Discrete equations and software reliability growth models

Daisuke Satoh  
NTT Service Integration Laboratories  
3-9-11 Midori-cho Musashino-shi,  
Tokyo 180-8585 Japan  
satoh.daisuke@lab.ntt.co.jp

Shigeru Yamada  
Faculty of Engineering, Tottori University  
4-101 Koyama Tottori-shi,  
Tottori 680-0945 Japan  
yamada@sse.tottori-u.ac.jp

Abstract

We propose a criterion to determine a more appropriate software reliability growth model in the early development phase. The criterion with a discrete software reliability growth model determines the absolute worth of a model because the discrete software reliability growth model perfectly reproduces the parameter estimates when the data are used as an exact solution of the equation. The values of the proposed criterion of a discrete software reliability growth model are smaller than those of another software reliability growth model in all periods during the test phase with actual data sets. The discrete software reliability growth models are described with difference equations which have exact solutions. The models yield accurate parameter estimates in spite of a small amount of input data in an actual software testing. Therefore, the criterion and discrete software reliability growth models enable us to predict in the early development phase when software can be released.

1. Introduction

The size and complexity of computer-intensive systems have grown dramatically over the past decade, and this trend will certainly continue in the future. In complex software systems, reliability is the most important aspect of software quality. Software reliability assessment is thus a key technology for reducing software costs and producing highly reliable software. A quantitative measurement of software reliability is important for managing software development; it is needed in order to assess software performance and to minimize development and maintenance costs. To meet these needs, many models have been proposed and analyzed for measuring the growth in software reliability [4, 9, 11, 13, 15, 17]. These models basically estimate the number of software faults or software failures that will be debugged during the testing phase. A software failure is defined as an unacceptable departure of program operation caused by a software fault remaining in the software system.

These models have been useful to the extent that many different approaches have been explored. However, software engineers and managers have two problems. One is how to get parameter estimates of software reliability growth models. The other is that engineers and managers have had little guidance as to which models may be best or which may be best for a particular application.

Software reliability growth models generally do not provide accurate parameter estimates using the data available during the early testing phase. Accurate parameter estimates can only be obtained at the end of the testing. For a software reliability growth model to be practical, accurate parameter estimates must be obtained early in the testing. Recently, discrete analogs of software reliability growth models have overcome this problem [11, 13]. A differential equation that a model is specified as is discretized to a discrete equation that has an exact solution. The discrete equation tends to the differential equation as the time interval goes to zero. Therefore, the discrete equation keeps the characteristics of the differential equation. Although Tohma et al.[14] have proposed a discrete model, it does not have an exact solution. The discrete equation that has an exact solution is easily applied to a regression equation to get parameter estimates and has advantages compared to an ordinary difference equation [12]. It yields accurate parameter estimates in the early testing phase [11, 13].

In this paper, we discuss the second problem: the criterion as to which models may be best or which may be best for a particular application. Even a software reliability growth model that yields accurate parameter estimates in the early testing phase is not practical if software engineers and managers cannot decide on the most appropriate model for their applications. The criterion must be able to identify the most appropriate model in the early testing phase. A model that fits the data well in the early testing phase can provide a good fit in the later phase if a software does not include faults that have different difficulties to detect, e.g.,
the exponential-S-shaped software reliability growth model [3].

The remainder of this paper is organized as follows. In Section 2, we describe a discrete logistic curve model [13]. In Section 3, we describe a discrete Gompertz curve model [11]. In Section 4, we propose a criterion to determine an appropriate model and compare conventional and discrete logistic curve models and conventional and discrete Gompertz curve models. We show that the criterion identifies the better of the two models, a logistic curve model and a Gompertz curve model, when using even the small amount of data that is available early in the testing phase. We conclude with a summary in Section 5.

2. Discrete logistic curve models

A logistic curve model is one of the simplest models for estimating an S-shaped software reliability growth curve model. Although it is deterministic, it has been applied to many software-reliability growth processes in software testing [8, 10].

A logistic curve model is described as

\[
\frac{d L(t)}{d t} = \frac{\alpha}{k} L(t) (k - L(t)),
\]

where \( L(t) \) is the cumulative number of software failures by testing time \( t \), and \( \alpha \) and \( k \) are constant parameters to be estimated by regression analysis.

A solution of Eq. (1) is given by

\[
L(t) = \frac{k}{1 + m \exp(-\alpha t)},
\]

where \( k > 0, m > 0 \), and \( \alpha > 0 \). Parameter \( k \) represents the total number of potential software failures that could occur in an infinitely long duration or the initial number of faults already contained in the software (hereafter called initial fault content) as

\[
L(t) \to k \quad (t \to \infty).
\]

Parameter \( m \) is the constant of integration. It is defined as

\[
L(0) = \frac{k}{1 + m}.
\]

2.1. Discrete logistic curve model with Morishita’s equation

Morishita [6] proposed the following equation as a discrete equation of Eq. (1):

\[
L_{n+1} - L_n = \delta \frac{\alpha}{k} L_{n+1} (k - L_n),
\]

This equation states that the number of faults detected during time-difference is proportional to the product of the cumulative number of faults detected up to discrete time \( n + 1 \) and the number of remaining faults at discrete time \( n \) in the software.

It has an exact solution:

\[
L_n = \frac{k}{1 + m(1 - \delta \alpha)^N},
\]

Parameter \( m \) is defined as

\[
L_0 = \frac{k}{1 + m},
\]

where \( t_n = n \delta \).

The right-hand side of Eq. (6) in the small limit of \( \delta \) is the right-hand side of Eq. (2). Moreover, under the condition

\[
|1 - \delta \alpha| < 1,
\]

we get

\[
L_n \to k \quad as \quad t_n \to \infty.
\]

Thus Eq. (6) has the same characteristic as Eq. (2).

Let

\[
\alpha_c = \alpha \text{ in Eq. (2), and let}
\]

\[
\alpha_{dm} = \alpha \text{ in Eq. (6)}.
\]

Comparing Eqs. (2) and (6), we get

\[
\alpha_c = -\frac{1}{\delta} \log(1 - \delta \alpha_{dm}).
\]

To derive the regression equation for the parameters \( k, \alpha_{dm}, \) and \( m \), we rewrite Eq. (5) as

\[
Y_n = A + BL_{n+1},
\]

where

\[
Y_n = \frac{L_{n+1}}{L_n},
\]

\[
A = \frac{1}{1 - \delta \alpha_{dm}}
\]

\[
B = \frac{\delta \alpha_{dm}}{k(1 - \delta \alpha_{dm})}
\]

\[
t_n = n \delta.
\]

Parameters \( k, \alpha_{dm}, \) and \( m \) are estimated by

\[
\hat{k} = \frac{1 - \hat{A}}{\hat{B}},
\]

\[
\hat{\delta \alpha_{dm}} = 1 - \frac{1}{\hat{A}}
\]

\[
\hat{m} = \frac{\sum_{n=1}^{N} (\hat{k} - L_n)}{\sum_{n=1}^{N} (L_n(1 - \delta \alpha_{dm})^n)},
\]
where \( \hat{A} \) and \( \hat{B} \) are the estimates of parameters \( A \) and \( B \), respectively.

\( Y_n \) in Eq. (13) is independent of the time interval \( \delta \) because \( \delta \) is not used in Eq. (13). The estimates of \( \hat{k} \), \( \delta \hat{\alpha}_{dm} \), and \( \hat{m} \) are determined uniquely, whatever value of \( \delta \) we choose.

### 2.2. Discrete logistic curve model with Hirota’s equation

Hirota [1] discretized Eq. (1) as

\[
L_{n+1} - L_n = \frac{\alpha}{\delta} L_n (k - L_{n+1}).
\]

This equation states that the number of faults detected during time-difference is proportional to the product of the cumulative number of faults detected up to discrete time \( n \) and the number of remaining faults at discrete time \( n + 1 \) in the software.

He gave an exact solution:

\[
L_n = \frac{k}{1 + m(1 + \delta \alpha)^n}.
\]  

(22)

Parameter \( m \) is defined as

\[
L_0 = \frac{k}{1 + m}.
\]  

(23)

where \( t_n = n\delta \).

The limit of the right-hand side of Eq. (22) as \( \delta \) tends to 0 is the right-hand side of Eq. (2). Moreover, under the condition

\[
|\frac{1}{1 + \delta \alpha}| < 1,
\]

we get

\[
L_n \to k \quad \text{as} \quad t_n \to \infty.
\]  

(25)

Thus Eq. (22) has the same characteristic as Eq. (2).

Let

\[
\alpha_{dh} = \alpha \quad \text{in Eq. (22)}.
\]  

(26)

Comparing Eqs. (2) and (22), we get

\[
\alpha_c = \frac{1}{\delta} \log(1 + \delta \alpha_{dh}).
\]  

(27)

To derive the regression equation for parameters \( k, \alpha_{dh}, \) and \( m \), we rewrite Eq. (21) as

\[
Y_n = A + B L_{n+1}.
\]  

(28)

where

\[
Y_n = \frac{L_{n+1}}{L_n},
\]  

(29)

\[
A = \delta \alpha_{dh} + 1,
\]  

(30)

\[
B = \frac{\delta \alpha_{dh}}{k}, \quad \text{and}
\]  

(31)

\[
t_n = n\delta.
\]  

(32)

The estimates of parameters \( k, \alpha_{dh}, \) and \( m \) are given by

\[
\hat{k} = \frac{1 - \hat{A}}{\hat{B}},
\]  

(33)

\[
\hat{\delta} \hat{\alpha}_{dh} = \hat{A} - 1,
\]  

(34)

\[
\hat{m} = \frac{\sum_{n=1}^{N} (\hat{k} - L_n)}{\sum_{n=1}^{N} (L_n(1 + \frac{1}{\delta \alpha_{dh}})^n)}
\]  

(35)

where \( \hat{A} \) and \( \hat{B} \) are the estimates of parameters \( A \) and \( B \), respectively.

\( Y_n \) in Eq. (28) is independent of the time interval \( \delta \) because \( \delta \) is not used in Eq. (28). The estimates of \( \hat{k}, \delta \hat{\alpha}_{dh}, \) and \( \hat{m} \) are determined uniquely, whatever value of \( \delta \) we choose.

The regression equation (28) is the same as Eq. (13). Moreover, the same estimate of \( k \) is given by both equations. Though the estimate of \( \alpha \) depends on discrete equations, both discrete equations yield the same estimate of \( \alpha_c \).

The same estimate of \( m \) is obtained because

\[
1 - \delta \alpha_{dm} = \frac{1}{1 + \delta \alpha_{dh}} = \exp(-\alpha_c) = \frac{1}{A}.
\]  

(36)

Therefore, the models with Morishita’s and Hirota’s equations both give the same value of \( L_n \). We identify Morishita’s equation with Hirota’s hereafter. We call them a discrete logistic curve model.

### 3. Discrete Gompertz curve model

A Gompertz curve model is also one of the S-shaped software reliability growth curve models. The Gompertz curve model gave good approximations to a cumulative number of software faults observed in testing software[8]. Although it is deterministic, it has been modified into a stochastic software reliability growth model[16].

The Gompertz curve model is specified as

\[
\frac{d G(t)}{dt} = (\log b)G(t) \log \frac{G(t)}{k},
\]  

(37)

where \( G(t) \) is the cumulative number of software faults detected up to testing time \( t \). By integrating either equation and assuming that \( G(0) = ka, G(t) \) can be written as

\[
G(t) = ka^b^t \quad (k > 0, \ 0 < a < 1, \ 0 < b < 1),
\]  

(38)

where \( a, b, \) and \( k \) are parameters whose constant values are estimated using regression analysis. Parameter \( k \) represents the initial fault content as

\[
G(t) \to k \quad (t \to \infty).
\]  

(39)
Satoh proposed a difference equation of the Gompertz curve model [11]; it is a discrete analog of Eq. (37):

\[ G_{n+1} = G_n \left( \frac{G_n}{k} \right)^{\delta \log b} \].

An exact solution of this equation is

\[ G_n = ka^{(1 + \delta \log b)n} (k > 0, 0 < a < 1, \frac{1}{e} < b^\delta < 1). \]

Equation (41) satisfies Eq. (39) for any \( \delta \):

\[ G_n \rightarrow k \quad (n \rightarrow \infty). \]

We derived a regression equation from Eq. (40) by first rewriting it:

\[ \log G_{n+1} - \log G_n = -(\delta \log b) (\log k) + (\delta \log b) \log G_n. \]

From this equation, the regression equation is obtained:

\[ Y_n = A + B \log G_n, \]

where

\[ Y_n = \log G_{n+1} - \log G_n, \]

\[ A = -\delta (\log b)/(\log k), \]

\[ B = \delta \log b. \]

Using Eq. (44), we can estimate parameters \( k, a, \) and \( b \):

\[ \hat{k} = \exp \left( -\frac{A}{B} \right), \]

\[ \hat{a} = \exp \left( \frac{\sum_{n=1}^{N} \log \frac{G_n}{k}}{\sum_{n=1}^{N} (1 + \delta \log b)^n} \right), \]

\[ \hat{b} = \exp \left( \frac{\hat{B}}{\delta} \right), \]

where \( \hat{a}, \hat{b}, \) and \( \hat{k} \) are the estimated values of \( a, b, \) and \( k \), and \( A \) and \( \hat{B} \) are the estimated values of \( A \) and \( B \) in Eq. (44).

\( Y_n \) in Eq. (44) is independent of time interval \( \delta \) because \( \delta \) is not used in Eq. (44). Hence, the same estimates of \( \hat{k}, \hat{a}, \) and \( \hat{b} \) are obtained, even when we choose any value of \( \delta \). Therefore, Eq. (41) is determined uniquely for any value of \( \delta \).

4. Criteria for comparison and determining the absolute worth of a model

The discrete logistic curve model and the discrete Gompertz curve model can predict the total number of potential software failures in the early test phase [11, 13].

To predict the total number of potential software failures, it is also important and difficult to determine which model is the most appropriate model to use in the early testing phase [2, 7].

To determine an appropriate model, we propose the following criterion:

\[ C = \frac{1}{N} \sum_{i=1}^{N} \left( \frac{X_i - \hat{X}_i}{X_i} \right)^2, \]

where \( N \) denotes the number of available data points, \( X_i \) the real data of the \( i \)th data point, and \( \hat{X}_i \) the value estimated by a software reliability growth model. Although the error is usually evaluated as the mean squared error (MSE), MSE is not appropriate to determine the appropriate model because it is significantly affected by the absolute values of the data. The proposed criterion, however, is not significantly affected by the absolute values of the data; rather, it is affected by the but ratio between the data and estimates.

We arranged four data sets; data set A: an exact solution of the logistic equation for each period \((t = 0 \text{ to } 21)\), data set B: an exact solution of the Gompertz equation for each period \((t = 0 \text{ to } 25)\), data set C: debugging data for a software, and data set D: data used by Mitsutaka [5].

Comparing the four values of the proposed criterion corresponding to the four models by using data sets A and B, which are not actual data sets, is important because the proposed criterion has to determine an appropriate model when an exact solution of the appropriate model is used as a data set.

4.1. Data set A: exact solution of a logistic equation

We used the same values as those Satoh and Yamada [13] used: \( k = 100, \alpha = 0.8, \) and \( m = 999 \). This data inflected at the point where \( t^* = 8.63 \) and \( L(t^*) = 50 \). In our evaluation, we used the same time interval of 1.

We analyzed four sets of data set A: the four sets covered the data up to (A-i) the first three data \((t = 0, 1, 2)\), (A-ii) the data just before the point of inflection \((t = 0, 1, \ldots, 8)\), (A-iii) the data just after the point of inflection \((t = 0, 1, \ldots, 9)\), and (A-iv) the ceiling \((t = 0, 1, \ldots, 21)\).

The results of the comparisons among the models are shown in Table 1, where C-I, D-I, C-G, and D-G denote the conventional logistic curve model, the discrete logistic curve model, the conventional Gompertz curve model, and the discrete Gompertz curve model, respectively. The discrete logistic curve model matched the data completely. The discrete logistic curve model reproduces the values of the parameters of an exact solution when the exact solution is used as the input data [13]. Thus, the values of criterion C were exactly zero. The conventional logistic curve model would be expected to provide a better fit in terms of criterion
C because each data set: (A-i), ..., (A-iv) was composed of an exact solution of the logistic equation. However, the conventional logistic curve model provided a poorer fit as shown in Table 1 than did the conventional Gompertz curve model in each data set: (A-i), ..., (A-iv). The conventional logistic curve model is based on an ordinary forward or central difference equation of the logistic equation. The discretization error causes the poor fit as shown in Fig. 1, which shows comparison between data set (A-iv) and estimates by the conventional logistic curve model.

<table>
<thead>
<tr>
<th>Table 1. Criterion C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data set</td>
</tr>
<tr>
<td>A-i</td>
</tr>
<tr>
<td>A-ii</td>
</tr>
<tr>
<td>A-iii</td>
</tr>
<tr>
<td>A-iv</td>
</tr>
</tbody>
</table>

Figure 1. Comparison between data set (A-iv) and estimates by the conventional logistic curve model

We estimated \( k \): initial fault content by using each model. The results of comparisons are shown in Table 2. The value of \( k \) estimated using the discrete logistic curve model matched the target value for all the four data sets. The conventional logistic curve model provided the estimates of \( k \), which became more accurate as the number of available data points increased. The estimate provided by using data set (A-iv) gave a good approximation to the target value.

The discrete and conventional Gompertz curve models, on the other hand, had much worse accuracy than the discrete and conventional logistic curve models for all the four data sets. First several values of an exact solution of Gompertz equation increase faster than ones of logistic equation when both equations have the same value of parameter \( k \). Hence, estimates by the Gompertz models were much larger than the target value.

<table>
<thead>
<tr>
<th>Table 2. Estimated parameter ( k )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data set</td>
</tr>
<tr>
<td>A-i</td>
</tr>
<tr>
<td>A-ii</td>
</tr>
<tr>
<td>A-iii</td>
</tr>
<tr>
<td>A-iv</td>
</tr>
</tbody>
</table>

4.2. Data set B: Exact solution of a Gompertz equation

We arranged data as an exact solution of Eq. (37) for each period \((t = 0 \text{ to } 25)\). We used the same parameter values of \( k = 100 \), \( a = 0.01 \), and \( b = 0.5 \) that Satoh [11] used. This data inflected at the point where \( t^* = 1.44209 \) and \( G(t^*) = 36.7879441 \). We analyzed three sets of this data: the three sets covered the data up to (B-i) just before the point of inflection \((t = 0, 1, 2)\), (B-ii) just after the point of inflection \((t = 0, 1, 2, 3)\), and (B-iii) the ceiling \((t = 0, 1, \ldots, 25)\).

The results of the comparisons among models are shown in Table 3. The discrete Gompertz curve model matched the data completely. The discrete Gompertz curve model reproduces the values of the parameters of an exact solution when the exact solution is used as the input data [11]. Thus, the values of criterion C were exactly zero. The discrete and conventional logistic curve models provided a poorer fit in terms of criterion C as shown in Table 3 than did the discrete and conventional Gompertz curve models in each data set: (B-i), ..., (B-iii). This result is reasonable because these data sets are composed of an exact solution of the Gompertz equation.

<table>
<thead>
<tr>
<th>Table 3. Criterion C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data set</td>
</tr>
<tr>
<td>B-i</td>
</tr>
<tr>
<td>B-ii</td>
</tr>
<tr>
<td>B-iii</td>
</tr>
</tbody>
</table>

We estimated \( k \) by using each model. The results of comparisons are shown in Table 4. The value of \( k \) estimated us-
ing the discrete Gompertz curve model matched the target value for all the three data sets. The conventional Gompertz curve model provided the estimates of \( k \), which became more accurate as the number of available data points increased. The estimate by using data set (B-iii) gives a good approximation to the target value. However, the discrete and conventional logistic curve models had much worse accuracy than the discrete and conventional Gompertz curve models for all the three data sets.

### Table 4. Estimated parameter \( k \)

<table>
<thead>
<tr>
<th>Data set</th>
<th>C-I</th>
<th>D-I</th>
<th>C-G</th>
<th>D-G</th>
</tr>
</thead>
<tbody>
<tr>
<td>B-i</td>
<td>12.85</td>
<td>38.46</td>
<td>46.53</td>
<td>100</td>
</tr>
<tr>
<td>B-ii</td>
<td>31.44</td>
<td>55.36</td>
<td>78.16</td>
<td>100</td>
</tr>
<tr>
<td>B-iii</td>
<td>96.30</td>
<td>97.27</td>
<td>99.63</td>
<td>100</td>
</tr>
</tbody>
</table>

4.3. Data set C: Actual data set 1

We obtained an actual data set [13], which was debugging data for a software. Satoh and Yamada evaluated the parameter estimates and showed the discrete logistic curve model fit the actual data very well as shown in Fig. 2. We compared the discrete logistic curve model and the discrete Gompertz curve model because both models yield accurate parameter estimates.

![Figure 2. Comparison both models with actual data](image)

We evaluated the parameter estimates with all the data and with the data available in the early test phase and calculated criterion \( C \) by using the parameter estimates. As shown in Fig. 3, the values of criterion \( C \) of the discrete logistic curve model were smaller than those of the discrete Gompertz curve model in all periods during the test phase.

We estimated \( k \). The results of comparisons are shown in Fig. 4. The estimated values are normalized by the total number of actual software failures. The discrete logistic curve model provided more accurate parameter estimates than the discrete Gompertz model. Moreover, the discrete logistic curve model provided accurate parameter estimates even in the early test phase.

4.4. Data set D: Actual data set 2

We arranged a second actual data set [5] that was appropriate to the Gompertz curve model as Mitsuhashi showed. As shown in Fig. 5, the discrete Gompertz curve model fit the actual data very well [11].

We evaluated the parameter estimates with all the data and with the data available in the early test phase, and calculated criterion \( C \) by using the parameter estimates. As shown in Fig. 6, the values of criterion \( C \) of the discrete Gompertz curve model were smaller than those of the discrete logistic curve model in all periods during the test phase.

We estimated \( k \) by using each model. The results of comparisons are shown in Fig. 7. The discrete Gompertz curve model provided more accurate parameter estimates than the discrete logistic curve model. Moreover, the discrete Gompertz curve model provided accurate parameter estimates even in the early test phase.
5. Conclusion

We have proposed a criterion to determine a more appropriate software reliability growth model. The proposed criterion is not significantly affected by the absolute values of the data; rather, it is affected by the ratio between the data and estimates. However, the mean squared error is significantly affected by the absolute values of the data. The criterion is used with conventional and discrete software reliability growth models. The criterion with discrete software reliability growth models determines the absolute worth of a model because the discrete software reliability growth model perfectly reproduces the parameter estimates when the data are used as an exact solution of the equation. The conventional model yielded worse parameter estimates and provided a poorer fit in terms of the criterion than did the discrete model.

We evaluated the proposed criterion with two actual data sets. One set was appropriate to a logistic curve model and the other was appropriate to a Gompertz curve model. The values of the proposed criterion of the discrete logistic curve model were smaller than those of the discrete Gompertz curve model in all periods during the test phase with the former actual data set. The discrete logistic curve model provided accurate parameter estimates with even a small amount of the data. And the values of the criterion of the discrete Gompertz curve model were smaller than those of the discrete logistic curve model in all periods during the test phase with the latter actual data set. The discrete Gompertz curve model provided accurate parameter estimates with even a small amount of the data. Therefore, the proposed criterion and the discrete model enable us to identify the potential number of faults in software in the early testing phase although further studies are left to manifest reasons why the discrete models yield more accurate parameter estimates in the early testing phase.

Acknowledgments

One of the authors (D. S) is grateful to Mr. Takaaki Fukuda and Mr. Keisuke Ohmori at NTT-AT for providing the data used in this work.

References


Figure 6. Criterion in terms of available data points

Figure 7. Estimates of parameter $k$


A. Data set D[5]

Table 5. Data set D

<table>
<thead>
<tr>
<th>Week</th>
<th>Cumulative number of software faults</th>
<th>Week</th>
<th>Cumulative number of software faults</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>248</td>
<td>31</td>
<td>4351</td>
</tr>
<tr>
<td>2</td>
<td>262</td>
<td>32</td>
<td>4401</td>
</tr>
<tr>
<td>3</td>
<td>372</td>
<td>33</td>
<td>4439</td>
</tr>
<tr>
<td>4</td>
<td>526</td>
<td>34</td>
<td>4488</td>
</tr>
<tr>
<td>5</td>
<td>742</td>
<td>35</td>
<td>4548</td>
</tr>
<tr>
<td>6</td>
<td>958</td>
<td>36</td>
<td>4596</td>
</tr>
<tr>
<td>7</td>
<td>1215</td>
<td>37</td>
<td>4629</td>
</tr>
<tr>
<td>8</td>
<td>1471</td>
<td>38</td>
<td>4680</td>
</tr>
<tr>
<td>9</td>
<td>1738</td>
<td>39</td>
<td>4713</td>
</tr>
<tr>
<td>10</td>
<td>1936</td>
<td>40</td>
<td>4749</td>
</tr>
<tr>
<td>11</td>
<td>1971</td>
<td>41</td>
<td>4783</td>
</tr>
<tr>
<td>12</td>
<td>2147</td>
<td>42</td>
<td>4817</td>
</tr>
<tr>
<td>13</td>
<td>2258</td>
<td>43</td>
<td>4849</td>
</tr>
<tr>
<td>14</td>
<td>2418</td>
<td>44</td>
<td>4877</td>
</tr>
<tr>
<td>15</td>
<td>2567</td>
<td>45</td>
<td>4901</td>
</tr>
<tr>
<td>16</td>
<td>2688</td>
<td>46</td>
<td>4928</td>
</tr>
<tr>
<td>17</td>
<td>2809</td>
<td>47</td>
<td>4950</td>
</tr>
<tr>
<td>18</td>
<td>2925</td>
<td>48</td>
<td>4970</td>
</tr>
<tr>
<td>19</td>
<td>3026</td>
<td>49</td>
<td>4998</td>
</tr>
<tr>
<td>20</td>
<td>3205</td>
<td>50</td>
<td>5024</td>
</tr>
<tr>
<td>21</td>
<td>3348</td>
<td>51</td>
<td>5060</td>
</tr>
<tr>
<td>22</td>
<td>3476</td>
<td>52</td>
<td>5085</td>
</tr>
<tr>
<td>23</td>
<td>3573</td>
<td>53</td>
<td>5088</td>
</tr>
<tr>
<td>24</td>
<td>3719</td>
<td>54</td>
<td>5090</td>
</tr>
<tr>
<td>25</td>
<td>3750</td>
<td>55</td>
<td>5110</td>
</tr>
<tr>
<td>26</td>
<td>3952</td>
<td>56</td>
<td>5129</td>
</tr>
<tr>
<td>27</td>
<td>4048</td>
<td>57</td>
<td>5139</td>
</tr>
<tr>
<td>28</td>
<td>4137</td>
<td>58</td>
<td>5167</td>
</tr>
<tr>
<td>29</td>
<td>4251</td>
<td>59</td>
<td>5186</td>
</tr>
<tr>
<td>30</td>
<td>4301</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>