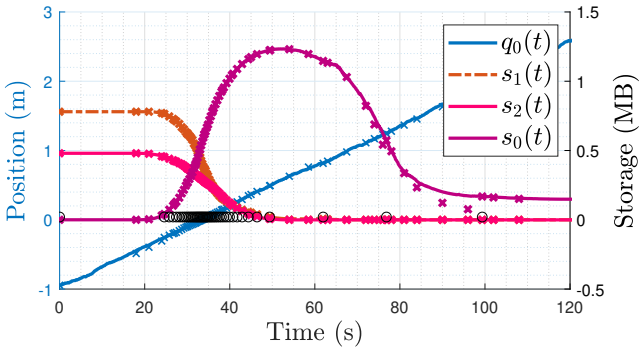

Fig. 1. Closed-loop state trajectories of triggered control based on Theorem 1. Control update times $\mathcal{T}_u$ are shown by open circles.

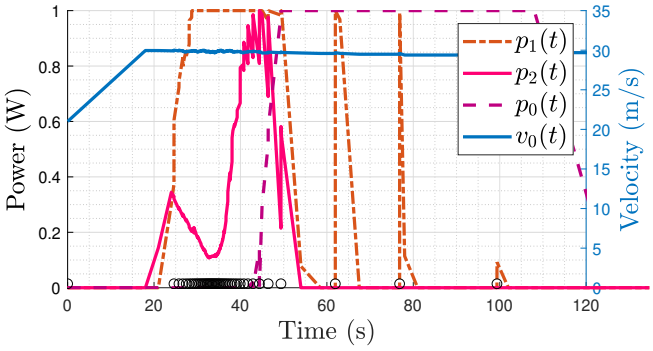

Fig. 2. Closed-loop transmission powers and command velocity of triggered control based on Theorem 1. Control update instances $\mathcal{T}_u$ are shown by open circles.

control every 0.26 s, coinciding with the minimum IUT. In fact, if there were no positional uncertainty, our scheme would only trigger during transmission events. Accounting for positional uncertainty allows us to partially pre-empt situations where the UAV may travel past the transmitting node, losing access to the most favourable channels.

If we were to solve (6) to a higher accuracy, then we may be able to reduce the number of updates further. However the possibilities of channel outages limit the potential of this, and means there is little gain from using more compute for a better prediction. On the contrary, condition (7) allows us to actually use a relatively poor prediction in the STC while still being assured of the satisfaction of (5).

5. CONCLUSIONS

Accounting for solution accuracy is imperative in situations where we may run systems in open loop for extended periods based on potentially inaccurate predictions. Combining mesh refinement with triggering conditions like (7) yield an effective framework for trading off the cost of computation with the frequency of computation/communication. We have applied these principles to triggered nonlinear predictive control of a UAV-enabled wireless network, showing energy savings of over 70% by 'idling' nodes when not awaiting control or mission data.

REFERENCES

Betts, J.T. (2010). *Practical methods for optimal control and estimation using nonlinear programming*, volume 19. Siam.

Betts, J.T. and Huffman, W.P. (1998). Mesh refinement in direct transcription methods for optimal control. *Optimal Control Applications and Methods*, 19(1), 1–21.

Faqir, O. and Kerrigan, E. (2020a). Mesh refinement for event-triggered nonlinear model predictive control. In *21st International Federation of Automatic control (IFAC) World Congress*, –. IFAC.

Faqir, O. and Kerrigan, E. (2020b). Inaccuracy matters: accounting for solution accuracy in event-triggered nonlinear model predictive control. *Submitted to IEEE Transactions on Automatic Control*.

Faqir, O., Yuanbo, N., Kerrigan, E.C., and Gündüz, D. (2018). Energy-efficient communication in mobile aerial relay-assisted networks using predictive control. In *Nonlinear Model Predictive Control (NMPC), 2018 IFAC 6th Conference on*. IFAC.

Gommans, T., Antunes, D., Donkers, T., Tabuada, P., and Heemels, M. (2014). Self-triggered linear quadratic control. *Automatica*, 50(4), 1279–1287.

Kelly, M. (2017). An introduction to trajectory optimization: How to do your own direct collocation. *SIAM Review*, 59(4), 849–904.

Kerrigan, E.C., Nie, Y., Faqir, O., Kennedy, C.H., Niederer, S.A., Solis-Lemus, J.A., Vincent, P., and Williams, S.E. (2020). Direct transcription for dynamic optimization: A tutorial with a case study on dual-patient ventilation during the COVID-19 pandemic. In *2020 59th IEEE Conference on Decision and Control (CDC)*, 2597–2614. IEEE.

Paiva, L.T. and Fontes, F.A. (2017). Sampled–data model predictive control using adaptive time–mesh refinement algorithms. In *CONTROL 2016*, 143–153. Springer.

Schwarm, A.T. and Nikolaou, M. (1999). Chance-constrained model predictive control. *AIChE Journal*, 45(8), 1743–1752.

Thammawichai, M., Baliyarasimhuni, S.P., Kerrigan, E.C., and Sousa, J.B. (2018). Optimizing communication and computation for multi-UAV information gathering applications. *IEEE Transactions on Aerospace and Electronic Systems*, 54(2), 601–615.

Tse, D. and Viswanath, P. (2005). *Fundamentals of wireless communication*. Cambridge university press.

Tse, D.N.C. and Hanly, S.V. (1998). Multiaccess fading channels. I: Polymatroid structure, optimal resource allocation and throughput capacities. *IEEE Transactions on Information Theory*, 44(7), 2796–2815.

Wächter, A. and Biegler, L. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1), 25–57. doi:10.1007/s10107-004-0559-y. URL `https://doi.org/10.1007/s10107-004-0559-y`.

Wu, Q., Xu, J., and Zhang, R. (2018). Capacity characterization of uav-enabled two-user broadcast channel. *IEEE Journal on Selected Areas in Communications*, 36(9), 1955–1971.

Yang, D., Wu, Q., Zeng, Y., and Zhang, R. (2018). Energy tradeoff in ground-to-uav communication via trajectory design. *IEEE Transactions on Vehicular Technology*, 67(7), 6721–6726.

Zeng, Y. and Zhang, R. (2017). Energy-efficient UAV communication with trajectory optimization. *IEEE Trans. Wireless Communications*, 16(6), 3747 – 3760.

Zeng, Y., Zhang, R., and Lim, T.J. (2016). Throughput maximization for mobile relaying systems. *IEEE Trans. Communications*, 64(12), 4983–4996.

Zhou, Y., Li, J., Lamont, L., and Rabbath, C.A. (2012). Modeling of packet dropout for uav wireless communications. In *2012 International Conference on Computing, Networking and Communications (ICNC)*, 677–682. IEEE.

Appendix A. PROOF OF THEOREM 1

The ETC (5) guarantees $\|\delta(t)\|_M \leq \Delta$. Without loss of generality, say (P) was solved at $t_i^u = 0$, with measurement $\tilde{x}(0) = x(0)$. The prediction error in state i is

$$\delta_i(t) = \int_0^t \dot{\tilde{x}}_i(\tau) - f_i(\tau, x(\tau), u(\tau)) + v_i(\tau)\mathrm{d}\tau. \quad \text{(A.1)}$$

As time-invariant scalar $w_i > 0$, we rewrite (4) as

$$\epsilon_{k,i} = \int_{t_k}^{t_{k+1}} \underbrace{\left|(\dot{\tilde{x}}_i(\tau) - f_i(\tau, \tilde{x}(\tau), u(\tau)))(w_i+1)^{-1}\right|}_{:=|\alpha_i(\tau)|} \mathrm{d}\tau.$$

Substituting $\dot{\tilde{x}}(t) = (w_i+1)\alpha(t) + f_i(x(t))$ into (A.1),

$$\begin{aligned}\delta_i(t) = \int_0^t &(w_i+1)\alpha_i(\tau) + f_i(\tau, \tilde{x}(\tau), u(\tau)) \\ &- f_i(\tau, x(\tau), u(\tau)) - v_i(\tau)\mathrm{d}\tau.\end{aligned}$$

Using the triangle inequality, the prediction error magnitude in each state is bounded from above as

$$\begin{aligned}|\gamma_i(t)| \leq (w_i+1)\int_0^t |\alpha_i(\tau)|\,\mathrm{d}\tau + \int_0^t |v_i(\tau)|\,\mathrm{d}\tau \\ + \int_0^t |f_i(\tau, \tilde{x}(\tau), u(\tau)) - f_i(\tau, x(\tau), u(\tau))|\,\mathrm{d}\tau. \quad \text{(A.2)}\end{aligned}$$

Now, consider the set $\mathcal{T}_t \subseteq \mathcal{T}_\mathrm{h}$ of open intervals of T, i.e. $\mathcal{T}_t := \{T \in \mathcal{T}_\mathrm{h} | T \cap \overline{(0,t)} \neq \emptyset\}$ which is a mesh on $\Omega_t := (0,t)$ and by construction satisfies the properties detailed in Definition 1. Mesh refinement certifies $\epsilon_{k,i} \leq \hat{\epsilon}_i$, so we bound the first integral in (A.2) as

$$(w_i+1)\int_0^t |\alpha_i(\tau)|\,\mathrm{d}\tau \leq (w_i+1)\sum_{k\in\mathcal{K}_t} \epsilon_{k,i},$$

We then bound the prediction error in the i^{th} state as

$$|\delta_i(t)| \leq (w_i+1)\sum_{k\in\mathcal{K}_t}\epsilon_{k,i} + L_x\int_0^t |\delta_i(\tau)|\,\mathrm{d}\tau + \int_0^t |v_i(\tau)|\,\mathrm{d}\tau,$$

where the first integral falls from Lipschitz $f(\cdot)$. Combining element-wise inequalities, we find the bound

$$\begin{aligned}\|\delta(t)\|_M \leq \left\|(w+1)\sum_{k\in\mathcal{K}_t}\epsilon_k\right\|_M + \\ \int_0^\tau \|\hat{v}(t)\|_M + L_x\|\delta(\tau)\|_M\,\mathrm{d}\tau\end{aligned}$$

From the integral form of the Gronwall-Bellman inequality,

$$\|\delta(t)\|_M \leq \left(\left\|(w+1)\sum_{k\in\mathcal{K}_t}\epsilon_k\right\|_M + \int_0^\tau \|\hat{v}(t)\|_M\right) e^{L_x t}$$

Restricting the RHS $\leq \Delta$ yields the explicit IUT in (7).