A Simple Algorithm for Automated High-Efficiency Tracking

K.-M. Cheung
Communications Systems and Research Section

In this article, we describe a system-level optimization tool for high-efficiency tracking (HET). This scheme allows missions to plan the downlink data rates during a pass to (1) maximize overall data return and (2) minimize the tracking time.

I. Introduction

This article describes a system-level optimization tool for high-efficiency tracking (HET). HET refers to the operational procedure of coordinating the spacecraft and the ground systems to transmit and receive downlink data at different data rates during a pass. The goals are to (1) maximize overall data return and (2) minimize the tracking time. HET is one of the components in the proposed Demand Assess Automation paradigm.[1] This tool was developed to support Henry Hotz in demonstrating HET at the Flight System Testbed. A specific case of HET in the context of using punctured convolutional codes was described in [1]. This article describes an automated HET algorithm for a general communication scenario. This algorithm currently is implemented in Matlab and can be used together with the Multimission Spacecraft Analysis Subsystem (MSAS) telcommunication analysis tool to provide telecommunication forecasting and prediction services.

II. Definitions and Problem Formulation

In operation, there are only a finite number of data rates to choose from. The maximum possible data rate that the communication link can support is the largest one that can guarantee a certain required data quality, which usually is expressed in terms of bit-error rate (BER). The communication link refers to the entire communications path, from the information source, through all the coding and modulation steps, through the transmitter and the channel, up to and including the receiver with all its signal processing steps, and terminating at the information sink. The link budget is a balance sheet of the gains and losses of various parameters in the communication path. Many of these parameters are statistical or time varying, or both. The required BER of a link is a direct function of the bit signal-to-noise ratio, (E_b/N_o), which in turn is a function of the error-correction coding used. Due to the statistical nature (uncertainty) of many communication parameters, a safety margin or link margin, M, is required to guarantee the transmission data quality at any given time.

[1] H. B. Hotz, "Demand Access Automation," Draft Interim Report (JPL internal website), Jet Propulsion Laboratory, Pasadena, California, 1997.

To put this data-rate change problem into perspective, we need more definitions. Let P_r/N_o be the total received power-to-noise spectral density ratio, which encompasses all the power gain and loss terms in the communication path. P_r/N_o also can be expressed conveniently as

$$\frac{P_r}{N_o} = \frac{\text{EIRP } G_r/T^o}{\kappa L_s L_o} \tag{1}$$

where EIRP is the effective isotropic radiator power of the spacecraft, G_r/T^o is the receiver sensitivity of the ground system, κ is Boltzmann's constant, L_s is the space loss, and L_o denotes all other losses and degradation factors not specifically addressed in the above equation. Note that all the gains and losses are functions of time, and so P_r/N_o is a function of time. Let Rs_i be the ith symbol rate and rc_j be the jth code rate ($rc_j < 1$). The data rate, or bit rate Rb with symbol rate Rs_i and code rate rc_j, is given by $Rb = Rs_i rc_j$. To guarantee a reliable link at time t, the constraining dependency (in dB) between E_b/N_{oj}, M, P_r/N_o, Rs_i, and rc_j is expressed in the following equation:

Constraint 1

$$\frac{E_b}{N_o}(t)_j + M(t) \leq \frac{P_r}{N_o}(t) - Rs(t)_i - rc(t)_j \qquad \text{(in dB)}$$

Thus, the problem of HET is to find the right combination of i and j at time t that maximizes $Rb(t) = Rs(t)_i rc(t)_j$, subjected to Constraint 1.

III. A Practical Issue With Symbol-Rate Changes

The possibility of tracking symbol-rate changes was examined by Kinman.[2] It was suggested by Kinman that, in principle, if sufficiently accurate predictions of bit-rate changes are made available to the receiver, the symbol loop can be programmed to follow the changes with a minimal probability of falling out of lock. This, however, often is impractical, especially in the case of higher symbol rates. It also is suggested that the symbol loop in the receiver can track through a step change in symbol rate, but only if the step size is not too large and the steps are not placed too closely. It is unclear if tracking through symbol-rate changes is feasible in a real operational scenario, and it definitely increases the cost of the receiver. An alternate method for changing the data rate is to change the code rate. Code-rate change does not require very accurate timing predicts, which is the case for symbol-rate change.[3] The popular classes of error-correcting codes, e.g., convolutional codes and Reed–Solomon codes, offer a wide range of code rates to suit the purpose of data-rate change. Also, there are encoding and decoding architectures that allow switching from one code rate to another in a seamless fashion. We, therefore, prefer code-rate change to symbol-rate change, and we impose an additional operational constraint:

Constraint 2

Minimize the number of symbol-rate changes, Rs, in a pass.

To satisfy Constraint 2, we modify the data-rate search algorithm to favor code-rate change rather than symbol-rate change.

[2] P. Kinman, "Dynamic Changes in Telemetry Bit Rate During a Tracking Pass," unpublished report, Case Western Reserve University, Cleveland, Ohio, July 19, 1996.

[3] Symbol-rate change requires timing predicts accurate to the sample level, whereas code-rate change requires timing predicts accurate only to the symbol level.

IV. An Automated High-Efficiency Tracking Algorithm

HET requires good coordination between the spacecraft and ground systems since the link resource, P_r/N_o, is composed of both the spacecraft and the ground elements, as shown in Eq. (1). Under normal circumstances, the spacecraft EIRP does not drift or change significantly during a pass. We propose a simple handshake procedure by which the spacecraft will send the EIRP value to the ground at the beginning of a pass.[4] The ground system will model the P_r/N_o profile of the pass and compute the symbol rates, the code rates, and the update time. This information will be uploaded to the spacecraft along with the acknowledgement and initialization uplink message. In this way, both the spacecraft and the ground will have the same data-rate change information at the beginning of a pass. The details of the handshake protocol are being investigated as part of the Demand Access Automation Study.[5] In this article, we concentrate on the algorithm to compute the data-rate changes in a pass subjected to Constraints 1 and 2. This algorithm currently is implemented in Matlab.

V. An Example

We will here make up a P_r/No profile as an example to show how this HET algorithm works. This profile of an 8-hour pass consists of 17 P_r/N_o values (in dB) taken at every half-hour interval of the pass. These are shown in Table 1.

Table 1. Samples of P_r/N_o.

Time of pass, h	P_r/N_o, dB-Hz
0.0	23.5
0.5	25.8
1.0	25.1
1.5	28.4
2.0	29.1
2.5	30.1
3.0	33.5
3.5	33.0
4.0	33.8
4.5	34.5
5.0	33.0
5.5	34.1
6.0	31.0
6.5	31.5
7.0	29.0
7.5	26.1
8.0	24.0

Notice that the P_r/N_o samples are not quite a smooth function of time. This is to simulate the measurement uncertainty. We curve fit these points (denoted by circles in Fig. 1) with a fourth-order polynomial, $p_4(t)$, and the result is shown in Fig. 1. The available symbol rates are as follows:

---

[4] It can be part of the beacon mode message.

[5] H. B. Hotz, op cit.

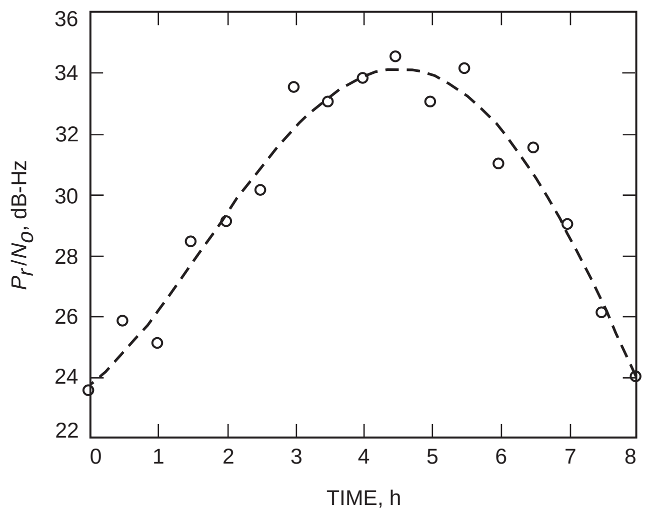

Fig. 1. P_r/N_o versus time.

(1) 100 symbols/s

(2) 200 symbols/s

(3) 400 symbols/s

(4) 1000 symbols/s

The error-correction codes used are constraint-length 7 convolutional codes with code rates k/n 1/3, 1/2, 2/3, and 3/4. Assuming the required BER is 10^{-3}, the required E_b/N_o's are as follows:

(1) $k/n = 1/3$, $E_b/N_o = 2.6$ dB

(2) $k/n = 1/2$, $E_b/N_o = 3.0$ dB

(3) $k/n = 2/3$, $E_b/N_o = 3.8$ dB

(4) $k/n = 3/4$, $E_b/N_o = 4.2$ dB

We also require a link margin of 1.5 dB. We present three cases here.

1. Symbol-Rate Change Only. We compute the total data return and the number of symbol-rate changes using all available symbol rates with each of the four (7, k/n) codes. The results are presented in Table 2. We show the data-rate profile for the (7,1/2) code in Fig. 2. The solid line represents the actual data rate, and the dotted line denotes a theoretical data-rate profile, assuming a continuum of symbol rate to perfectly fit the P_r/N_o profile.

2. Code-Rate Change Only. We compute the total data return and the number of code-rate changes, using all available code rates with each of the four symbol rates. The results are presented in Table 3. We show the data-rate profile for a symbol rate of 1000 symbols/s in Fig. 3.

3. Both Symbol-Rate and Code-Rate Changes. In this case, we use all four symbol rates and four code rates. The total data return is 10.30 Mbits; the number of symbol-rate changes is 6; the number of code-rate changes is 14; and the duration of the pass is 8.0 hours. We show the data-rate profile in Fig. 4.

Table 2. Performance for symbol-rate change only.

Code rate, k/n	Data return, Mbits	No. of symbol rate changes	Duration of pass, h
1/3	7.09	4	8.0
1/2	8.75	6	8.0
2/3	8.17	6	8.0
3/4	5.87	6	7.7

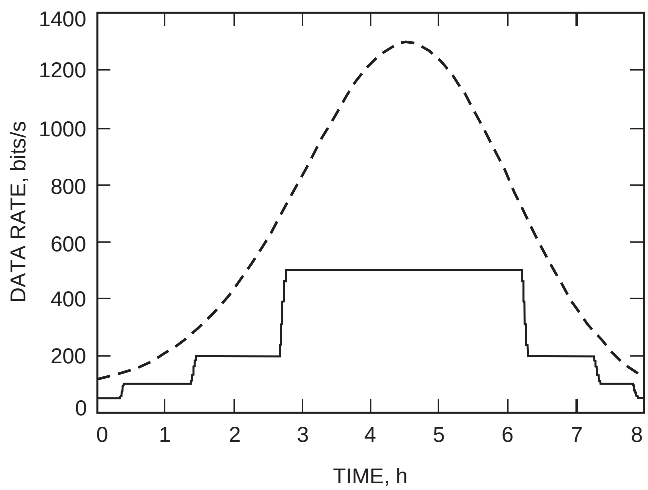

Fig. 2. Data-rate profile for symbol-rate change with (7,1/2) code.

Table 3. Performance for code-rate change only.

Symbol rate, symbols/s	Data return, Mbits	No. of code rate changes	Duration of pass, h
100	2.15	0	8.0
200	3.93	5	8.0
400	6.41	7	7.1
1000[a]	8.97	4	4.9

[a] HET using a high symbol rate and changing only the code rates.

VI. Discussion

We compare the above results and observe that HET using a *high symbol rate* and *changing only the code rates* (the last row in Table 3) produces an overall data return that is higher than all cases of HET using symbol-rate change (Table 2) and uses less tracking time. Despite the fact that this example is fictitious, this phenomenon is consistent with the result reported in [1], which is based on Galileo's real P_r/N_o profile. We think that HET using code-rate change is a practical and cost-effective way to

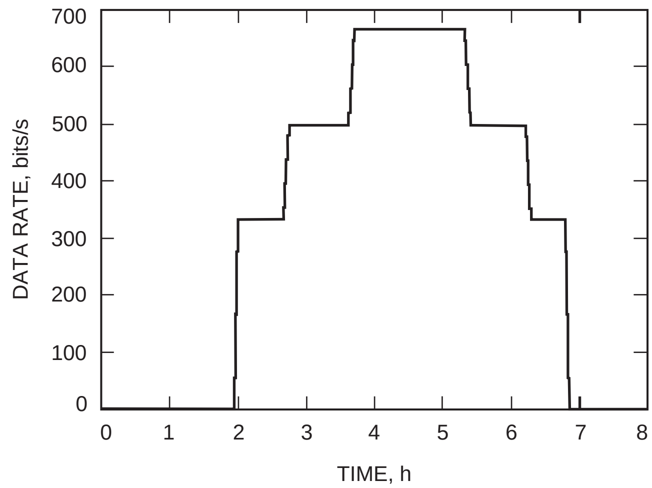

Fig. 3. Data-rate profile for code-rate change with a 1000 symbols/s symbol rate.

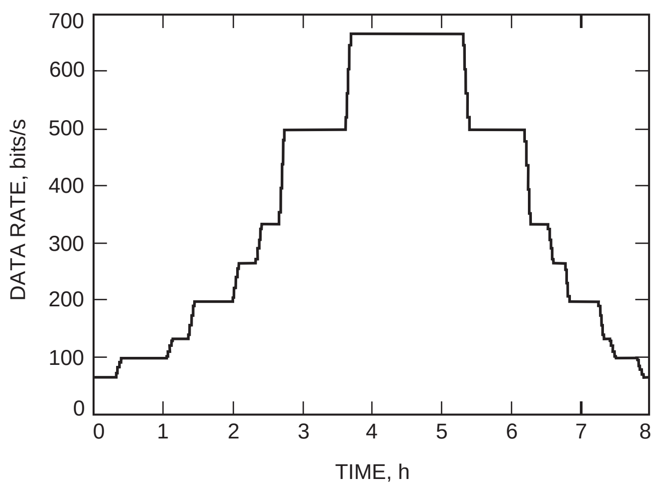

Fig. 4. Data-rate profile for simultaneous symbol-rate and code-rate changes.

maximize data return and minimize tracking time. To summarize, HET using code-rate change has the following advantages:

(1) High data return

(2) Seamless data-rate change

(3) Less tracking time

VII. Future Work

We currently have a Matlab tool that computes the optimal combination of Rs and rc that maximizes the total data return for a given pass. This tool first accepts a discrete set of P_r/N_o samples as a function of time, and curve fits these points with a fourth-order polynomial. It then compares this curve against the allowable code rates and symbol rates to generate a data-rate change profile that maximizes the data return for a pass. There is no constraint on the number of symbol-rate changes in the above calculation. However, it always is desirable to minimize the number of symbol-rate changes per pass because symbol-rate changes usually cause loss of track and require reacquisition. We propose enhancing the current Matlab tool to find the right combination of code rate and symbol rate subjected to a maximum number of symbol-rate changes.

To realize HET, we need to develop a detailed operational protocol, which might involve other onboard functions like command and data handling (CD&H) and sequencing. We need to decide

(1) Where the symbol-rate and code-rate changes occur in the data stream

(2) How the spacecraft and the ground coordinate the data-rate change information

(3) How the time-based sequencing process ensures data-rate change at exactly the right place

We also propose simulating the above protocol in a workstation to prove the concept as well as at the Telecommunication Development Laboratory (TDL) with hardware in the loop (receiver, decoder) to demonstrate operability.

Reference

[1] Y. Feria and K.-M. Cheung, "Seamless Data-Rate Change Using Punctured Convolutional Codes for Time-Varying Signal-to-Noise Ratios," *The Telecommunications and Data Acquisition Progress Report 42-120, October–December 1994*, Jet Propulsion Laboratory, Pasadena, California, pp. 18–28, February 15, 1995. http://tda.jpl.nasa.gov/tda/progress_report/42-120/120F.pdf