Modeling Round-off Error in the Fast Gradient Method for Predictive Control

Ian McInerney, Eric C. Kerrigan, and George A. Constantinides

***Abstract*— We present a method for determining the smallest precision required to have algorithmic stability of an implementation of the Fast Gradient Method (FGM) when solving a linear Model Predictive Control (MPC) problem in fixed-point arithmetic. We derive two models for the round-off error present in fixed-point arithmetic. The first is a generic model with no assumptions on the predicted system or weight matrices. The second is a parametric model that exploits the Toeplitz structure of the MPC problem for a Schur-stable system. We also propose a metric for measuring the amount of round-off error the FGM iteration can tolerate before becoming unstable. This metric is combined with the round-off error models to compute the minimum number of fractional bits needed for the fixed-point data type. Using these models, we show that exploiting the MPC problem structure nearly halves the number of fractional bits needed to implement an example problem. We show that this results in significant decreases in resource usage, computational energy and execution time for an implementation on a Field Programmable Gate Array.**

I. Introduction

Model Predictive Control (MPC) has played a large role in the rapid growth of cyber-physical systems by facilitating controllers that provide operational and safety guarantees. With the introduction of the internet of things, MPC is expected to play a similar role [1], but with control units that are smaller, cheaper and lower-power than today. Past results have demonstrated that first-order methods, such as the Fast Gradient Method (FGM), are well suited for these resource-constrained devices, since they require only simple computations (vector addition and matrix-vector multiplication), while having a fast solution time (on the order of micro-seconds) [2].

Field Programmable Gate Arrays (FPGAs) and resource-constrained embedded devices, such as microcontrollers, perform better with a fixed-point number representation than with floating-point due to the lack of an efficient floating-point computational unit. For this reason, many of the MPC algorithms such as FGM, dual gradient projection and proximal Newton have been implemented in fixed-point arithmetic — they have been analyzed to ensure algorithmic stability and that no values in the computations will be larger than the maximum representable value of the chosen data type.

Initial analysis for the FGM was done in [2], which proposed two design steps when choosing the parameters of the fixed-point data type. First, optimization problems are solved to compute the largest possible magnitude of the numbers in the computation. Second, a heuristic that uses the problem data chooses the precision so that the algorithm is stable in fixed-point. This work replaces the second design step with a new framework for computing the data type required to have stability of the FGM iteration in fixed-point.

As part of this framework, we present a new measure that we call the rounding stability margin, based on the pseudospectrum of the MPC Quadratic Program's (QP), to quantify how much round-off error can be experienced by the QP's Hessian before the FGM becomes unstable. We then present two models for the round-off error introduced by moving the Hessian into a fixed-point representation. The first is a generic model that can be applied to any MPC problem formulated as a QP, but depends on the length of the prediction horizon. The second is a structure-exploiting parametric model for the Constrained Linear Quadratic Regulator problem that requires the predicted system to be Schur-stable, but provides a horizon-independent error approximation.

These round-off error models are then combined with the rounding stability margin to compute the number of fractional bits required for algorithmic stability. We also examine how the rounding stability margin and the number of fractional bits needed changes as the cost function (e.g. the weighting matrices) is scaled. We demonstrate that using the structure-exploiting parametric model reduces the number of fractional bits needed by 30–45%, and reduces the hardware usage and solution time by up to 77% and 25%, respectively, for an FPGA implementation of FGM.

A. Notation

Let A' and A^* represent the transpose and conjugate transpose of the matrix A, respectively. The set of all eigenvalues of the matrix A is represented by $\lambda(A)$. The matrix norm of A is represented by $\|A\|_p$, where $p = 1, 2, \infty$ are the induced norms and $\|\cdot\|_{\max}$ is the largest absolute-value of the elements in the matrix. The H_∞ norm of a complex-valued function $\mathcal{P}_s(\cdot)$ is given by $\|\mathcal{P}_s\|_{H_\infty}$. For a matrix A, its fixed-point representation is given by $\hat{A}$, and the block on its i^{th} diagonal is given by A_i. The set $\mathbb{T}$ is defined to be the unit circle in the complex plane, i.e. $\mathbb{T} \coloneqq \{z \in \mathbb{C} \mid |z| = 1\}$.

II. Preliminaries

A. MPC Problem Formulation

In this work we focus on the input constrained LQR problem with a horizon of length N, which can be written

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.

The authors are with the Department of Electrical & Electronic Engineering, Imperial College London, SW7 2AZ, U.K. E.C. Kerrigan is also with the Department of Aeronautics. Email: {i.mcinerney17, e.kerrigan, g.constantinides}@imperial.ac.uk

 DOI: 10.1109/CDC40024.2019.9029910

as the quadratic program

$$\min_{u,x} \frac{1}{2}x'_N P x_N + \frac{1}{2}\sum_{k=0}^{N-1}(x'_k Q x_k + u'_k R u_k) \tag{1a}$$

$$\text{s.t. } x_{k+1} = Ax_k + Bu_k,\ k = 0,\dots,N-1 \tag{1b}$$

$$x_0 = \bar{x}_0$$

$$Eu_k \le c_u,\ k = 0,\dots,N-1 \tag{1c}$$

where $x_k \in \mathbb{R}^n$ and $u_k \in \mathbb{R}^m$ are the states and inputs at time k, respectively, and $\bar{x}_0 \in \mathbb{R}^n$ is the current measured system state. The matrices $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$ describe the discrete-time system $\mathcal{G}_s$. The stage constraints for the inputs are composed of the matrix $E \in \mathbb{R}^{j\times m}$ and the vector $c_u \in \mathbb{R}^j$. The matrices $Q = Q' \in \mathbb{R}^{n\times n}, R = R' \in \mathbb{R}^{m\times m}, P = P' \in \mathbb{R}^{n\times n}$ are the weighting matrices for the system states, inputs, and final states respectively, and are chosen such that $Q \succeq 0$, $R \succ 0$ and $P \succeq 0$.

We use the condensed MPC problem formed by removing the state variables from (1) to leave only the control inputs $u := \begin{bmatrix} u'_0 & u'_1 & \cdots & u'_{N-1}\end{bmatrix}'$ in the optimization problem

$$\min_u \frac{1}{2}u'Hu + \bar{x}'_0 J'u \tag{2a}$$

$$\text{s.t. } Gu \le g \tag{2b}$$

with the matrices described in [3, Appendix A].

In this paper, the numerical examples use the 2-input, 4-state dynamical system from [4] with $N = 20$ and weight matrices $Q = \text{diag}(0.1, 0.2, 0.3, 0.4), R = \text{diag}(0.01, 0.02)$ and P the solution of the discrete-time Lyapunov equation $A'PA+Q=P$.

B. Fixed-Point Number Representation

To store a fractional number, the fixed-point representation uses a fixed number of bits (the word length), which is divided into two segments: integer bits and fraction bits. The integer bits store the part of the number to the left of the radix point, while the fractional bits store the part to the right. The number of bits in each section is fixed at design-time, forcing a fixed range and precision on the numbers represented.

Any values larger than the number of integer bits will overflow, and have the information to the left of the integer portion lost. Any values with fractional components smaller than the precision of the number will have those parts rounded to the precision of the fixed-point number, with the difference between the represented number and the actual number termed the round-off error. The common rounding modes available for fixed-point numbers are shown in Table I along with their largest possible round-off error.

C. Fast Gradient Method

The Fast Gradient Method is an algorithm originally developed by Nesterov for solving a strictly convex optimization problem, and was subsequently adapted to solve the condensed CLQR problem (2) in [5]. This algorithm is an accelerated gradient descent method with the inequality constraints (2b) handled through a projection operator on u. To aid in its implementation, usually only upper/lower bounds are used in FGM on embedded platforms so that saturation can be used instead of projection.

TABLE I: Rounding modes in fixed-point arithmetic data types and their maximum round-off error.

Description	**Maximum Round-off Error with f fractional bits**
Round to $+\infty$	$\epsilon_f = 2^{-f}$
Round to 0 (truncation to 0)	$\epsilon_f = 2^{-f}$ for negative numbers $\epsilon_f = -2^{-f}$ for positive numbers
Round to $-\infty$ (truncation)	$\epsilon_f = -2^{-f}$
Round to ∞	$\epsilon_f = -2^{-f}$ for negative numbers $\epsilon_f = 2^{-f}$ for positive numbers
Round towards the nearest value (convergent rounding)	$\epsilon_f = \pm 2^{-(f+1)}$

Prior work has shown how to select the number of integer bits required to prevent overflow [2, Prop. 1], and also proposed the following necessary requirement to have stability of the FGM iteration in the presence of round-off errors.

Requirement 1 ([2, §IV-D]): For the Fast Gradient Method to be stable in fixed-point arithmetic, it is necessary (but not sufficient) for the fixed-point Hessian $\hat{H}$ to have all its eigenvalues in the open interval $(0, 1)$.

D. Matrix Pseudospectrum

In this work we will utilize a property of a matrix called its *pseudospectrum*.

Definition 1 (Pseudospectrum [6, §2]): Let $A \in \mathbb{C}^{n\times n}$ and $\epsilon > 0$ arbitrary. The ϵ-pseudospectrum $\lambda_\epsilon(A)$ of A is the set of $\tilde{\lambda} \in \mathbb{C}$ given by:

1. $\left\|(\tilde{\lambda}I - A)^{-1}\right\|_p > \frac{1}{\epsilon}$, (or $\lambda_{min}(\tilde{\lambda} - A) < \epsilon$ if $p = 2$).
2. $\tilde{\lambda} \in \lambda(A + E)$ for some $E \in \mathbb{C}^{n\times n}$ with $\|E\|_p < \epsilon$.

Note that statements 1 and 2 in Definition 1 are equivalent. Statement 1 presents the pseudospectrum as being the subset of the complex plane where the norm of the resolvent of A is greater than ϵ^{-1}. Alternatively, statement 2 presents the pseudospectrum as the subset of the complex plane containing the eigenvalues of A when A is perturbed by a matrix E with a given norm less than ϵ.

The pseudospectrum is a useful computational tool for describing the behavior of linear operators, especially those that are nonnormal. In this work though we focus on normal operators (since H is normal), which means that the ϵ-pseudospectrum can be interpreted as the set of numbers that are ϵ-close to an eigenvalue of A. This can be seen in Figure 1, where the inverse-resolvent of H is plotted.

III. Choosing the Fractional Precision

When the Fast Gradient Method is implemented using a fixed-point data format, care must be taken to ensure that Requirement 1 is satisfied. The placement of the eigenvalues of $\hat{H}$ is determined by both the number of integer bits i (through overflow), and also by the number of fractional bits f (through loss of precision). We will focus only on the fractional length and assume the integer length is chosen using [2, Prop. 1] to prevent overflow.

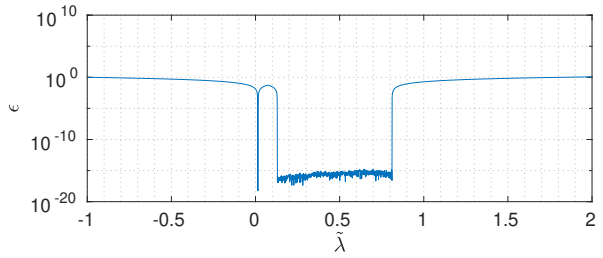
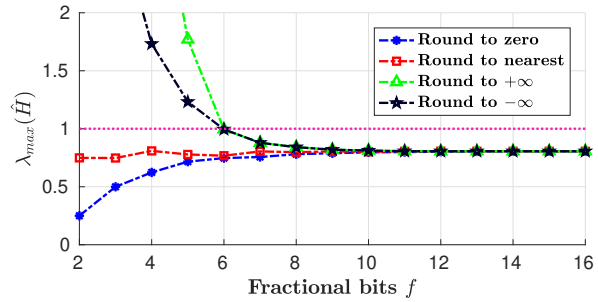
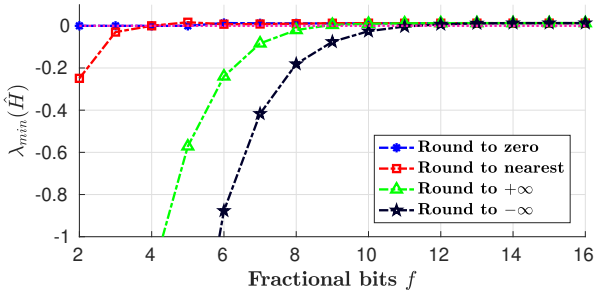


Fig. 1: The inverse of the resolvent of H when $\tilde{\lambda}$ is constrained to the real line. The ϵ-pseudospectra of H are the level-sets taken at the value ϵ.



(a) Maximum Eigenvalue



(b) Minimum Eigenvalue

Fig. 2: Extremal values of $\lambda(\hat{H})$ with $N = 50$.

The effect of the fractional length on the spectrum of the Hessian $\hat{H}$ can be seen in Figure 2 for four of the rounding modes available in fixed-point arithmetic. In this example, using round to zero/nearest gives a Hessian that is always Schur-stable but becomes indefinite for $f < 6$. If round to $\pm\infty$ is used, it is both unstable and indefinite in low precision. This shows that the rounding choice makes a large impact on the error between $\hat{H}$ and H, and must be factored into any analysis to determine the required bit length.

To analyze how the rounding affects the matrix, we model the round-off error as an additive matrix disturbance to H:

$$\hat{H} = H + E. \tag{3}$$

The values contained in the round-off matrix E in (3) will depend on the rounding mode chosen, but the magnitude of the values will always be less than the ϵ_f given in Table I.

To quantify the effect that the round-off error has on the spectrum of $\hat{H}$, we present a metric called the *rounding stability margin*.

Definition 2 (Rounding stability margin): Let $\hat{H}{=}H{+}E$ with $\|E\|_2 = \beta$ and $\lambda(H) \in (0, 1)$. The rounding stability margin η is the smallest value of β that causes the eigenvalues of $\hat{H}$ to leave the interval $(0, 1)$.

This margin represents the largest possible disturbance matrix that can be added to H before causing Requirement 1 to be violated. The margin can be calculated for symmetric matrices using the pseudospectrum of H as follows.

Lemma 1: Let $H = H'$ be a matrix with eigenvalues $\lambda(H) \in (0, 1)$. The rounding stability margin η of H is

$$\eta(H) = \min\left\{\|(-H)^{-1}\|_2^{-1}, \|(I - H)^{-1}\|_2^{-1}\right\}.$$

Proof: We begin by noting that H is symmetric, meaning its spectrum is composed of only real eigenvalues and that its ϵ-pseudospectrum will be an interval on the real line. By statement 2 of Definition 1, the eigenvalues of H perturbed by E with $\|E\|_2 < \epsilon$ will leave the interval $(0, 1)$ when either 0 or 1 is contained inside $\lambda_\epsilon(A)$. The largest allowable value of ϵ can then be computed using statement 1 of Definition 1 by evaluating the resolvent of H at the points 0 and 1, and computing $1/\epsilon$ at each point. ■

The result in Lemma 1 holds for any symmetric matrix H, so it can be used to find the rounding stability margin for the Hessian of (2) with any system or weight matrices.

A. Generic Rounding Model

We now present a framework for computing the necessary number of fractional bits for the FGM under a generic round-off error model that encompasses all of the rounding methods described in Table I. The basis for this model is that every element in H will experience an error of at most $\pm\epsilon_f$, so the worst-case perturbation matrix would then have entries of $\pm\epsilon_f$.

Definition 3 (Generic round-off error model): Let ϵ_f be the maximum round-off error created when a value is converted into a fixed-point representation with f fractional bits. Define $E_g \in \mathbb{R}^{k\times k}$ to be the worst-case component-wise round-off error matrix with $\pm\epsilon_f$ in every entry, i.e.

$$E_g := \begin{bmatrix} \pm\epsilon_f & \pm\epsilon_f & \dots \\ \pm\epsilon_f & \pm\epsilon_f & \dots \\ \vdots & \vdots & \ddots \end{bmatrix}. \tag{4}$$

Since each element of (4) is the same, modulo the sign, the matrix $1/\infty$-norms can be computed exactly and then be used to upper-bound the 2-norm of E_g as follows.

Lemma 2: Let $E_g \in \mathbb{R}^{k\times k}$ be the round-off error matrix from Definition 3 and let ϵ_f be the maximum round-off error possible with the rounding method. It follows that

$$\|E_g\|_2 \leq \|E_g\|_\infty = |\epsilon_f|k.$$

Proof: Recall that the matrix infinity (or one) norm is the largest absolute row (or column) sum of the matrix. Since the $1/\infty$ norms take the absolute value of the entries before summing them, the sign of the rounding error is irrelevant. Since H is symmetric, this gives $\|E_g\|_1 = \|E_g\|_\infty = |\epsilon_f|k$. Then, note that $\|E_g\|_2 \leq \sqrt{\|E_g\|_1\|E_g\|_\infty}$ [7, Fact 9.8.23], which means that $\|E_g\|_2 \leq \|E_g\|_\infty$. ■

To find the number of fractional bits needed to satisfy Requirement 1, we must find the number of bits needed to make $\|E_g\|_2 < \eta$. This can be done in closed-form, as follows.

Theorem 1: Let $f \in \mathbb{N}^+$ be the number of fractional bits in the fixed-point number representation, and ϵ_f the maximum round-off error a number may experience through rounding in that representation. If H has a rounding stability margin of η, then the number of fractional bits sufficient to guarantee that $\lambda(\hat{H}) \in (0,1)$ is

$$f = \begin{cases} \lceil -\log_2\left(\frac{\eta}{mN}\right) \rceil - 1 & \text{if using round to nearest,} \\ \lceil -\log_2\left(\frac{\eta}{mN}\right) \rceil & \text{otherwise.} \end{cases}$$

Proof: Using statement 2 of Definition 1 and the concept of the rounding stability margin introduced in Definition 2, we can say that we need $\|E_g\|_2 \leq \eta$ to guarantee that $\lambda(\hat{H}) \in (0,1)$. Using Lemma 2 and the fact that H is of dimension $mN \times mN$, we can transform that requirement into $|\epsilon_f| mN \leq \eta$. Turning the inequality into an equality and isolating ϵ_f then gives $|\epsilon_f| = \frac{\eta}{mN}$. Using the fact that $|\epsilon_f| \in \{2^{-f}, 2^{-(f+1)}\}$ depending on the rounding mode, we can substitute for ϵ_f and simplify to find f. ■

The closed-form expressions for f given in Theorem 1 hold for any rounding mode, and any system/weight matrix combination. Note that all rounding modes have the same fractional length with the exception of round to nearest, where 1 less bit is required.

The value of f from Theorem 1 is dependent upon both the horizon length and the system input dimension, and is monotonically increasing in both, as shown for the horizon length in Figure 3. This increase in the bound is caused by the monotonic increase in $\|E_g\|_\infty$ when the number of fraction bits is held constant and the horizon length increases. This means that in general the fraction length computed for a specific horizon can be used with a shorter horizon (e.g. in a decreasing-horizon controller), but may not be sufficient for longer horizons.

B. Parametric Rounding Model

The generic model in Section III-A is conservative for some rounding modes when applied to FGM with long horizons, especially round to nearest and round to zero. To reduce the conservatism of the error estimation, we introduce a parametric model for the round-off error experienced by H that incorporates knowledge of both the decay of terms in H and its Toeplitz structure. This model will only be valid in two rounding modes: round to nearest and round to zero.

A matrix is Toeplitz if it contains the same value down the entire length of each diagonal. Matrices of this type can be linked to a complex-valued function, called its matrix symbol, that has its Fourier coefficients given by the elements on the diagonals (see [8] for an overview). Recent work in [3] has shown that the matrix symbol for H is composed of the transfer function matrix for $\mathcal{G}_s$ and the weighting matrices, and provides horizon-independent analysis for Schur-stable systems. We can use this Toeplitz structure to then compute a horizon-independent η, as shown in Lemma 3.

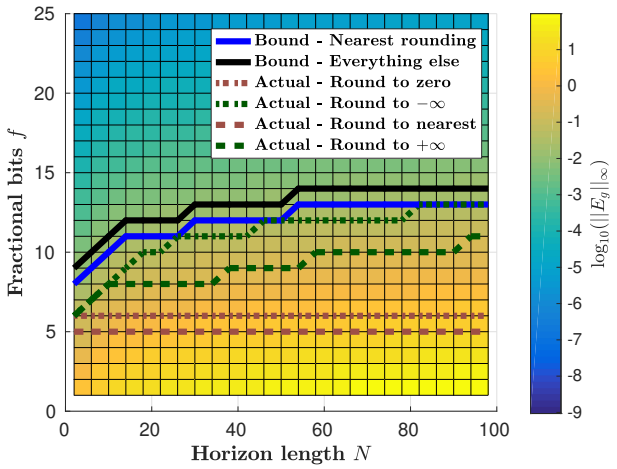


Fig. 3: The minimum number of fractional bits required for a given horizon length when using the generic rounding model from Theorem 1. The background shows the log of the bound for $\|E_g\|_2$ computed using Lemma 2.

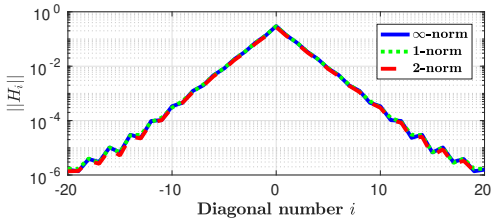


Fig. 4: Value of $\|H_i\|$ as the diagonal number grows.

Lemma 3: Let H be the Hessian from (2) with eigenvalues $\lambda(H) \in (0,1)$, $\mathcal{G}_s$ be Schur-stable, $\mathcal{P}_H$ the matrix symbol of H, and P be the solution of the discrete-time Lyapunov equation. Then the rounding stability margin η is

$$\eta(H) = \min\left\{ \left\|(-\mathcal{P}_H)^{-1}\right\|_{H_\infty}^{-1}, \left\|(I_m - \mathcal{P}_H)^{-1}\right\|_{H_\infty}^{-1} \right\}.$$

Proof: Since the resolvent in Definition 1 can be found using $\lambda_{min}(\tilde{\lambda} - H)$, we can use the results of [3] to replace H in Lemma 1 with its matrix symbol $\mathcal{P}_H$. ■

We further exploit the Toeplitz structure of H by noting that the diagonal blocks, H_i for diagonal $i \in \mathbb{Z}$, are given by

$$H_i = \begin{cases} B'(A^i)'PB & \text{if } i > 0, \\ B'PB + R & \text{if } i = 0, \\ B'PA^{|i|}B & \text{if } i < 0, \end{cases} \tag{5}$$

with $H_i = H'_{-i}$. We define diagonals as positive (or negative) if they are in the upper-triangular (or lower-triangular) region. If A corresponds to the state transition matrix of a Schur-stable system, then as the diagonal number i increases, the block H_i tends to 0, as shown in Figure 4.

The idea behind the parametric model is to exploit this decay and switch from modeling the worst-case round-off error to instead modeling the actual round-off error after a certain diagonal number. We now present a formal definition for this model.

Definition 4 (Parametric round-off error model): Let $T_i \in \mathbb{R}^{l\times l}$ be the block on the i^{th} diagonal of the Toeplitz matrix T that has the property that $\lim_{i\to\infty}\|T_i\|_{\max} = 0$. Let ϵ_f be the round-off error associated with the conversion to fixed-point representation using either round to nearest or round to zero, and k to be the diagonal beyond which all blocks in fixed-point representation $\hat{T}_i$ are 0, i.e.

$$k \coloneqq \min_i \left\{ i \in \mathbb{N}^+ \mid \|T_j\|_{\max} < \epsilon_f, \forall |j| \geq i, j \in \mathbb{Z} \right\}.$$

Define the parametric round-off error matrix as

$$E_p \coloneqq E_G + E_T,$$

where E_G is a matrix of bandwidth $k-1$ with E_g as its blocks, i.e.

$$(E_G)_i \coloneqq \begin{cases} E_g & \text{if } i < k, \\ 0 & \text{otherwise,} \end{cases}$$

and E_T is composed of the diagonal components of T that are after diagonal k, i.e.

$$(E_T)_i \coloneqq \begin{cases} T_i & \text{if } |i| \geq k, \\ 0 & \text{otherwise.} \end{cases}$$

Note that in this model, the value of k is inclusive of the first diagonal block which becomes 0, and $k \neq 0$, since this would imply that the entire matrix has been rounded to 0.

The matrix E_T in Definition 4 is Toeplitz, so its spectrum can be found using system-theoretic techniques similar to those in [3]. This then allows for the computation of the round-off error for the Hessian of (2) as follows.

Theorem 2: Let ϵ_f be the maximum round-off error when using either round to zero or round to nearest to convert H into fixed-point representation. Let η be the rounding stability margin of H from Lemma 3, and k be from Definition 4. If the parametric rounding model is used with a Schur-stable system $\mathcal{G}_s$ with P the solution to the discrete-time Lyapunov equation, then the fraction bit lengths sufficient for $\hat{H}$ to satisfy Requirement 1 also satisfies the inequality

$$|\epsilon_f| m(2k-1) + 2\|\mathcal{P}_{\bar{H}}(k,\cdot)\|_{H_\infty} < \eta,$$

where

$$\mathcal{P}_{\bar{H}}(n,z) \coloneqq z\mathcal{G}_P(z) - B'P\mathcal{P}_n(z)B \quad \forall z \in \mathbb{T},$$

$$\mathcal{G}_P \coloneqq \left\{ \begin{array}{l} x^+ = Ax + Bu \\ y = B'Px \end{array} \right. ,$$

$$\mathcal{P}_n(z) \coloneqq \sum_{i=0}^{n-1} A^i z^{-i} \quad \forall z \in \mathbb{T}.$$

Proof: To guarantee that Requirement 1 holds, we need to find when $\|E_p\|_2 < \eta$. First apply the sub-additive property of the matrix norm to get $\|E_G\|_2 + \|E_T\|_2 < \eta$. The matrix E_G is banded, with non-zero blocks on diagonals $\{-(k-1), \ldots, 0, \ldots, k-1\}$. This gives $\|E_G\|_2 \leq |\epsilon_f| m(2k-1)$, since there are $2k-1$ diagonals containing blocks of E_g with dimension $m \times m$.

Since E_T is a Toeplitz matrix, we can construct the Fourier series of the components of E_T as

$$\mathcal{P}_{\bar{H}} = \sum_{i=k}^{\infty} B'PA^iBz^{-i} + \sum_{i=k}^{\infty} B'(A^i)'PBz^i. \tag{6}$$

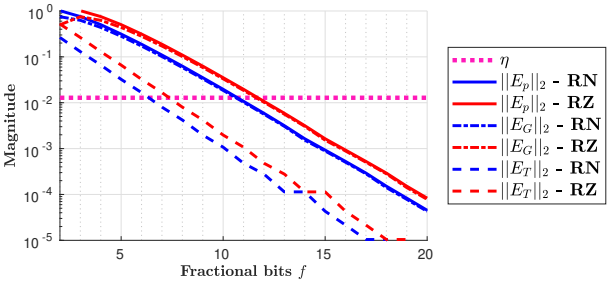


Fig. 5: The matrix norms from Theorem 2, with round to nearest (RN), and round to zero (RZ).

Adding and subtracting the first k terms of the summations then allows (6) to simplify to

$$\mathcal{P}_{\bar{H}} = B'P\left((I - z^{-1}A)^{-1} - \mathcal{P}_k(z)\right)B + B'\left((I - zA')^{-1} - \mathcal{P}_k(z)^*\right)PB. \tag{7}$$

The first term in (7) then can be simplifed further to

$$z\mathcal{G}_P(z) - B'P\mathcal{P}_n(z)B. \tag{8}$$

Note that the two terms in (7) are the conjugate transpose of each other. This means that when the H_∞ norm of $\mathcal{P}_{\bar{H}}$ is taken, it will simply be twice the H_∞ norm of (8). We can then use the H_∞ norm of $\mathcal{P}_{\bar{H}}$ as a size-independent upper bound for $\|E_T\|_2$. ■

To compute the fractional bit length from the inequality in Theorem 2, we do the following:

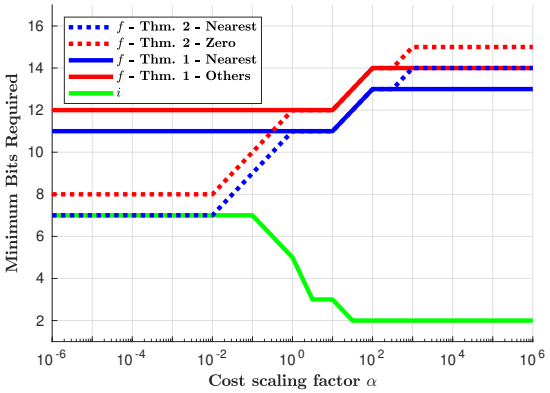
1) Compute $\|H_i\|_2$ using (5) for various values of i.
2) Iterate through each fraction length to determine if the inequality in Theorem 2 holds.

The result of these calculations can be seen in Figure 5 for the example problem. In this example E_G dominates E_T, so the worst-case round-off error from the non-zero banded component dominates the error caused by truncating the tail of H. The minimum number of fractional bits needed to satisfy Requirement 1 can be seen from the intersection of $\|E_p\|_2$ with η in Figure 5. It is also of note that the horizon length was not used in any of the calculations in Theorem 2, meaning the computed bit length is valid for any horizon.

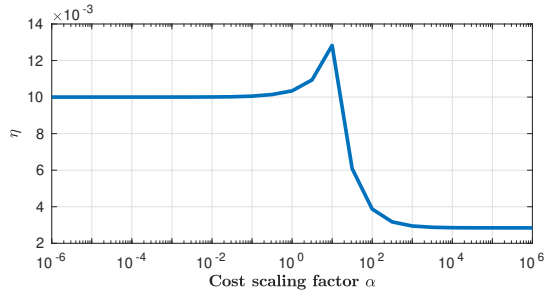
IV. Numerical Experiments

From Theorems 1 and 2, we can see a correlation between the problem data and the required data type size. To examine this, we performed experiments where Q was scaled by α and R was held constant, with the results reported in Figures 6 and 7. Note that for $\alpha > 10$ the eigenvalues of H leave $(0,1)$, so we introduce a scaling factor $c = \frac{1}{0.9\lambda_{max}(H)}$ to scale the matrices H and J in (2a) to bring the eigenvalues of H into $(0,1)$ before performing the analysis.

It can be seen that for small values of α, the number of fraction bits needed is small, with the structure-exploitation in Theorem 2 producing a saving of nearly 40% compared to the generic model from Theorem 1. Additionally, once α becomes large and the scale factor c is needed, the number of integer bits decreases. This decrease in integer bits offsets



(a) Fixed-point Bits required



(b) Rounding stability margin

Fig. 6: The effect of scaling the cost function on the rounding margin and fixed-point representation. Q was scaled by α while R was held constant. The finite horizon used was $N = 20$.

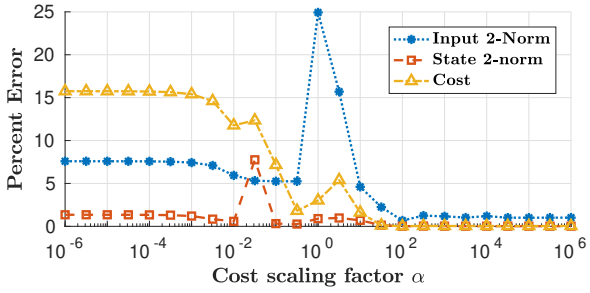


Fig. 7: Percent error of fixed-point versus double-precision floating-point implementations of FGM using round to zero and the fractional lengths from Theorem 2.

the increase in fractional bits leaving the overall number of bits nearly the same as the number needed for small α.

Reducing the fractional length to the minimum needed can lead to a large decrease in the required resources, power requirements, and solution times for FPGA implementations compared with simply choosing either floating-point or a larger fixed-point data type to get stability. This can be seen in Table II, where we present results for an FPGA implementation of the FGM using ProtoIP [9] targeting the Xilinx Zynq 7020 with a clock speed of 100MHz. An implementation with $f = 12$ uses 77% fewer memory

TABLE II: Resource usage for FGM implemented using ProtoIP [9] on a Zynq 7020 at 100MHz with $N = 20, i = 5$.

Fractional Length	Logic Resources[1]				Power (mW)	Solve Time (μs)
	LUT	FF	DSP	BRAM		
f=12	947	768	4	2	20	532.17
f=16	1,136	912	4	2	25	612.17
f=21	887	1,033	8	8	43	701.77
f=26	993	1,237	12	9	48	701.77
float[2]	2,161	1,545	5	14	51	982.17

[1] LUT = Lookup Tables, FF = Flip Flops, DSP = Digital Signal Processing cores, BRAM = Block RAM memory units
[2] Single-precision floating-point representation

blocks, 33% fewer Digital Signal Processing (DSP) computation blocks, and 25% less time when compared with an implementation for $f = 26$, and 85% fewer memory blocks taking 45% less time when compared to a single-precision floating-point implementation. This lower precision can lead to convergence to a suboptimal solution though, with experiments showing that using the minimum data type can lead to as much as a 15% increase in the cost and degraded closed-loop performance, as shown in Figure 7.

V. Conclusions

In this paper we developed two methods to size the data types in FGM to satisfy Requirement 1, then demonstrated the resource savings that can be achieved by using the smallest data type needed. Future work could explore the effect of the data type size on the suboptimality of the solution and stability of the closed-loop system to create data type sizing rules that provide closed-loop performance guarantees.

References

[1] S. Lucia, M. Kögel, P. Zometa, D. E. Quevedo, and R. Findeisen, "Predictive control, embedded cyberphysical systems and systems of systems - a perspective," *Annual Reviews in Control*, vol. 41, pp. 193–207, 2016.

[2] J. L. Jerez, P. J. Goulart, S. Richter, G. A. Constantinides, E. C. Kerrigan, and M. Morari, "Embedded online optimization for model predictive control at megahertz rates," *IEEE Transactions on Automatic Control*, vol. 59, no. 12, pp. 3238–3251, 2014.

[3] I. McInerney, E. C. Kerrigan, and G. A. Constantinides, "Bounding computational complexity under cost function scaling in predictive control," *arXiv preprint*, no. arXiv:1902.02221 [math.oc], 2019.

[4] C. N. Jones and M. Morari, "The double description method for the approximation of explicit MPC control laws," in *47th IEEE Conference on Decision and Control (CDC)*. Cancun, Mexico: IEEE, 2008, pp. 4724–4730.

[5] S. Richter, C. N. Jones, and M. Morari, "Computational complexity certification for real-time MPC with input constraints based on the fast gradient method," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1391–1403, 2012.

[6] L. N. Trefethen and M. Embree, *Spectra and Pseudospectra: The Behavior of Nonnormal Matrices and Operators*. Princeton, NJ, USA: Princeton University Press, 2005.

[7] D. S. Bernstein, *Matrix Mathematics*, 2nd ed. Princeton: Princeton University Press, 2009.

[8] J. Gutiérrez-Gutiérrez and P. M. Crespo, "Block toeplitz matrices: Asymptotic results and applications," *Foundations and Trends in Communications and Information Theory*, vol. 8, no. 3, pp. 179–257, 2012.

[9] B. Khusainov, E. C. Kerrigan, A. Suardi, and G. A. Constantinides, "Nonlinear predictive control on a heterogeneous computing platform," in *Proceedings of the 20th IFAC World Congress*, Toulouse, France, 2017.